X.509 Version 2 certificate

X.509 was created to support the authentication of the entries in an X.500 directory. The latest version, the third, has evolved beyond its X.500 roots. Currently, version 3 is the official standard. We will first describe X.509v2, before moving on to the extensions added under version 3.

The X.509v2 certificate is illustrated in Fig. 3.7. It contains the following fields.

- Version: The X.509 version that the certificate conforms to.
- Serial number: A unique number assigned to the certificate by its issuing CA.
- CA signature algorithm: An identifier for the algorithm used by the CA to sign the certificate. Identifiers are further discussed below under Object Registration.
- Issuer name: The X.500 name of the issuing CA.
- Validity period: A pair of dates/times between which the certificate is considered valid.
- Subject name: The X.500 name of the entity who holds the private key corresponding to the public key being certified.
- Subject public key information: The value of the subject's public key along with an identifier of the algorithm with which the key is intended to be used.
- Issuer unique identifier (optional, version 2 only): A bit string used to make the X.500 name of the issuing CA unambiguous. It is possible for an X.500 name to be assigned to a particular entity, then de-assigned, then re-assigned to a new entity.[1] The unique identifier fields address this concern. These fields are not widely used, as they have turned out to be difficult to manage and are ignored or omitted in most implementations. The preferred method used to address this problem is to design the RDNs in such a way as to ensure that they are *never* reused. For example, rather than use just the CommonName attribute, a better form of RDN might use both the CommonName and an EmployeeNumber.
- Subject unique identifier (optional, version 2 only): A bit string used to make the X.500 name of the subject unambiguous.

[1] For example, in Fig. 3.6, if Mr. Riel changes companies, his DN, in particular the Organization=Bombardier component, is no longer valid and so is de-assigned. Later, if another person named Louis Riel comes to work for Bombardier, he would be assigned the same DN as the first Louis Riel.

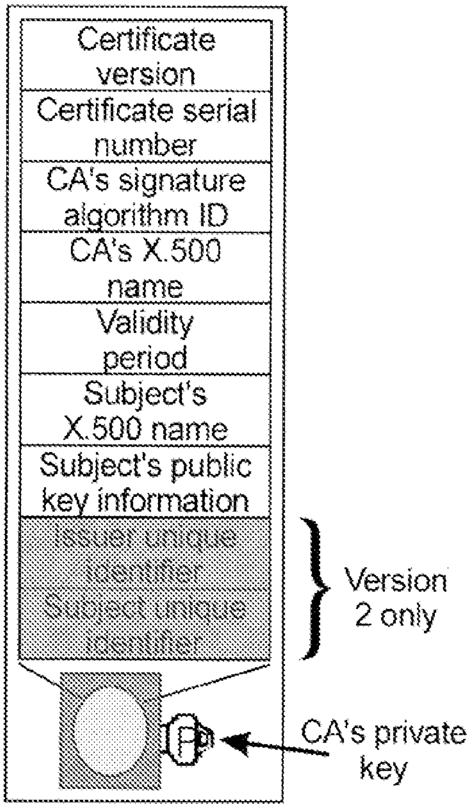


Fig. 3.7 The X.509 version 2 certificate

Because of X.509's close ties with X.500, its CAs are usually arranged in a hierarchy that closely follows the X.500 DIT.

X.509, and X.500, were originally designed in the mid-1980's, before the current explosive growth of the Internet. They were therefore designed to operate in an offline environment, where computers are only intermittently connected to each other. X.509 thus employs CRLs. Versions 1 and 2 of X.509 use very simple CRLs that do not address size issues and the time-granularity problem. Version 3 makes several attempts to solve these problems, with varying success. Fig. 3.8 illustrates the CRL format used in X.509 versions 1 and 2.

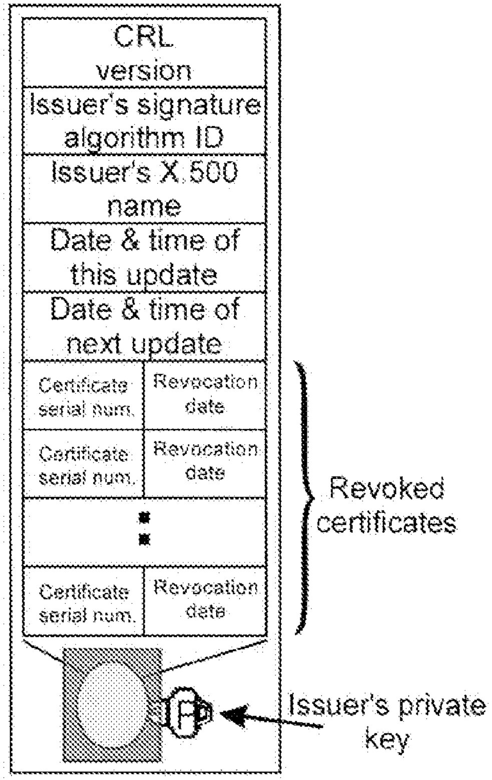


Fig. 3.8 X.509 version 1 CRL format

X.509 Version 3

Version 3 introduced significant changes to the X.509 standard. The fundamental change was to make the certificate and CRL formats extensible. X.509 implementers can now define certificate contents as they see fit. Also, a number of standard extensions were defined to provide key and policy information, subject and issuer attributes, certification path constraints, and enhanced CRL functionality. These extensions are fully described in [3.10] and elsewhere. We concentrate here on those extensions which affect the basic PKI characteristics of X.509.

Version 3 Certificate Extensions

Certificate policies and policy mapping. X.509v3 gives CAs the ability to include with the certificate a list of policies that were followed in creating the certificate. These policies are intended to help users decide if a certificate is suitable for a particular purpose. For example, a policy might indicate that a certified key can be used for casual email messages but not for financial transactions. In general, a certificate policy indicates such things as CA security procedures, subject identification measures, legal disclaimers or provisions, and others. Policy mapping allows a CA to indicate whether one of its policies is equivalent to another CA's policy.

Alternative names. An X.509v3 certificate can contain one or more alternative names for the subject or issuer. This allows X.509 to operate without an underlying X.500 directory. Examples of alternative names include email addresses and World Wide Web universal resource locators. Implementers can also define their own alternative name forms. Alternative names can also be used to identify the issuer of a CRL.

Subject directory attributes. This extension allows any of the subject's X.500 directory entry attribute values to be included in the certificate. This allows the certificate to carry additional identifying information beyond the subject's name(s).

Certification path constraints. These extensions allow CAs to link up their infrastructures in meaningful ways. A CA can restrict the kinds of certification paths that can grow from certificates it issues for other CAs. The CA can state whether or not a certificate's subject is in a fact a CA (to prevent an end user from fraudulently acting as a CA). The CA can also constrict paths growing from the certificate to certificates issued in a particular name space (e.g., within a given Internet domain) and/or to certificates that follow a specific set of certification policies. This is an important extension because it allows CAs to employ a *progressive-constraint* trust model that prevents the formation of infinite certification paths. The concept is illustrated in Fig. 3.9. User a uses D as her certification authority, so she places complete trust in D. D has certified another certification authority, E, for example only trusting E to issue certificates for other CAs (perhaps E performs some kind of national CA registration). Constraint X would then state that D only trusts E to certify other certification authorities.

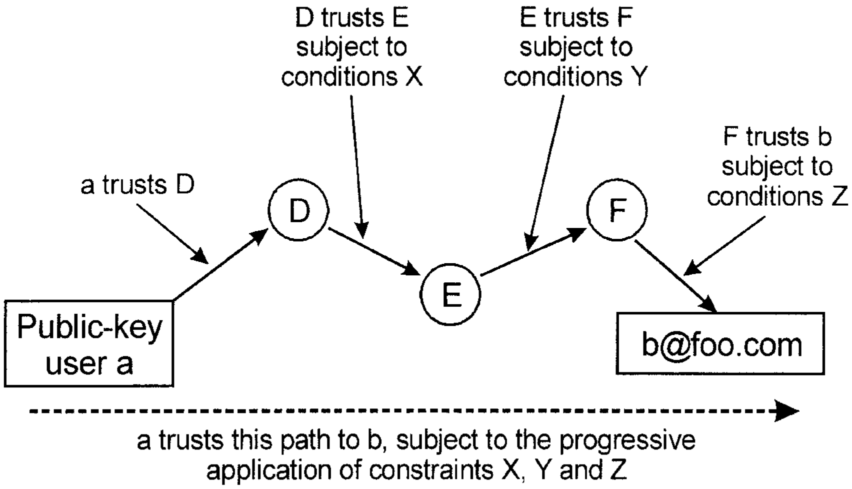

D trusts E subject to conditions X
E trusts F subject to conditions Y
F trusts b subject to conditions Z
a trusts D
D
E
F
Public-key user a
b@foo.com
a trusts this path to b, subject to the progressive application of constraints X, Y and Z

Fig. 3.9 A progressive-constrained trust chain

E has issued a certificate for certification authority F stating that it only trusts F to issue certificates for end users in the domain foo.com. So constraint Y would state that E trusts certificates issued by F only if they certify an end user *and* that user's name is in the foo.com domain. Finally, F issues a certificate for user b, but only trusts b for casual email (as opposed to, say, making financial commitments on F's behalf). So constraint Z states that the certificate issued for b by F should only be used for casual email.

In this way the unlimited trust that a places in D becomes increasingly constrained as the certification path grows. When a obtains a certificate for b she knows that she should only use it for casual email, and she has greater confidence in the strength of the authentication than with, say, PGP's web of trust because she can see how trust has been restricted along the certification path. Given these constraints, she would not accept a certificate issued by E for b (or any other user), nor would she accept any certificates from any certification authority certified by F. If CAs define the tightest practical conditions when they certify other CAs, then as a certification path grows it becomes progressively more constrained until it can grow no longer.

Version 3 CRL Extensions

CRL number and reason codes. Each CRL issued for a given certificate population is assigned a number from a monotonically increasing sequence. This allows users to determine if a CRL was missed. Also, each certificate in a CRL can now have a revocation reason attached to its CRL entry. These two extensions allow for the more sophisticated CRL extensions described below.

CRL distribution points. This extension helps reduce the sizes of CRLs processed by a CA's users. Rather than forcing users to accept the full CRL, the CA can partition the CRL in some way, and issue each partition from a different distribution point. For example, a corporate CA might issue a different CRL for each division of the company. Then when a user wants to verify a certificate for someone from a particular division, they need only check that division's CRL rather than the full CRL. Another way of partitioning the CRL is according to revocation reason. Routine revocations, for example, those due to a name change, can be placed on a different CRL than revocations due to a security compromise. The compromise list can then be updated and checked more frequently without having to also process all the routine revocations that might occur.

Delta-CRLs. This extension provides another method of reducing CRL sizes. Rather than issue a full CRL (or a full partition of a CRL), the CA can simply issue a list of the changes that have occurred since the last time a full CRL was issued. Users that maintain their own CRL database can use a delta-CRL to keep their copies updated without having to download and process all the entries of a full CRL, saving bandwidth and computing time.

Indirect CRLs. This extension allows a CRL to be issued from an entity other than the CA that issued its certificates. This allows for CRL clearing houses that would gather the CRLs from multiple CAs and provide one distribution point for them all.

All of these CRL extensions still do not overcome the fundamental time-granularity problem. Even with partitioned CRLs and frequent delta-CRL issuance, there is still a window of opportunity for a compromised certificate to be used. The X.509v3 framework can be used for online operation, avoiding the need for CRLs altogether. PKIX defines a *online certificate status protocol* [3.48] to do this work.

Object Registration

The extensibility of X.509v3 gives it a tremendous amount of flexibility. However, the way in which that extensibility is provided hampers the widespread application of user-defined extensions for a global PKI.

Every time X.509 needs to identify some object, such as a signature algorithm, certification policy, user-defined alternative name, or a user-defined extension, it uses an internationally defined *object identifier* (OID) mechanism. An OID is a numeric value, composed of a sequence of integers, that is unique with respect to all other OIDs.

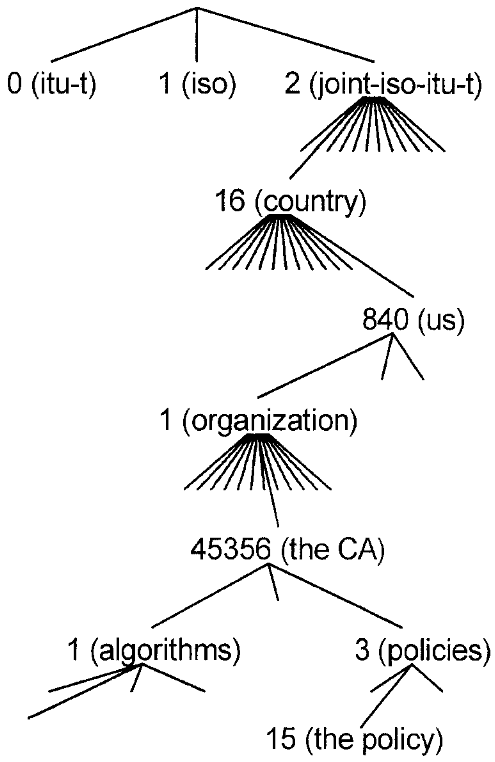


Fig. 3.10 Object registration example

The OIDs are assigned following a hierarchical structure of value-assigning authorities [3.10, 3.64, 3.66]. Essentially, any company or organization can become a value-assigning authority. The company is itself assigned a value that serves as a prefix for all the values that it defines. Take, for example, the OID pictured in Fig. 3.10. Imagine a CA operating in the United States. The CA would be assigned an OID, say 2-16-840-1-45356.[2] This OID would then be the prefix used for the OIDs of any objects that the CA cares to register. The CA might want to register a

[2] The numbers only have meaning within the hierarchy. The leading 2 indicates the branch of the hierarchy administered jointly by the ISO and ITU. The 16 is the number assigned to the branch used by national registration authorities. 840 is the country code for the U.S., whose national registration authority (ANSI) uses 1 as the prefix for all the organizations it registers. The 45356 is simply a number assigned to the CA by ANSI.

particular certification policy to which it has assigned a number, say 15, beneath the policy's branch of its hierarchy (branch number 3, for example). Then the CA's policy could be identified as object number 2-16-840-1-45356-3-15.

This system works well for assigning numbers to objects, and it is used extensively in X.509. For example, if the CA in Fig. 3.10 were to use its policy in a certificate, that policy would be identified solely by its OID. Difficulty arises, however, when OIDs are used without prior agreement as to their meaning. If the CA in our example wants to use their policy in their certificates, they have to ensure that the meaning of the OID identifying their policy is known *a priori* by any entity wishing to use the certificate. Otherwise, when an ignorant entity encounters the value 2-16-840-1-45356-3-15 it will have no idea how to interpret the policy.

Confusion can also arise when the same object is assigned multiple OIDs. For example, imagine that two CAs have each assigned an OID to a particular signature algorithm, such as SHA-with-RSA. As long as the CAs and their users don't interact, there will be no problems. However, if a user from one CA tries to use the other CA's certificate, they won't recognize the second CA's OID for SHA-with-RSA, and might assume that they can't verify the signatures of the certificate's subject even though they may be perfectly capable of doing so. The problem is compounded if the two CAs ever try to link their infrastructures. Then the CAs must either let all their users know that the two OIDs are equivalent, or one CA (or both) has to change its OID and communicate that change to all its users.

The OID problem prevents X.509's extensibility from being used freely on a large scale, since whoever creates a new extension must ensure that the relevant OIDs are known by all parties concerned. There is at present no systematic method for determining the meaning of an OID. They are neither regularly published, nor are they reliably listed in a central registry. The only way you can find out the meaning of an OID is to have the OID's creator tell it to you.

3.2.3 X.509 on the Internet

Privacy Enhanced Mail

Privacy enhanced mail (PEM) was proposed in early 1993 as an Internet standard for cryptography-enhanced email (see [3.34-3.37]). The intention was to endow Internet email with confidentiality, authentication, message integrity assurance, and non-repudiation of origin, using public-key cryptography. To this end, [3.35] proposed an Internet PKI to support PEM. The standard never caught on in the Internet community for various reasons, one of which was that its proposed PKI model proved to be a poor fit to the Internet's peer-based structure.

PKIX

The PKIX working group (WG) was established in the fall of 1995 with the intent of developing Internet standards needed to support an X.509-based PKI. Several informational and standards track documents in support of the original goals of the WG have been approved by the IESG. The first of these standards, RFC 2459 [3.43], profiles the X.509 version 3 certificates and version 2 CRLs for use in the Internet. The certificate management protocol (CMP) (RFC 2510) [3.44], the online certificate status protocol (OCSP) (RFC 2560) [3.48], and the certificate management request format (CRMF) (RFC 2511) [3.45] have been approved, as have profiles for the use of LDAP v2 for certificate and CRL storage (RFC 2587) [3.50] and the use of FTP and HTTP for transport of PKI operations (RFC 2585) [3.49]. RFC 2527 [3.46], an informational RFC on guidelines for certificate policies and practices also has been published, and the IESG has approved publication of an information RFC on use of KEA (RFC 2528) [3.47] and is expected to do the same for ECDSA. Work continues on a second certificate management protocol, CMC, closely aligned with the PKCS publications and with the cryptographic message syntax (CMS) developed for S/MIME. A roadmap, providing a guide to the growing set of PKIX document, is also being developed as an informational RFC.

The working group is now embarking on additional standards work to develop protocols that are either integral to PKI management, or that are otherwise closely related to PKI use. Work is ongoing on alternative certificate revocation methods. There also is work defining conventions for certificate name forms and extension usage for "qualified certificates," certificates designed for use in (legally binding) non-repudiation contexts. Finally, work is underway on protocols for time stamping and data certification. These protocols are designed primarily to support non-repudiation, making use of certificates and CRLs, and are so tightly bound to PKI use that they warrant coverage under this working group.

3.3 Credential-Based PKI Systems

Much recent work has focused on moving away from identity-based PKIs to a more general system based on attributes or credentials. At present, there are two main proposals for this kind of system: the simple distributed security infrastructure (SDSI), and the simple public key infrastructure (SPKI).

3.3.1 Simple Distributed Security Infrastructure (SDSI)

SDSI was created by Ron Rivest and Butler Lampson and is described in [3.54]. SDSI is designed to facilitate the construction of secure systems and provides simple, clear terminology for defining access-control lists and security policies. It is also an attempt to move away from identity-based certification and towards a system based on roles and credentials.

The SDSI system is key-centric. Rather than attach a public key to an identity, SDSI entities are the keys themselves. Specifically, SDSI calls its entities "principals" and defines them to be digital signature verification keys. The idea is that the key is a proxy for the individual who controls its associated private key. Thus SDSI principals are public keys that can make declarations by issuing verifiable signed statements.

3.3.2 SDSI Certificates

Those signed statements come mainly in the form of certificates. SDSI provides for three types of certificates, and any principal can create any kind of certificate. In no particular order, the three certificate types are:

- Identity certificates
- Group-membership certificates
- Name-binding certificates

SDSI identity certificates bind some identifying information to a principal. The main goal of a SDSI identity certificate is to allow a human reader to identify the individual behind a principal. As such, the certificates are designed to be human-friendly, usually containing some free-form text and perhaps a photograph or other information. Machine-readable tags, such as OIDs, are not used, because SDSI's designers believe that determining the identity behind a principal will almost always involve a human anyway.

Identity certificates play a relatively small role in the SDSI system. More important are group-membership certificates, which assert that a principal does or does not belong to some group (more on SDSI groups below), and name-binding certificates, which bind a name to some value (typically, but not necessarily, a principal).

3.3.3 SDSI Names

When a principal creates a certificate binding a name to some value, that name is said to exist in the principal's *local name space.* Each principal can create their own local names which they can use to refer to other principals. The names are arbitrarily chosen – there is no naming system to follow, and no attempt is made make names that are "globally" unique across all local names spaces. Thus some principal that Alice has named bob may be completely different from the principal that Carl calls bob.

SDSI provides a simple method to *link* local name spaces together. If Alice has named a principal Bob, and Bob has named another principal Jim, then Alice can refer to that second principal as Bob's Jim. Alice can refer to any of bob's principals in this way, and the chain can be extended indefinitely, for example, as in bob's jim's mother's doctor. Names can also be "symbolically" defined. For example, Alice's local name bob can denote company's Bob-Smith. If the principal that Alice calls company changes the principal it calls Bob-Smith, then the principal that she calls bob changes as well.

SDSI achieves this name linking because it has an online orientation. Principals that issue certificates are assumed to be able to provide an on-line Internet server to distribute those certificates upon request. Thus for Alice to find the actual principal behind the name bob's jim, she simply connects to bob's server and requests the name-binding certificate that defines the name jim.

SDSI also provides for multiple global name spaces. These are the name spaces defined by a small set of *distinguished root* principals. These principals have special reserved names (that end with !!), which are bound to the *same* principal in *every* name space. SDSI does not describe how this is achieved in any detail. However, it does give SDSI the power to access "standard" name spaces, for example VeriSign!!'s Microsoft's CEO or DNS!!'s com's microsoft's "Bill Gates". Here, the name VeriSign!! evaluates to the *same* principal in *all* name spaces. The name DNS!! also resolves to another, unique principal in all name spaces. Note that this does not mean that all principals have a single, unique global name. Rather, a principal can have multiple global names that start from different distinguished roots (as in our example).

3.3.4 SDSI Groups

SDSI allows its principals to define groups, or sets, of principals. Each group has a name and a set of members. The name is local to some principal, which is the "owner" of the group. Only a group's owner may change its definition. A group can be an explicit list of the group's members (either as a list of principals and/or names of principals), or it can be defined in terms of other groups. Any principal can define his own groups and export them via his servers in much the same way as name bindings. The servers can issue membership certificates based on the groups' definitions.

Groups provide the fundamental mechanism by which SDSI operates. When defining a security policy (for example, specifying who is allowed access to a particular resource), SDSI allows you to define the group of authorized principals, then place the group's name on the resource's access-control list(s). SDSI's naming system allows a person to easily understand security policies created in this way.

3.3.5 SDSI in Action

To better illustrate SDSI's ideas, we now provide a small example of how SDSI would operate in a typical situation. SDSI defines *protocols* in which *messages* are exchanged. Our example, illustrated in Fig. 3.10, shows how the SDSI Membership and Get protocols are used to access an FTP server.

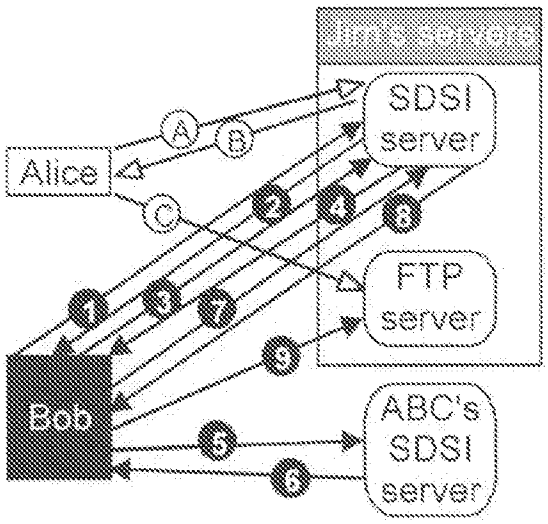


Fig. 3.11 SDSI protocol example

The FTP server is administered by Jim, an employee of ABC Inc. Jim wants to give FTP access to his friends and to other ABC employees. Jim defines a group called ftp-users on his SDSI server. That group contains two entries, the groups named friends and abc's employees, meaning that for a principal to be a member of the ftp-users group it must either be a member of friends or a member of abc's employees (or both). Jim has also defined a group he calls friends on his server, which contains the names alice, stanley and Bob, corresponding to the principals of Jim's friends. Furthermore, Jim has bound the name abc to ABC Inc.'s principal. Finally, ABC Inc. has created a group it calls employees on its SDSI server, which lists all the principals of its employees, including one that they have named BobSmith103456, which is Bob's principal. These group definitions are shown in Fig. 3.11.

We begin our example by illustrating how Jim's friend Alice gains FTP access, then follow with the more complicated example of how Bob gains the same access. The messages sent and received by Alice are depicted in Fig. 3.10 with white-headed arrows, while those involving Bob are shown with black-headed arrows.

To gain access to the FTP server, Alice must show that she is a member of Jim's ftp-users group. She sends a SDSI Membership.Query message (arrow A in Fig. 3.10) to Jim's SDSI server, in which she specifies her principal and the group name ftp-users. The message is a request for a certificate stating the membership status of the given principal for the given group. That status may be one of true (i.e., the principal is a member), false (is not a member) or fail (may or may not be a member, additional credentials are needed for a full determination).

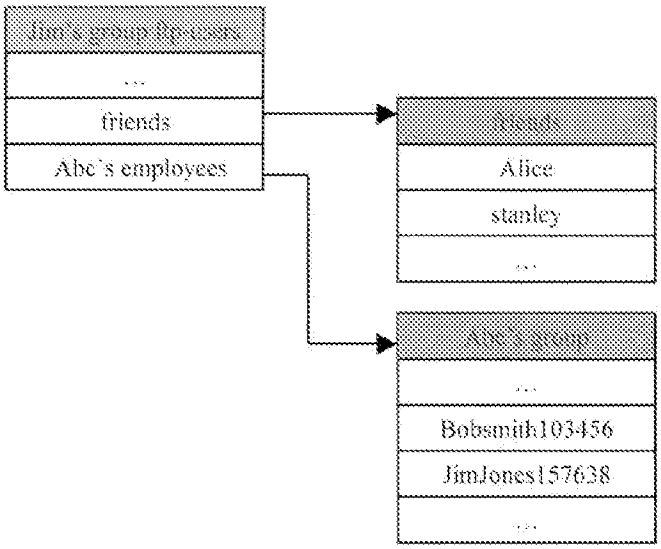


Fig. 3.12 Sample SDSI groups

In Alice's case when Jim's SDSI server performs the membership check it finds that the principal that Jim has named alice matches the principal in the Membership.Query message and is a member of Jim's friends group, which satisfies the membership requirements for the ftp-users group. Jim's SDSI server replies to Alice's query with a true membership certificate for Alice's principal (arrow B). Alice then presents the membership certificate to Jim's FTP server (arrow C) to gain access.

Bob's case is a bit more complicated. Bob is an employee of ABC Inc. but his principal is not a member of Jim's friends group. When Bob sends a Membership.Query to Jim's SDSI server (arrow 1), the reply (arrow 2) is a fail membership certificate along with an indication that if Bob can show membership (or non-membership) in Jim's abc's employees group it would help in determining his membership in the ftp-users group.

Bob needs to find out which principal Jim has named abc, so he sends a SDSI Get protocol Get.Query message to Jim's SDSI server (arrow 3). The Get protocol is used to retrieve certificates from a server. In this case, Bob requests all of Jim's name-binding certificates that specify the local name abc. Jim's SDSI server replies with a certificate showing that Jim's local name abc corresponds to ABC Inc.'s principal (arrow 4).

Bob now contacts ABC's SDSI server with a Membership.Query message for the employees group (arrow 5). ABC's SDSI server finds that Bob's principal is a member of the group, and returns a true membership certificate (arrow 6). Now Bob can send another ftp-users Membership.Query message (arrow 7) to Jim's SDSI server, this time including the membership certificate he obtained form ABC's SDSI server. Using this new credential, Jim's SDSI server can verify that Bob is a member of the ftp-users group and return a true membership certificate (arrow 8) which Bob can present to the FTP server to gain access (arrow 9).

3.3.6 The Simple Public Key Infrastructure

At the beginning of 1996, just before the publication of the SDSI paper, an Internet working group was formed to propose an alternative PKI to the X.509v3-based PKIX. This new group is called the simple public key infrastructure (SPKI) Working Group. So far, the group has only published a requirements statement, [3.59], and a draft certificate format, [3.58].

There are several similarities between the SPKI and SDSI. In particular, one of the SPKI's requirements is to support, where possible, the SDSI local name space mechanism. SDSI is, and the SPKI will be, key-centric (SDSI speaks of principals" while the SPKI uses the term keyholders"), and both provide a mechanism for attaching credentials (the SPKI calls them attributes) to public key values (SDSI through its groups, the SPKI by issuing certificates).

Although the SPKI will use SDSI names, it considers global naming schemes to be irrelevant. To quote the SPKI requirements document: "A user of a certificate needs to know whether a given keyholder has been granted some attribute and that attribute rarely involves a name." The SPKI recognizes the need to uniquely identify keyholders, and considers the public key value itself (or its hash) adequate for that purpose.

The SPKI will be a credential-based system. Its certificates will carry the minimum attributes necessary to get a job done. This is to protect, as much as possible, the privacy of keyholders. Using monolithic certificates that contain many attributes, most of which are irrelevant in a given situation, would reveal more information about the keyholder than he might like. Also, to discourage keyholders from sharing their private key values, the SPKI will allow a certificate holder to delegate the attributes she acquires from the certificate. Finally, SPKI certificates are to have several validation and revocation mechanisms: validity periods, periodic reconfirmation, CRLs, or some user-defined conditions to be tested online or through other certificates.

3.4 Summary

This chapter introduces the basic concept of PKI. There are several international PKI standards. Among them, X.509 has been widely used around the world. Many companies have released their PKI products for various applications, especially for e-business. Much recent work has focused on moving away from identity-based PKIs, such as X.509, to a more general system based on attributes or credentials. There are two main proposals for this kind of system: the simple distributed security infrastructure (SDSI), and the simple public key infrastructure (SPKI). These two proposals will play an important role in some special enterprise applications.

3.5 References

[3.1] M. Blaze, J. Feigenbaum, J. Lacy (1996) Decentralized trust management. In: Proceedings of the IEEE Conference on Security and Privacy.

[3.2] Data Encryption Standard (1993) Federal Information Processing Standards Publication 46-2.

[3.3] W. Diffie, M. E. Hellman (1976) New directions in cryptography. IEEE Trans Infor Theory 22: 644–654.

[3.4] C. Liu, P. Albitz (1992) DNS and BIND. O'Reilly, Beijing Cambridge Farnham Koln Tokyo.

[3.5] T. ElGamal (1985) A public key cryptosystem and a signature scheme based on discrete logarithms. IEEE Trans Infor Theory 31: 469–472.

[3.6] C. M. Ellison (1996) Generalized certificates.

[3.7] C. H. Fancher (1996) Smart cards. Scientific American 275(2): 40–45.

[3.8] W. Ford (1995) Advances in public-key certificate standards. ACM SIGSAC Security Audit & Control Review 13(3).

[3.9] W. Ford (1995) A public key infrastructure for US government unclassified but sensitive applications. Produced by Nortel and BNR for NIST.

[3.10] W. Ford, M. Baum (1997) Secure electronic commerce: building the infrastructure for digital signatures and encryption. Prentice-Hall, Englewood Cliffs New York.

[3.11] M. Froomkin: The essential role of trusted third parties in electronic commerce.

[3.12] Government of Canada (1995) The challenge of the Information Highway: Final Report of the Information Highway Advisory Council.

[3.13] M. Branchaud (1997) A survey of public-key infrastructures. Master's degree thesis, McGill University.

[3.14] N. McBurnett: PGP Web of trust statistics.

[3.15] N. Negroponte (1995) Being digital. Alfred A. Knopf, New York.

[3.16] Public Key Infrastructure Study Final Report (1994) Produced by the MITRE Corporation for NIST.

[3.17] W. Polk (ed.) (1996) Federal public key infrastructure (PKI) technical specifications (version 1) Part A: requirements. NIST PKI Technical Working Group.

[3.18] N. Nazareno (ed.) (1996) Federal public key infrastructure (PKI) technical specifications (version 1) Part B: technical security policy. NIST PKI Technical Working Group.

[3.19] W. Burr (ed.) (1995) Federal public key infrastructure (PKI) technical specifications (version 1) Part C: concept of operations. NIST PKI Technical Working Group.

[3.20] Federal public key infrastructure (PKI) technical specifications (version 1) Part D: interoperability profiles (1995) Produced by CygnaCom Solutions, Inc. for the NIST PKI Technical Working Group.

[3.21] D. Trcek, B. J. Blazic (1995) Certification infrastructure reference procedures. NIST PKI Technical Working Group (W. Burr, ed), NISTIR 5788, NIST.

[3.22] M. S. Baum (1994) Certification authority liability and policy. NIST-GCR-94-654, NTIS Doc. No. PB94-191-202. National Technical Information Service, Springfield, VA.

[3.23] B. S. Jr. Kaliski (1993) An overview of the PKCS standards. RSA Laboratories.

[3.24] RSA Laboratories (1993) PKCS #10: certification request standard.

[3.25] RSA Laboratories (1993) PKCS #6: extended-certificate syntax standard.

[3.26] RSA Laboratories (1993) PKCS #9: selected attribute types.

[3.27] R. Housley, W. Ford, D. Solo: Internet public key infrastructure, Part I: X.509 certificate and CRL profile (draft). IETF X.509 PKI (PKIX) Working Group.

[3.28] S. Farrell, C. Adams, W. Ford: Internet public key infrastructure, Part III: certificate management protocols (draft). IETF X.509 PKI (PKIX) Working Group.

[3.29] M. Stahl (1987) Domain administrators guide. RFC1032.

[3.30] M. Lottor (1987) Domain administrators operations guide. RFC1033.

[3.31] P. Mockapteris (1987) Domain names – concepts and facilities. RFC1034.

[3.32] P. Mockapteris (1987) Domain names – implementation and specification. RFC1035.

[3.33] R. Rivest (1992) The MD5 message-digest algorithm. RFC1321.

[3.34] J. Linn (1993) Privacy enhancement for Internet electronic mail, Part I: message encryption and authentication procedures. RFC1421.

[3.35] S. Kent (1993) Privacy enhancement for Internet electronic mail, Part II: certificate-based key management. RFC1422.

[3.36] D. Balenson (1993) Privacy enhancement for Internet electronic mail, Part III: algorithms, modes, and identifiers. RFC1423.

[3.37] B. Kaliski (1993): Privacy enhancement for Internet electronic mail, Part IV: key certification and related services. RFC1424.

[3.38] J. Kohl, B. C. Neuman (1993) The Kerberos network authentication service (version 5). RFC1510.

[3.39] C. Malamud, M. Rose (1993) Principles of operation for the TPC.INT subdomain: general principles and policy. RFC1530.

[3.40] R. Atkinson (1995) Security architecture for the Internet protocol. RFC1825.

[3.41] J. Galvin, S. Murphy, S. Crocker, N. Freed (1995) Security multiparts for MIME: multipart/signed and multipart/encrypted. RFC1847.

[3.42] D. Eastlake, C. Kaufman (1997) Domain name system security extensions. RFC2065.

[3.43] R. Housley, W. Ford, W. Polk, D. Solo (1999) Internet X.509 public key infrastructure certificate and CRL profile. RFC2459.

[3.44] C. Adams, S. Farrell (1999) Internet X.509 public key infrastructure certificate management protocols. RFC2510.

[3.45] M. Myers, C. Adams, D. Solo, D. Kemp (1999) Internet X.509 certificate request message format. RFC2511.

[3.46] S. Chokhani, W. Ford (1999) Internet X.509 public key infrastructure certificate policy and certification practices framework. RFC2527.

[3.47] R. Housley, W. Polk (1999) Internet X.509 public key infrastructure representation of key exchange algorithm (KEA) keys in Internet X.509 public key infrastructure certificates. RFC2528.

[3.48] M. Myers, R. Ankney, A. Malpani, S. Galperin, C. Adams (1999) X.509 Internet public key infrastructure online certificate status protocol – OCSP. RFC2560.

[3.49] R. Housley, P. Hoffman (1999) Internet X.509 public key infrastructure operational protocols: FTP and HTTP. RFC2585.

[3.50] S. Boeyen, T. Howes, P. Richard (1999) Internet X.509 public key infrastructure LDAPv2 schema. RFC2587.

[3.51] R. Rivest, A. Shamir, L. Adleman (1978) A method for obtaining digital signatures and public key cryptosystems. Commun ACM 21: 120-126.

[3.52] M. Saeki (1996) Elliptic curve cryptosystems. M.Sc. thesis, McGill University.

[3.53] B. Schneier (1996) Applied cryptography (2nd ed). Wiley, New York.

[3.54] R. Rivest, B. Lampson (1996) SDSI – A simple distributed security infrastructure.

[3.55] MasterCard and Visa (1996) Secure electronic transaction (SET) specifications.

[3.56] Secure Hash Standard (1995) Federal information processing standards publication 180-1.

[3.57] D. R. Stinson (1995) Cryptography: theory and practice. CRC Press, Boca Raton New York.

[3.58] C. M. Ellison, B. Frantz, B. M. Thomas (1996) Simple Public Key Certificate.

[3.59] C. M. Ellison (1997) SPKI requirements.

[3.60] D. Trcek, B. J. Blazic, N. Pavesic (1996) Security policy space definition and structuring. Computer Standards & Interfaces 18(2): 191–195.

[3.61] D. Trcek, T. Klobucar, B. J. Blazic, F. Bracun (1994) CA-browsing system – A supporting application for global security services. In: ISOC Symposium on Network and Distributed System Security, pp. 123–128.

[3.62] ITU/ISO (1988) Recommendation X.208. Specification of abstract syntax notation one (ASN.1).

[3.63] ITU/ISO (1998) Recommendation X.209. Specification of basic encoding rules for abstract syntax notation one (ASN.1).

[3.64] ITU/ISO (1993) Recommendation X.500. Information technology – open systems interconnection – the directory: overview of concepts, models, and services.

[3.65] ITU/ISO (1993) Recommendation X.509. Information technology – open systems interconnection – the directory: authentication framework.

[3.66] ITU/ISO (1996) Final text of draft amendments to X.500/9594 for certificate extensions.

[3.67] P. Zimmermann: PGP user's guide vol. 1 and 2.

4 Biometrics for Security in E-Commerce

David Zhang[1] and Li Yu[2]

[1] Department of Computing
Hong Kong Polytechnic University, Hong Kong

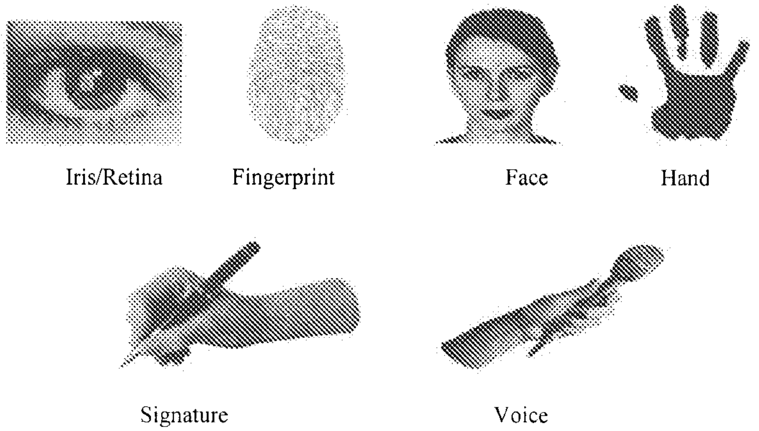
[2] Department of Computer Science and Technology
Harbin Institute of Technology, China

4.1 An Overview of Biometrics

The advance of technology is always inspired by the practical applications, and the emergence of automatic biometrics technology is rooted in the requirement for real-world security applications. Whether this new technology can last for a long time will be decided by how well it can solve security problems. Although biometric technology is at the development stage, it has been implemented in various applications and some of them work well. Along with the widespread application of biometrics technology, more funds and more attention are being given to this ascending technology [4.1-4.4, 4.20-4.22, 4.24, 4.32, 4.34].

Biometrics refers to the automatic identification of a person based on his/her physiological or behavioral characteristics [4.19]. Today a variety of biometric technologies are used, each with its own strengths to make it more appropriate than others for certain types of applications. Fig. 4.1 shows the major biometric technologies:

- Finger-scan
- Hand-scan, aka, hand geometry
- Retina-scan
- Iris-scan
- Facial-scan, aka, facial geometry
- Signature-scan, aka, dynamic signature verification
- Voice-scan, aka, voice or speaker verification



Iris/Retina Fingerprint Face Hand

Signature Voice

Fig. 4.1 Biometrics technologies

Biometric is the most secure and convenient authentication tool. It cannot be borrowed, stolen, or forgotten, and forging one is practically impossible. Common physical biometrics includes fingerprints, hand or palm geometry, retina, iris, and facial characteristics. Behavioral characteristics include a person's signature, voice (which also has a physical component), keystroke pattern, and gait. Technologies for signature and voice are the most developed for the behavioral biometrics [4.30-4.33, 4.35].

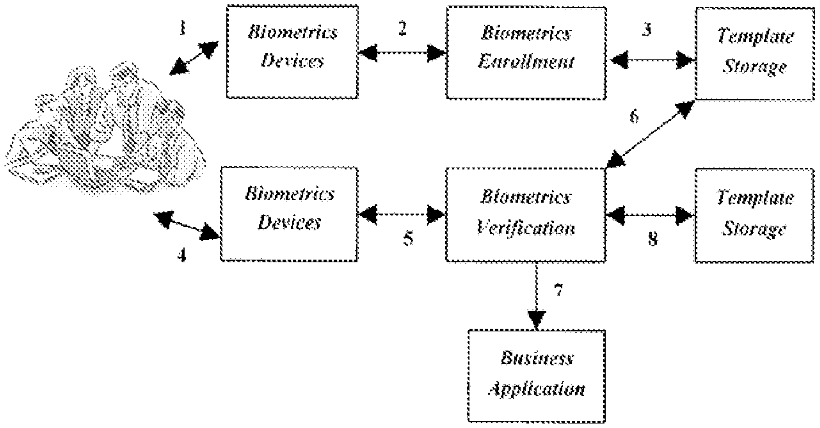


Fig. 4.2 How a biometric system works

In Fig. 4.2 the process involved in using a biometric system is described. The text descriptions of these processes are as follows:

(1) Capture the chosen biometric data.
(2) Process the biometric data, extract it and enroll it to form a biometric template.
(3) Store the template in a local repository, a central repository, or a portable token such as a smart card.
(4) Live-scan the chosen biometric.
(5) Process the biometric data and extract features to form a biometric template.
(6) Match the new template against stored templates.
(7) Calculate a matching score for the business application.
(8) Record a secure audit trail with respect to system use.

4.1.1 Finger-Scan Technology

Finger-scan biometrics is based on the distinctive characteristics of the human fingerprint. A fingerprint image is read from a capture device, features are extracted from the image, and a template is created. If appropriate precautions are followed, the result is a very accurate means of authentication [4.2-4.4, 4.6-4.8, 4.23, 4.27].

Fingerprint matching techniques can be placed into two categories: minutiae-based and correlation-based [4.25]. Minutiae-based techniques first find minutiae points and then map their relative placement on the finger. However, there are some difficulties for this approach when the fingerprint is of such low quality such that accurate extraction of minutiae points is difficult. Also, this method does not take into account the global pattern of ridges and furrows. In contrast, the correlation-based method is able to overcome the problems of the minutiae-based approach. However, correlation-based techniques require the precise location of a registration point and are affected by image translation and rotation.

The most widely used methods of controlling access to computers and data are passwords and PINs. While passwords and PINs are easy to use, they provide weak proof of identity. They are rarely changed, frequently shared, often used in plain sight, and easily defeated using widely available hacker programs. Implementing fingerprint authentication and replacing passwords and PINs makes access to corporate information more efficient and secure. Fingerprint verification may be a good choice for in-house systems that operate in a controlled environment where you can give users adequate explanation and training. A fingerprint authentication solution can provide secure online banking transactions, secure customer financial information, new online services, and non-repudiation. The benefits include fraud protection, customer confidence and retention, time/cost efficiencies, and the ability to extend services to non-local customers.

4.1.2 Hand-Scan Technology

This approach uses the geometric shape of the hand for authentication. Authentication of identity using hand geometry is challenging work, as hand features are not very descriptive. The problem can be tackled by combining various individual features to attain robust verification [4.2-4.4].

Hand-scan is occasionally misunderstood as "palm reading," as the placement of the hand palm-down on the reader can be confusing to those unfamiliar with the technology.

Hand-scan is a relatively accurate technology, but does not draw on as rich a data set as finger-, face-, or iris-scans. A decent measure of the distinctiveness of a biometric technology is its ability to perform one-to-many searches, that is, the ability to identify a user without the user first claiming an identity. Hand-scan does not perform one-to-many identification very well, as similarities between hands are not uncommon. It has an advantage in failure-to-enroll (FTE) rates, which measures the likelihood that a user is incapable of enrolling in the system. In contrast, finger-scan is prone to FTEs in the case of poor-quality fingerprints, and facial-scan requires consistent lighting to properly enroll a user. Since nearly all users will have the dexterity to use hand-scan technology, only a few employees and visitors will need to be processed outside the biometric.

Organizations use hand-geometry readers in various scenarios, including time and attendance recording, where they have proved extremely popular. Hand-scan technology offers the following benefits:

(1) Cards and associated administration costs can be eliminated.
(2) "Buddy-punching" is impossible; this is particularly rated by leading time and attendance software systems.
(3) True-time clock functionality including department transfers, supervisor override, and time restrictions.

Various applications, such as credit card and ATM transactions, check cashing, and even picking up a child from daycare, benefit from this technology.

The benefits of hand-geometry scanning in personal identification are:

(1) High user acceptance, non-intrusive technology.
(2) Fast and easy enrollment use.
(3) Low false reject rate equates to a positive user experience.

Ease of integration into other systems and processes, coupled with ease of use, makes hand-geometry scanning an obvious first step for many biometric projects.

4.1.3 Retina-Scan Technology

Along with iris recognition technology, retina scan is perhaps the most accurate and reliable biometric technology. However, it is difficult to use and is perceived as being moderately to highly intrusive. In films, portrayals of retina-scan devices reading at an arm's length, with a non-stationary subject, are false. Retina-scan biometrics requires a cooperative, well-trained, patient audience, or else performance will fall dramatically [4.2, 4.4-4.5].

Even when those unfamiliar with the rudimentary anatomy of the eye are reminded that all vision is based upon light passing through the pupil to the retina, there is still notable resistance to retina-scan technology. This is perhaps due to an unusually high degree of sensitivity on issues of the eye; iris-scan biometrics, where the patterns of the iris are read, which requires less effort on the part of the user, is also frequently met with similar expressions of hesitation.

Retina-scan devices read through the pupil; this requires the user to situate their eye within 1/2 inch of the capture device, and to hold still while the reader ascertains the patterns. The user looks at a rotating green light as the patterns of the retina are measured at over 400 points. By comparison, a fingerprint may only provide 30-40 distinctive points (minutiae) to be used in the enrollment, template creation, and verification process. Retina-scanning has a very high level of accuracy compared to most other biometrics.

Retina-scanning can be quite accurate but requires the user to look into a receptacle and focus on a given point. This is not particularly convenient if you wear glasses or are concerned about having close contact with the reading device. For these reasons, retina-scanning is not warmly accepted by all users, even though the technology itself works well.

4.1.4 Iris-Scan Technology

Iris recognition uses the unique features of the human iris to provide an unmatched identification technology. So accurate are the algorithms used in iris recognition that the entire planet could be enrolled in an iris database with only a small chance of false acceptance or false rejection [4.2, 4.4-4.5].

Iris identification technology is a tremendously accurate biometric. Only retina-scan can offer nearly the security that iris-scan offers, and the interface for retina-scan is thought by many to be more challenging and intrusive. More common biometrics provide reasonably accurate results in verification schematics, whereby the biometric verifies a claimed identity, but they cannot be used in large-scale identification implementations like iris recognition.

An iris-based biometric, on the other hand, involves analyzing features found in the colored ring of tissue that surrounds the pupil. Iris-scan, undoubtedly the less intrusive of the eye-related biometrics, uses a fairly conventional camera element and requires no close contact between the user and the reader. In addition, it has the potential for a higher than average template-matching performance. Iris biometrics work with glasses in place and it is one of the few technologies that can work well in identification mode. Ease of use and system integration have not traditionally been strong points with iris-scanning devices, but you can expect improvements in these areas as new products emerge.

In some banks, new ATMs with iris-recognition technology have been used to control access to bank accounts. After enrolling once (a "30 second" process), the customer needs only to approach the ATM, follow the instruction to look at the camera, and be recognized within 2-4 seconds [4.44].

Again, the benefits of such a system are clear, that is, the customer who chooses to use an ATM with iris recognition will have a quicker, more secure transaction. Although one may question whether the risk of fraud at ATM's is very large, this type of integration makes long-term sense. First, ATMs are large, omnipresent, secured devices which, the public knows, visually record every transaction. It is a small cognitive leap to envision them moving from their current configuration to being biometrically enabled. Second, iris technology is being put before the public in a non-coercive, unobtrusive, fairly low-risk setting. As the accuracy and reliability of the technology are proven through time, the public should be accepting of less traditional implementations. Like the vast majority of biometric companies, iris recognition vendors are very eager to participate in securing Internet commerce. The potential market for the vendors whose technology is most widely embraced is unimaginably large.

4.1.5 Facial-Scan Technology

Similar to finger-scan and voice-scan biometrics, there are various methods by which facial-scan technology recognizes people. All share certain commonalities, such as emphasizing those sections of the face that are less susceptible to alteration, including the upper outlines of the eye sockets, the areas surrounding the cheekbones, and the sides of the mouth. Most technologies are resistant to moderate changes in hairstyle, as they do not utilize areas of the face located near the hairline. All of the primary technologies are designed to be robust enough to conduct one-to-many searches, that is, to locate a single face out of a database of thousands, and even hundreds of thousands, of faces [4.2, 4.4-4.5, 4.27].

Facial-scans can be used to control entry to buildings or computer networks by comparing the image of a person seeking access against the scan taken of that person at an earlier date, that is, a one-to-one check [4.47].

Facial-recognition solutions employ the same four-step process that all biometric technologies do, namely, sample capture, feature extraction, template comparison, and matching. The sample capture takes place in the enrollment process, during which the system takes multiple pictures of the face, usually from slightly different angles, to increase the system's ability to recognize the face. After enrollment, certain facial features are extracted and used to create a template. The specific features extracted vary depending on the type of facial-recognition technology used. No images of faces are stored. Instead, the templates consist of numeric codes that are usually encrypted. Many templates can be stored on one system because each is less than 1K in size, compared to between 150K and 300K for a facial image. When someone logs in using a facial-scan system, the template created upon attempted login is compared to a stored template for that person (one-to-one matching) or to a database of stored templates (one-to-many matching).

Face recognition involves the analysis of facial characteristics. This technique has attracted considerable interest, although many people do not completely understand its capabilities. Some vendors have made extravagant claims, which are very difficult, if not impossible, to substantiate in practice for facial-recognition devices. Because facial-scanning needs an extra peripheral not customarily included with basic PCs, it is more of a niche market for network authentication. However, the casino industry has capitalized on this technology to create a facial database of scam artists for quick detection by security personnel.

4.1.6 Handwriting and Signature Verification

Signature verification involves analysis of the way a user signs their name. Signing features, such as speed, velocity, and pressure, are as important as the finished signature's static shape. Signature verification enjoys a synergy with existing processes that other biometrics do not. People are used to signatures as a means of transaction-related identity verification, and most would see nothing unusual in extending this to encompass biometrics. Signature-verification devices are reasonably accurate in operation and obviously lend themselves to applications where a signature is an accepted identifier.

Electronic-signature verification is also gaining ground for retail and e-commerce applications. The implementation includes installation of electronic-signature software and the solution utilizes electronic signatures to automate processing of some lease-end documents. The solution is part of a transition from paper lease-end documents to electronic documents, which will reduce operation costs and provide an easy-to-use, legally binding electronic signature. E-Pad [4.45] captures a handwritten signature and converts it to a biometric e-signature, offering enhanced workflow and faster processing times. Electronic signatures may be bound into Microsoft Word and Outlook documents, Adobe Acrobat files, and many other forms and transactions. They also feature a handwriting profile that can be used to authenticate the identity of the signer. Surprisingly, relatively few significant signature applications have emerged compared with other biometrics methodologies. But if your application fits, it is a technology worth considering [4.2, 4.4-4.5].

4.1.7 Voice-Scan Technology

Of all the above-mentioned human traits used in biometrics, the one that humans learn to recognize first is the voice characteristic. Infants can identify the voice of their mothers and telephone users can identify a caller on a noisy telephone line. Furthermore, the bandwidth associated with speech is much smaller than the other image-based human traits. This implies quicker processing and smaller storage space [4.2, 4.4-4.5, 4.26, 4.29].

A speaker-recognition system can be divided into two categories, namely, text-dependent and text-independent systems. In text-dependent systems, the user is expected to use the same text (keyword or sentence) during training and recognition sessions. A text-independent system does not use the training text during recognition session. Both systems perform the following tasks: feature extraction, similarity analysis, and selection. Texture extraction uses the spectral envelope to adjust a set of coefficients in a predictive system. One voice sample can then be compared for similarity with another sample by computing the regression between the coefficients. This is a similarity analysis. A number of normalization techniques have been developed to account for variation of the speech signals.

A voice security system is responsible for an innovative method of security that dramatically reduces fraud and can prevent one's property from being use, if stolen or obtained fraudulently. This new breakthrough allows speaker-verification to be burned onto an existing microprocessor within a device. Examples of use of this technology are cell phones (to eliminate cell phone fraud), ATMs (to eliminate PIN fraud), and automobiles (to dramatically reduce theft and carjacking). This method is the only standalone technology that does not require management of a large user database, thus protecting the privacy of the user's biometric data. The software, algorithms and templates can be stored on the microprocessor that a device already employs to operate the functions of the electronic hardware inside [4.42].

Voice authentication is not based on voice recognition but on voice-to-print authentication, where complex technology transforms voice into text. Voice biometrics has the most potential for growth, because it does not require new hardware, that is, most PCs nowadays already come with a microphone. However, poor voice quality and ambient noise can affect verification. In addition, the enrollment procedure has often been more complicated than with other biometrics, leading to the perception that voice verification is not user friendly. Therefore, voice authentication software needs improvement. One day, voice may be integrated with finger-scan technology. Because many people see finger-scanning as a higher form of authentication, voice biometrics will most likely be relegated to replace or enhance PINs, passwords, or account names.

4.2 Potential Application Areas

Biometrics applications are not limited to the areas mentioned in the last section. In fact, as long as a system needs to recognize people, it can incorporate biometrics. In the law enforcement community, matching finger images or part of palm images is the most common method to process criminal suspects and bring guilty criminals to justice. Also we have seen many times in movies the police ask the witness to describe the criminal's physical features such as hair color, the length, width, and shape of the face, etc.; and they then reconstruct a picture of the criminal. In some movies, we see the criminal call the victim over the phone and the police can record the voice of the criminal and search for the criminal according to the voice. All these scenes are examples of finding people using their unique physical features (finger, palm, face, etc.) or behavioral trait (voice), and automatic biometrics can help in all these examples. It is not difficult to understand that the law enforcement community is the largest biometrics user group. Police forces throughout the world are using the Automatic Fingerprint Identification System to assist in crime detection. There are many biometrics vendors earning significant revenues in this area [4.9, 4.11-4.12, 4.14-4.15].

Businessmen always play an important role in spreading a new technology. As automatic biometrics technologies become more and more mature in the law enforcement area, they are also introduced in civilian applications by biometrics product vendors. Usually most civilian biometrics applications are some kind of access control. We may simply classify all the civilian biometrics applications as either physical access control or data access control. Physical access control ensures only authorized individuals can physically access secure areas while data access control secures access to sensitive data. Securing benefit systems against fraud, preventing illegal immigrants from entering a country, or prisoners from leaving a prison all belong to physical access control, while Internet banking, telephone banking, ATM, and Web Store belong to data access control. Automatic biometrics is a rapidly expanding market. Fraud is an ever-increasing problem and security is becoming a necessity in many walks of life. Civilian access control, therefore, will not be restricted to the application areas mentioned below and will branch out to other market opportunities as soon as a need is identified.

4.2.1 Benefit Systems

When applying biometrics in benefit systems, it plays a different role from that in banking or in physical access control. In banking or physical access control or other methods of access control, all unique physical features or behavioral traits are registered in a system before the system is used. When somebody needs to access the system, their unique feature is captured at that point and the system checks the newly captured feature against the template in the database to decide whether it is from the same person. This will prevent unauthorized people from accessing the system. In benefit systems, however, people do not need to register their features first. Only when they need to get the benefit is their unique feature extracted and stored in the system. The system checks whether this person has already registered and received the benefit before, by checking the database templates. Biometrics is well placed to capitalize on this phenomenal market opportunity, and vendors are building on the strong relationship currently enjoyed with the benefits community.

4.2.2 E-Commerce Applications

E-commerce developers are exploring the use of biometrics and smart cards to more accurately verify a trading party's identity. For example, many banks are interested in this combination to better authenticate customers and ensure non-repudiation of online banking, trading, and purchasing transactions [4.5, 4.9, 4.13].

Banks may embrace biometrics technologies from various aspects. Automated teller machines (ATMs) and transactions at the point of sale, telephone banking, Internet banking, and many other banking applications are vulnerable to fraud and can be secured by biometrics. Fig. 4.3 shows various biometrics and banking services and where they can be applied.

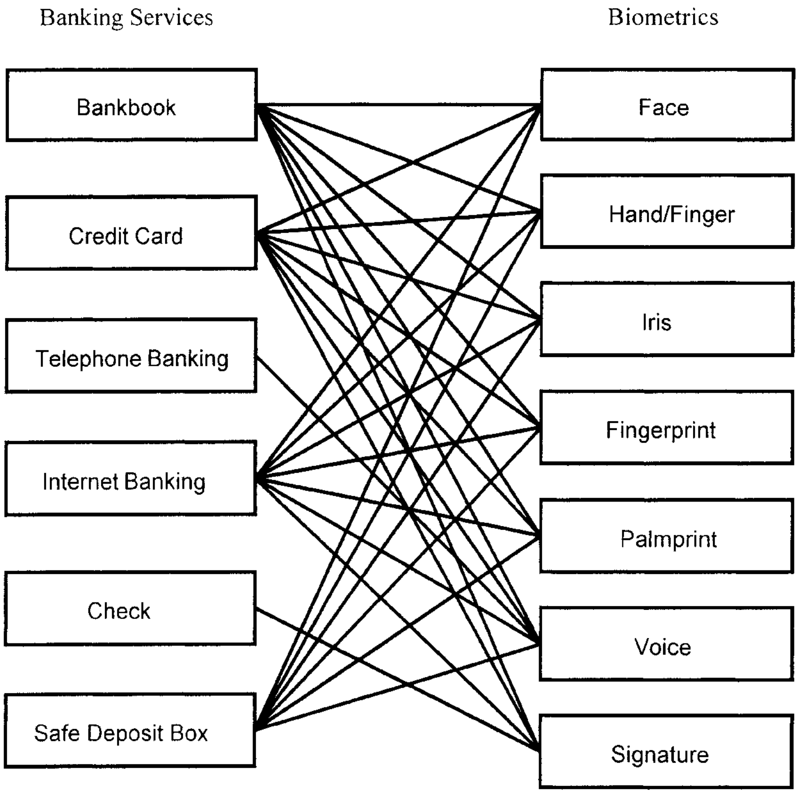


Fig. 4.3 Biometrics applications in banking

Point-of-sale (POS) system vendors are working on the cardholder verification method, which would enlist smart cards and biometrics to replace signature verification. MasterCard estimates that adding smart-card-based biometric authentication to a POS credit card payment will decrease fraud by 80%.

Merchants use biometrics to obtain secure services over the telephone through voice authentication. Voice authentication systems developed by Nuance Communications are currently deployed nationwide, by both the Home Shopping Network and Charles Schwab. The latter's marketing catch phrase is: "No PIN to remember, no PIN to forget."

4.2.3 Computer Systems

Currently, computer systems use passwords as their secure guards. On one hand, remembering tens of passwords and changing passwords very often becomes a headache for almost everyone who uses a computer; on the other hand, a password itself does not have direct connection to the end-user. If somebody gets the password, they will be considered as a legal user by the computer system even though he is a criminal and meanwhile, if a legal user forgets their password, they will be refused access to their own computer. Biometrics technology binds the authority directly to the end-user and removes the need for various passwords. Voice and fingerprint recognition are now the most promising techniques in this area.

4.2.4 Immigration

Terrorism, drug-running, illegal immigration, and an increasing throughput of legitimate travelers are putting a strain on immigration authorities throughout the world. It is essential that these authorities can quickly and automatically process law-abiding travelers and identify the lawbreakers. Biometrics is being employed in a number of diverse applications to make this possible. The US Immigration and Naturalization Service is a major user and evaluator of a number of biometrics. Systems are currently in place throughout the US to automate the flow of legitimate travelers and deter illegal immigrants. Elsewhere biometrics is capturing the imagination of countries such as Australia, Bermuda, Germany, Malaysia, and Taiwan.

4.2.5 National Identity

Biometrics is beginning to assist governments as they record population growth, identify citizens, and prevent fraud occurring during local and national elections. Often this involves storing a biometrics template on a card that in turn acts as a national identity document. Finger scanning is particularly strong in this area and schemes are already under way in Jamaica, Lebanon, The Philippines, and South Africa.

4.2.6 Telephone Systems

Global communication has truly opened up over the past decade, while telephone companies are under attack from fraud. Once again, biometrics is being called upon to defend this onslaught. Speaker ID is a technique for recognizing people by their voices. It is obviously well suited to the telephone environment and is catching these new markets quickly.

4.2.7 Monitoring Time and Attendance

Currently, some factories and companies use cards to monitor the movement of their employees. When they come to work, they need to punch a hole on their cards and another hole when they leave. Such things can be assisted by biometrics. With a biometrics system, employees may press their fingers on a small platform when they come or leave. This may prevent some forms of cheating. But using such a system to monitor employees' movement is still in question because some people think it may violate an employee's privacy.

4.2. 8 Covert Surveillance

One of the more challenging research areas involves using biometrics for covert surveillance. Using facial and body recognition technologies, researchers hope to use biometrics to automatically identify known suspects entering buildings or traversing crowded security areas such as airports. The use of biometrics for covert identification as opposed to authentication must overcome technical challenges such as simultaneously identifying multiple subjects in a crowd and working with uncooperative subjects. In these situations, devices cannot count on consistency in pose, viewing angle, or distance from the detector.

4.3 Multiple Authentication Technologies

From an application standpoint, widespread deployment of a user authentication solution requires support for an enterprise's heterogeneous environment. Support for legacy applications, client-server applications, and Web-based applications is extremely important. However, the complexity of this process is exacerbated by the number of application-specific identities one has to manage. This is compounded by the fact that individual authentication devices or methods may be required to maintain additional identities [4.2, 4.5, 4.11, 4.16].

When it comes to selecting and deploying specific verification methods for enterprise-wide use, one size does not fit all. Any solution should enable the selection of different methods, depending on the users, and be flexible enough to enable dynamic, multifactor authentication, allowing you to dial up the appropriate level of security without sacrificing convenience.

The distributed environment of an enterprise – organizationally, geographically, and technologically – means that employees access information from various access channels. For example, an employee who may access data from their desktop one day may use their laptop remotely the next day. The increasing use of PDA and other wireless devices makes it obvious that users must be authenticated regardless of the channel of access in the digital environment.

Finally, user authentication as a security infrastructure cannot be considered in isolation, often it is through a multifaceted security approach in which combinations of security solutions are deployed. An authentication solution should seamlessly extend the organization's existing security infrastructure.

Consolidating and streamlining user authentication enables the creation of an "Authentication Hub," as shown in Fig. 4.4, providing a single point of control to deploy and manage a combination of authentication methods such as passwords, smart cards, tokens, and biometrics (fingerprint, voice, face, iris, and signature recognition).

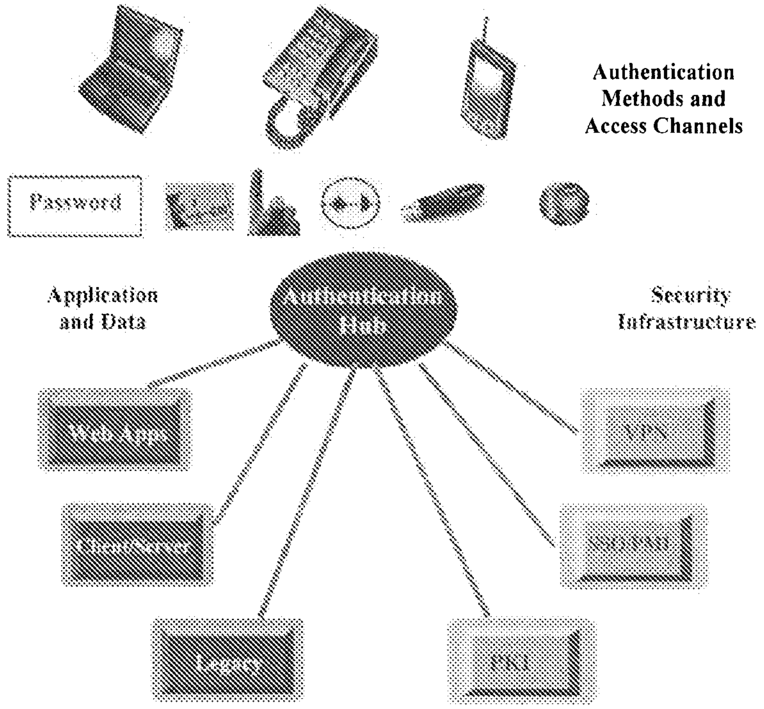


Fig. 4.4 Unified authentication management

An infrastructure that provides unified authentication management allows organizations to manage user authentication through context-based security policies and integrate authentication with existing security solutions such as virtual private networks (VPNs), access management, and public-key infrastructure (PKI). It mitigates interoperability problems between multiple applications, authentication methods, access channels, and platforms, thus driving cost savings, convenience, and security.

When implemented as a strategic component of the security infrastructure within the enterprise, the unified authentication management solution:

- Implements centralized authentication policies for a large number of users.
- Integrates the administration of many forms of authentication.
- Tailors authentication methods to specific information assets, sizes of transactions, roles of individuals and groups, and access channels and entry points.
- Provides a single point for managing user authentication for heterogeneous applications – legacy, client-server, and Web-based.
- Enables convenient but secure access to applications and data in a heterogeneous environment, allowing you to dial up the appropriate level of security without sacrificing user convenience.
- Streamlines the administration tasks necessary to enable convenient and cost effective implementation of strong authentication security policies such as enrollment, verification, policies, and unenrollment of users when multiple authentication methods (smart cards, tokens, fingerprint devices) are deployed.
- Extends existing security infrastructures – VPNs, privilege management (or SSO: single sign-on), policy making and implementation (PMI), and PKI – and seamlessly supports the adoption and migration to advanced authentication methods.

A major problem with biometrics is how and where to store the user's template. Because the template represents the user's personal character, its storage introduces privacy concerns. Furthermore, storing the template in a centralized database leaves that template subject to attack and compromise. On the other hand, storing the template on a smart card enhances individual privacy and increases protection from attack, because individual users control their own templates.

Vendors can enhance security by placing more biometric functions directly on the smart card. Some vendors have built a fingerprint sensor directly into the smart card reader, which in turn passes the biometric to the smart card for verification. At least one vendor, Biometric Associates, has designed a smart card that contains a fingerprint sensor directly on the card. This is a stronger secure architecture because cardholders must authenticate themselves directly to the card.

PKI uses public- and private-key cryptography for user identification and authentication. There are some advantages over biometrics: PKI is mathematically more secure, and it can be used across the Internet. The main drawback of PKI is the management of the user's private key. To be secure, the private key must be protected from compromise; to be useful, the private key must be portable. The solution to these problems is to store the private key on a smart card and protect it with a biometric.

In the Smart Access common government ID card program, the US General Services Administration is exploring this union of biometrics, smart cards, and PKI technology. The government of Finland is also considering using these technologies in the Finnish National Electronic ID card.

4.4 How to Select a Biometrics System

4.4.1 Difficulty in Selecting a Biometrics System

The performance of a biometrics system is greatly impacted by many factors, such as humidity, light, noise, and the end user's attitude to and familiarity with the system. Therefore, for accuracy, testing should be performed in real working circumstances. High levels of accuracy in one application do not qualify a system for an entirely different application. The quoted performance figures of a biometric system only can be applied to the specific application for which they are quoted. Each application is widely different in terms of system workload and throughput, environmental factors, and other variables [4.2, 4.5, 4.17-4.18, 4.28, 4.33, 4.36].

For example, the same fingerprint-verification system may have very high accuracy in a university restaurant while it may work badly in a village where most people have their fingerprints worn heavily.

This is not to say that the performance rulers, the fault accept rate (FAR) and the fault reject rate (FRR) are meaningless, just that the two rates may vary in diverse environments. The FAR and FRR provided by the developer can be used as a guide to understand a system's general ability.

In one sentence, a biometrics system's performance is application sensitive, and making a biometrics system adaptive to a particular application needs significant consideration.

4.4.2 Various Factors: Whether a Biometrics System Is Needed

Before applying biometrics, we must make clear what the business driver is. What is the main goal of the whole project? What are the constraints of the project, like deadlines, budgets, etc.? What security level is needed? What is the current system? What is the weakness of the current system? Is it necessary to apply biometrics? Can biometrics solve the existing problems? Are there any other choices for securing the system? Will biometrics cause any trouble for the system? Can biometrics integrate well with the current system? Will users of the system accept this new work style? Simply put, making sure biometrics is needed is the first step when applying this new technology.

4.4.3 Comparison of Different Biometrics Techniques

Tables 4.1 and 4.2 are two tables of comparison for existing biometrics systems. Factors considered in biometrics system evaluation are discussed below.

Vulnerability to Fraud

Biometrics systems aim at providing high-level security, so whether a biometrics trait is hard to mimic is an essential consideration in the construction of such applications. Extreme measures, such as gouging out eyes or truncating fingers to defraud a biometrics system, have appeared in some movies.

Ease of Use

One springboard for biometrics system popularity is to allow the public to get rid of the bother of remembering tens of passwords and keeping strings of keys. Such systems could be really user friendly, preventing headaches greater than a lost room key. Some biometric devices are not user friendly. For example, users without proper training may experience difficulty aligning their head with a device for enrolling and matching facial templates, while face, fingerprint, voice, and palm technologies are easy to use.

Intrusive for Human Beings

Certain biometrics systems are seen to be more intrusive than others. For example, retina capture involves exposing eyes to a bright beam, while voice-scan seems non-intrusive. However, sometimes, a higher accuracy may be gained using a more intrusive approach. Places where high levels of security are needed have to choose such intrusive methods. For example, workers at a nuclear power plant would probably acknowledge the need for a degree of intrusiveness, as security is a very important issue in that environment.

Applicability

Physical characteristics vary and some individuals will not be able to use a biometrics system. No single biometrics system can capture and match biometrics data for the global population in all circumstances. Human beings are as diverse and unpredictable as environments. Some individuals have damaged fingers, limbs, voice boxes, or eyes. This may make verification and identification with a single biometrics system impossible; but it may be possible to use a multiple-biometrics system. Also, this does not mean that a single biometrics system is unable to perform a task in an application where a minority of people cannot have a biometrics sample captured. It is simply that the minority cannot use the system automatically and must be dealt with in an appropriate manner.

Speed of Verification

Response time is a key issue for any computer system and for biometrics systems.

Size of Storage for One Biometrics Template

For an identification system, this factor directly affects the overall database size and searching speed. For a verification system, where the registered template is stored in some special media such as barcodes, magnetic cards, or smart cards, this factor could determine the cost of a card.

Long-Term Stability

The biometrics feature chosen to identify a person in a system should be stable for at least as long as the system is to be used, so that the system can work correctly during its lifetime.

Maturity of Technology

Some biometrics features, such as fingerprints and signatures, have been used for a long time and their accuracies have been proven widely. Meanwhile, other biometric systems, such as face-scanning and voice-scanning, are newcomers to this area and need to be proven in real-time applications.

From Tables 4.1 and 4.2, retina-scan appears to have the highest crossover accuracy. Even though iris-scan has a high cross over accuracy, its user acceptability is low. Fingerprints and hand geometries are equally "unique."

Signature dynamics and voice dynamics have the lowest accuracy rates. In addition, these two techniques rely on behavioral measurements as opposed to physical measurements. In general, behavioral biometrics is less reliable than physical biometrics.

Retina-scan has a high accuracy but also has a high data-collection-error rate and a low user-acceptability. For this reason, retina-scan broadly exists only in science fiction movies and not in real-life applications!

The fingerprint biometric has a low data-collection-error rate and a high user-acceptability. Further, fingerprint technology has been heavily invested in, and applied to both the identification and the authentication problem. Finally, fingerprint biometrics has the highest acceptance in the identification community and virtually every large biometrics system in operation today uses fingerprint biometrics. Notwithstanding its association with "criminal" applications, fingerprint biometrics is generally accepted by clients.

Table 4.1 Comparison of various biometrics techniques

Technical factor	*Hand geometry*	*Retina*	*Fingerprint*
False rejection rate	0.2 percent, one try	12.4 percent (one try), 0.4 percent (three tries)	1% - 5%, three tries
False acceptance rate	0.2 percent, one try	0 no false acceptances	0.01 - 0.0001 percent (three tries)
Vulnerability to fraud	Almost impossible to secretly obtain hand-geometry data. However, when the person cooperates, this seems not at all impossible	No counterfeits seem possible. False eyes, contact lenses, or eye transplants cannot breach the security of this device	Dummy fingers and dead fingers will be detected when high-security platen is installed
Ease of use	The first time one needs to get used to it. After some experience it is not difficult	Difficult to use. Socially difficult to accept because people do not like to have their eyes scanned	Easy to use, but it is associated with criminal investigations
Universality	Not suitable for people who have rheumatic hands or related physical impairments	Suitable for everyone with eyes	Not for people with damaged fingerprints due to daily handling of rough material
Speed of identification	Less than 3 seconds	1.5 seconds	Average verification time 2 seconds. Maximum is 20 seconds
Size for storage of template	Only 9 bytes	40 bytes	1203 bytes. After compression it is smaller than 800 bytes
Long-term stability	Sizes of hands will change for children and can change when someone gains or loses a lot of weight	The retinal vascular pattern is very stable. Only a few diseases or injuries will change this pattern	Sizes of fingerprints change for children. Apart from that they always remain the same
Maturity of technology	Worldwide used in many systems	Used in a fair number of systems	Worldwide used in many systems

Table 4.2 Comparison of various biometrics techniques

Technical factor	*Iris*	*Retina*	*Face*	*Finger-scanning*	*Voice*	*Hand geometry*	*Finger geometry*	*Palm*	*Signature*
Level of Accuracy	Very high	Very high	High	High	High	High	High	High	High
Ease of use	Medium	Low	Medium	High	High	High	High	High	High
Vulnerability to fraud	Very high	Very high	Medium	High	Medium	High	High	High	Medium
Intrusive for human beings	Medium	Medium	High	Medium	High	High	Medium	Medium	Very high
Long-term stability	High	High	Medium	High	Medium	Medium	Medium	High	Medium
Industry standards	-	-	-	ANSI/NIST Data Interchange & FBI Image Compression Standards	Speaker Verification API (SVAPI)	-	-	See finger scanning	-
Factors that may affect performance	Glasses worn by end user	-	Poor lighting, aging of face, glasses, facial hair	Dry, dirty or damaged finger images; age, gender, and race of end-user	Background and network noise, colds, and other factors can change the voice	Diseases such as arthritis and rheumatism in end-users	See Hand Geometry	Dry, dirty, or damaged palm images, age, gender, and race of end-user	Illiteracy; signatures that constantly change or are easily imitated

Deciding When to Apply Biometrics, and What Should Be Considered?

Of course, investigating various existing biometrics systems and products is mandatory. Besides this, there are still many questions that should be answered. What kind of biometrics system is required? Is it an identification system or a verification system? What are the characteristics of the end-user population? What are the ages, genders, ethnic origins, and occupations of the end-user group? In case something is wrong with the biometrics system, what will be the substitute method? What is the accuracy of the biometrics system? Will the population of the system grow? What does the environment look like? At last, a detailed testing plan must be prepared.

4.5 Summary

Biometric devices will continue to improve, becoming even more accurate and reliable as technology evolves. As biometric technologies are more widely accepted, the proliferation of applications should multiply into many phases of our daily activities. The growing interest in the combined use of biometrics and smart cards should also cause an increased growth path for both technologies in the future. Hopefully, in the near future, standards will be available which allow multiple reader technologies from various manufacturers to be utilized within the same system.

4.6 References

[4.1] International Biometrics Industry Association (IBIA). http://www.ibia.org

[4.2] International Biometrics Group (IBG). http://www.biometricgroup.com/

[4.3] The Biometrics Consortium. http://www.biometrics.org

[4.4] Biometrics Research. http://biometrics.cse.msu.edu

[4.5] S. Liu, M. Silverman: A practical guide to biometric security technology. IEEE Computer Society, IT Pro – Security. http://www.computer.org/itpro/homepage/Jan_Feb/security3.htm

[4.6] East Shore Technologies. http://www.east-shore.com/

[4.7] Fingerprint Technologies. http://www.fingerprint.com/

[4.8] FingerPrint USA. http://www.fpusa.com/

[4.9] Biometrics Reports. http://www.biometrics.org/REPORTS/CTSTG96/

[4.10] The Biometrics Consulting Group, LLC. http://biometric-consulting.com

[4.11] Association for Biometrics (AfB), UK. http://www.afb.org.uk/

[4.12] Australian Biotechnology Association. http://www.aba.asn.au/

[4.13] Financial Services Technology Consortium. Biometrics fraud prevention. http://www.fstc.org/

[4.14] Security Industry Association (SIA). http://www.siaonline.org/

[4.15] The Human Identification Project. http://www.asti.dost.gov.ph/

[4.16] GSA's SmartGov. http://policyworks.gov/smartgov/

[4.17] Biometrics in Human Services User Group. http://www.bioapi.org

[4.18] Biometrics and Security. http://www.infosyssec.org/infosyssec/biomet1.htm

[4.19] A. K. Jain, et al. (eds.) (1998) Biometrics: personal identification in networked society. Kluwer, Boston.

[4.20] B. Miller (1994): Vital signs of identity. IEEE Spectrum 32 (2): 22–30.

[4.21] D. Zhang (2000) Automated biometrics: technologies & systems. Kluwer, Boston.

[4.22] D. Zhang (ed.) (2002) Biometrics solutions for authentication in an e-world. Kluwer, Boston.

[4.23] M. Eleccion (1973) Automatic fingerprint identification. IEEE Spectrum 10(9): 36–45.

[4.24] G. Lawton (1998) Biometrics: a new era in security. Computer 16–18.

[4.25] A. Jain, et al. (1997) On-line fingerprint verification. IEEE Trans PAMI 19(4): 302–313.

[4.26] J.P. Campbell (1997) Speaker recognition: a tutorial. Proc IEEE 85(9): 1437–1462.

[4.27] L. Hong, et al. (1998) Integrating faces and fingerprints for personal identification. IEEE Trans PAMI 20(12): 1295–1307.

[4.28] J. Daugman (1993) High confidence visual recognition of persons by a test of statistical independence. IEEE Trans PAMI 15: 1148–1161.

[4.29] Y. Zhang, D. Zhang (2000) A novel text-independent speaker verification method based on the global speaker model. IEEE Trans SMC (Part A) 30(5): 598–602.

[4.30] D. Sims (1994) Biometrics recognition: our hands, eyes and faces give us away. IEEE Comput Graphics & Apps.

[4.31] J. D. Woodward (1997) Biometrics: privacy's foe or privacy's friend? Proc IEEE 85(9): 1480–1492.

[4.32] A. Davis (1997) The body as password. Wired, July Issue.

[4.33] D. R. Richards (1995) Rules of thumb for biometrics systems. Security Manage, October Issue.

[4.34] G. Lawton (1998) Biometrics: a new era in security. IEEE Computer, August Issue.

[4.35] R. Mandelbaum (1994) Vital signs of identity. IEEE Spectrum, February Issue.

[4.36] M. Golfarelli, D. Maio, D. Maltoni (1997) On the error-reject trade-off in biometrics verification systems. IEEE Trans PAMI 19(7): 786–796.

[4.37] R. P. Wildes (1997) Iris recognition: an emerging biometrics technology. Proc IEEE 85(9): 1348–1363.

[4.38] C. Seal, D. McCartney, M. Gifford (1997) Iris recognition for user validation. British Telecommunications Engineering 16.

[4.39] A. K. Jain, H. Lin, P. Harath, R. Bolle (1997) An identity-authentication system using fingerprints. Proc IEEE 85(9): 1365–1388.

[4.40] A. Jain, H. Lin, R. Bolle (1997) On-line fingerprint verification. IEEE Trans PAMI 19(4): 302–313.

[4.41] A. R. Roddy, J. D. Stosz (1997) Fingerprint features: statistical analysis and system performance estimates. Proc IEEE 85(9): 1390–1421.

[4.42] http://www.nwfusion.com/research/biometrics.html

[4.43] http://www.veridicom.com/technology/Biometric%20Applications.pdf

[4.44] http://www.iris-scan.com/iris_recognition_applications.htm

[4.45] http://www.biometritech.com/features/deploywp4.htm

[4.46] http://www.vanguard-fire-security.com/security.htm

[4.47] http://www.fcw.com/geb/articles/2002/0311/web-face-03-04-02.asp

[4.48] http://hydria.u-strasbg.fr/~norman/BAS/intro_to_biometrics.htm

[4.49] http://www.computer.org/itpro/homepage/jan_feb01/security3b.htm

5 Smart Cards and Applications

Weidong Kou[1], Simpson Poon[2], and Edwin M. Knorr[3]

[1] University of Hong Kong
Pokfulam Road, Hong Kong

[2] School of Information Studies
Charles Sturt University, Australia

[3] Department of Computer Science
University of British Columbia, Canada

5.1 Introduction

A smart card is a plastic card with an embedded integrated circuit (IC). A smart card resembles a credit card, with the difference being a chip and (for most smart cards) its metal contacts. A host computer or smart card terminal runs the off-card application and communicates with the card's embedded chip to exchange data and commands. The plastic card usually conforms to physical standards for bank/credit cards, and is a convenient and acceptable way of carrying the chip. Smart cards may contain a microprocessor, random access memory (RAM), read only memory (ROM), and electrically erasable programmable read-only memory (EEPROM). The first patent for a smart card was issued in 1974 to Roland Moreno of France.

Depending on how communication takes place, the smart card can be either contact-based or contactless. For contact-based smart cards, communication takes place through the contacts. The visible contacts cover an area approximately 1cmx1cm (i.e., 100 mm^2); however, the chip itself is usually no more than 25 mm^2. For contactless smart cards, communication takes place through wireless transmission.

Smart cards have some type of non-volatile storage, and can be classified according to whether or not they have a microprocessor. Over half of the smart cards in circulation today do not contain a microprocessor. Such cards are referred to as memory (smart) cards, and are used primarily for storing information or value. These cards have been successfully used for years (primarily in Europe and Asia) for electronic payment in pay phone, vending, and transportation applications, among others.

Smart cards with an embedded microprocessor are sometimes thought of as truly "smart" cards - at least in terms of processing functionality. These cards can perform very reliable security functions, such as authentication, digital signatures, and encryption. A microprocessor allows a smart card to operate independently of a host computer or smart card terminal, thereby enabling essential security operations, such as the creation of a digital signature without the signing key ever leaving the card. These smart cards play a crucial role in information and network security, digital identification, order authorization, and payment processing in electronic commerce applications.

Smart cards have a huge market potential. Currently, billions of smart cards are in use. According to a 1998 report by Gemplus and the Smart Card Industry Association, approximately 805 million smart cards were issued in 1996, and 2.8 billion cards were forecast for 2000. A more recent report from SchlumbergerSema, however, reports that only 1.8 billion cards were issued worldwide in 2001, and furthermore, this figure represents only a 1% increase from 2000. This mild increase stands in contrast to the annual 20% growth rate in recent years, but SchlumbergerSema predicts increases of 7% and 10%, respectively, in 2002 and 2003. Much of this growth is expected to be driven by smart-card-enabled PKI applications, including wireless applications, national ID programs, and network access for enterprise applications.

In terms of applications, prepaid phone cards are still the most popular, followed by mobile communications, and banking. Table 5.1 shows the usage of smart cards. Multi-application cards include applications such as healthcare, loyalty points, secure remote access, and "electronic purse" applications for electronic payment. On electronic-purse cards, real money is represented as a string of bits, and is exchanged among parties.

Table 5.1 Smart card breakdown[1]

By Application	2000 actual	2003 estimate
Pay Phones	1,040,000,000	990,000,000
Mobile Communications	450,000,000	550,000,000
Banking	120,000,000	220,000,000
Others	180,000,000	357,000,000
Total	1,790,000,000	2,117,000,000

[1] Source: SchlumbergerSema, March 2002

By Region	2000 actual	2003 estimate
Europe, Middle East, Africa	895,000,000	974,000,000
Asia Pacific	519,000,000	656,000,000
Latin America	340,000,000	402,000,000
North America	36,000,000	85,000,000
Total	1,790,000,000	2,117,000,000

By Technology	2000 actual	2003 estimate
Memory Cards	1,126,000,000	1,155,000,000
Microprocessor	664,000,000	962,000,000
Multiapplication (of which are Java Cards...)	115,000,000 (53,000,000)	530,000,000 (336,000,000)

5.2 Fundamentals of Smart Card Systems

A smart card *system* is a distributed computing system consisting of smart cards, smart card readers, smart card operating systems, file systems, and communication interfaces. In this section, we describe the components of a typical smart card system.

5.2.1 Smart Cards

Depending on chip type and method of communication with the reader, smart cards can be classified into different categories. First, the chip can either be a memory chip or a microprocessor. Second, smart cards can communicate with readers either using the contacts on the cards, or wirelessly (in the case of contactless cards). In the latter case, radio waves are used to energize and communicate with the chip. Third, smart cards come in at least two sizes: the size of a standard bank/credit card, and a smaller size for a subscriber identification module (SIM) for global system for mobile communications (GSM) cellular phones.

Memory Chips

Smart card memory chips are designed in accordance with their intended applications. The types of memory used for smart cards include:

- Random access memory (RAM):
  This type of memory is used as a short-term work area. Memory contents are lost when the power is switched off.

- Read-only memory (ROM):
  This type of memory is used for storing software.

- Erasable programmable read-only memory (EPROM):
  This type of memory can only be changed once, and is often used in prepaid service cards, such as telephone calling cards that count off minutes of use. (For telephone applications, the cards can be discarded when there are no units of value left because the cards cannot be reloaded with value.)

- Electrically erasable programmable read-only memory (EEPROM):
  This type of memory can store programs or data. The contents of the memory are preserved when power is switched off, and the memory can be modified up to about 100,000 times.

The architecture of a memory chip varies, depending on the intended application. An example is shown in Fig. 5.1. In particular, a smart card memory chip may contain the following data for communication with the reader:

- Smart card issuer
- Smart card serial number
- Counter logic
- Secret codes or keys

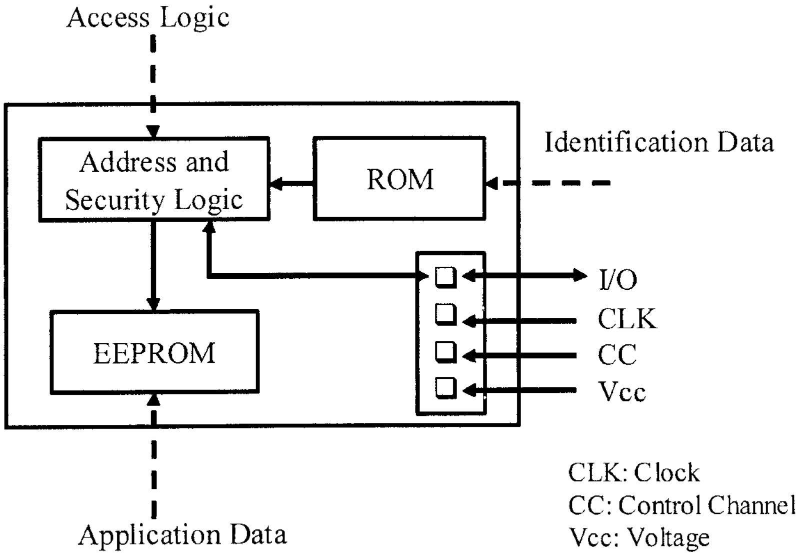


Fig. 5.1 Architecture of a memory smart card

Microprocessor Chips

As shown in Fig. 5.2, microprocessor chips for smart cards contain a CPU, RAM, ROM, EEPROM, I/O control port, and operating system. The CPU is usually an 8-bit processor, which is a smaller and slower version of the CPU used in a typical PC; however, the smart card's CPU could be a 16-bit, 32-bit, or 64-bit version instead. Besides the processor, there are memory components: ROM, which contains the chip's operating system; RAM, which serves as the processor's working memory; and EEPROM, which stores data and program code. Typically, with respect to memory size, RAM is in the range of 256 bytes to 1 KB, ROM is in the range of 16~32 KB, and EEPROM is in the range of 1~16 KB. The connection to outside the chip is via the I/O control port.

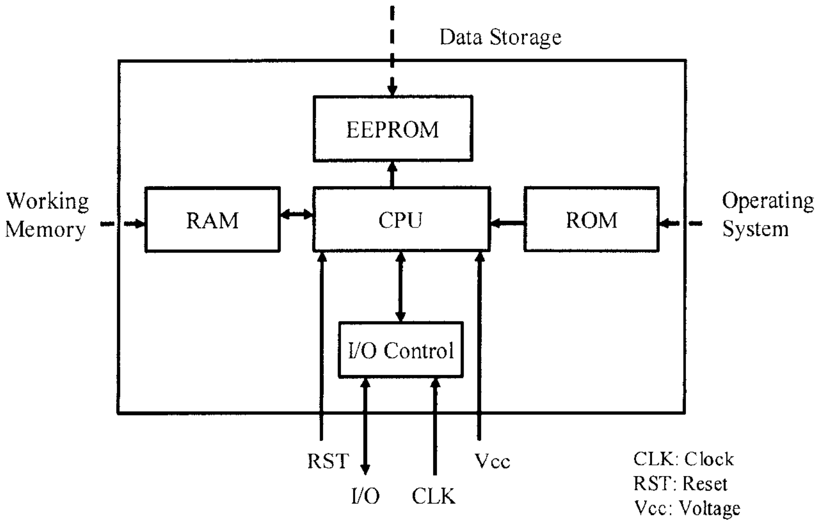


Fig. 5.2 Architecture of a microprocessor smart card

A smart card chip may have an arithmetic co-processor for cryptographic functions. Some chips also have a random-number generator that can facilitate mutual authentication and secure electronic payment.

Contact Smart Cards

Contact smart cards make their physical interface with a smart card reader through an eight-pin contact, as defined in Part 1 of the ISO 7816 standard. These pins are defined as I/O, reset, clock, ground, Vpp (programming voltage), etc. Note that the contacts may become dirty, worn (through use), or damaged (intentionally or unintentionally). Also, hackers use the contacts to launch security attacks. Contactless smart cards are a workaround to most of these problems.

Contactless Smart Cards

Contactless smart cards communicate with a card reader through an antenna embedded in the card. Many such cards have approximately a 10-cm range, but there are systems that work up to a distance of 1 m. The antenna on a smart card can be quite small (e.g., less than 1 cm in diameter). Through an antenna, a contactless smart card can collect its power from a radio frequency (RF) field generated by the card reader. The RF field also transfers information to and from the card.

Since there is no need to insert a contactless smart card into a card reader, such cards can be more convenient. A successful contactless smart card application for public transportation is the Octopus card in Hong Kong, which we describe in detail later in this chapter. Another successful contactless application is access control (via employee badges).

5.2.2 Smart Card Readers

A smart card reader is a device that sets up a communication link or interface between a smart card and a host, such as a PC. Smart card readers are also known as *card acceptance devices* (CADs). They interface with a host through RS232 serial ports, parallel ports, USB ports, PCMCIA slots, infrared IRDA ports, or floppy disk slots. They can also be integrated with a computer keyboard, or embedded into various devices or terminals, such as bank ATMs, kiosks, vending machines, TV set-top boxes, cellular phones, personal digital assistants (PDAs), or handheld computers.

Since a smart card contains no independent power source or clock signal to drive its processor, one of the functions that a reader needs to perform is to provide the card with both power and a clock. In the case of a contact smart card, this is done via the contact pins. In the case of a contactless smart card, the task is accomplished via the embedded antenna.

Depending on whether the smart card is a memory or a microprocessor card, the reader either acts as a translator between the host and the card, or it directly passes commands from the host to the card. For a memory smart card, the reader views the physical card structure to get the exact data address and perform the translation. For a microprocessor smart card, the operating system and logic stored on the smart card directly interpret the commands that have been passed by the reader from the host to the card.

Smart card readers can be classified into two categories: stationary readers and mobile readers. Stationary readers have a permanent connection to the host and are usually powered by the host through the data interface. The host drives the reader and the card, and is responsible for all signaling functions, including initialization and communication. Mobile readers, on the other hand, are stand-alone devices that are battery-powered. A mobile reader initializes communication with the smart card. The host is only concerned with communication with the reader (and not with the smart card).

5.2.3 Smart Card Operating Systems

A smart card operating system typically contains a few thousand bytes of code, and it loads, operates, and manages one or more applications on the card. Unlike DOS, Windows, Unix, or Linux, smart card operating systems are tiny and do not support the rich functionality that these other operating systems provide, such as providing user interfaces or controlling external resources other than the I/O port.

A smart card operating system allows a card reader to send a command to a smart card. The card executes the command, returns a result (if appropriate), and waits for the next command.

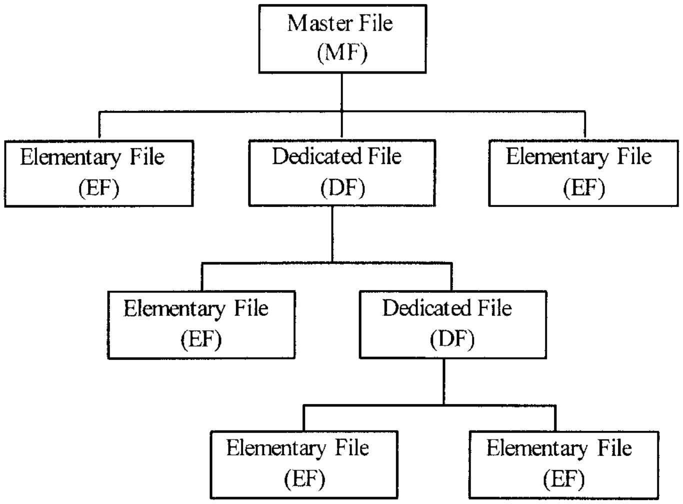


Fig. 5.3 A hierarchical file system for a smart card

Different smart card manufacturers offer different operating systems. Throughout the 1980s and 1990s, various smart card operating systems have been developed for specific applications, such as a data repository. These operating systems are written into ROM. They are proprietary and specific to a smart card chip. A hierarchical file structure, such as that shown in Fig. 5.3, has been used in these systems. Here, the master file serves as the root of the hierarchical file structure, a dedicated file is a directory, and an Elementary File is a leaf node.

As a smart card evolves from a data-storage device to a transaction device, a hierarchical file structure may not be the best choice. New data structures have been created using the industry standard "Create Table" and "Create View" SQL statements (i.e., from the database community). The key advantage is to allow different applications to share a common data structure.

In addition to sharing, it is important for an application not to read or overwrite another application's data, without that other application's consent. This means that the smart card operating system must control memory allocation for each application (e.g., for loading or retrieving data from RAM and EEPROM storage areas).

Given that a smart card application may need to be updated after a smart card is issued, and that various smart card readers may need to access information on the card, new extensions are required for smart card operating systems. Such extensions include an application programming interface (API) and an object-oriented programming language (e.g., Java) that can be used on many different platforms ranging from PCs to hand-held devices.

5.2.4 Communication Interface

Communication between a smart card and a reader typically goes through a half-duplex physical channel on which the reader and the card can only transmit in turn (i.e., the other party has to be in reception mode). In this section, we examine how communication protocols on top of this half-duplex physical channel are established, and how data is transferred.

When establishing communication between a smart card and a reader, it is always the reader that takes the initiative. The card never transmits data without an external request from the reader. To illustrate, consider a contact smart card. When the card is inserted into the reader, five contacts on the card are electrically activated, and the card automatically executes a power-on-reset. Then, the answer to reset (ATR) is sent to the reader. The reader may optionally send a PTS (protocol type select) request command to the card after it successfully evaluates the ATR, and the card responds to the reader with a PTS-response. Both ATR and PTS are independent of the transmission protocol, and they are used for initialization and for setting various transmission parameters. After initialization, the reader sends the card the first command. The card processes the command and sends a response. This kind of command-response protocol is how the reader and the card communicate.

Communication between the card and the reader takes place serially through a bit-serial data stream. The bit order for converting a byte into the bit-serial data stream must be considered. In the direct convention, the first data bit after the start bit is the lowest in the byte, where the start bit is used for indicating the beginning of each serially transmitted byte. A parity bit is at the end of each byte. One or two stop bits may also be added after the parity bit.

In terms of transmission, a basic question is whether data should be transferred in byte mode or in block mode. In answer to this question, there are two transmission protocols at the data link layer, namely T=0 and T=1 protocols. The T=0 protocol is the asynchronous, half-duplex, byte-oriented protocol, which is covered by the ISO/IEC 7816-3 standard, dominating in Europe and widely used in various smart card systems (e.g., GSM applications). The T=1 protocol is the asynchronous, half-duplex, block-oriented protocol, which is covered by the ISO/IEC 7816-3 Amendment 1 standard. The data structures used in the exchange between the reader and the card in the command-response protocol are called transmission protocol data units (TPDUs).

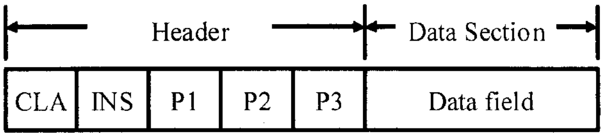
On top of the data link layer protocols (T=0 and T=1), application-layer protocols can be defined for smart card applications to exchange control and information between the card and the reader. There are two application protocols that have been defined in the ISO/IEC 7816-4 standard. One protocol is for providing a file system for storing and retrieving information on a smart card, and the other is for accessing security services on the card. The former is defined in the form of a collection of functions for selecting, reading, writing, and erasing files, while the latter is defined in the form of a series of security functions. To support these two protocols, the ISO/IEC 7816-4 standard defines data units in the application layer, called application protocol data units (APDUs), which are used for data exchange between the card and the reader.

Fig. 5.4 is a high-level summary of our discussion about the communication model between a smart card and a smart card reader.

Application Layer	ISO/IEC 7816-4 ISO/IEC 7816-7
Data Link Layer	ISO/IEC 7816-3 (T=0) ISO/IEC 7816-3 Amd. 1(T=1)
Physical Layer	ISO/IEC 7816-3 (Contact cards) ISO/IEC 10536-3 (Contactless cards)

Fig. 5.4 Communication model between a smart card and a reader

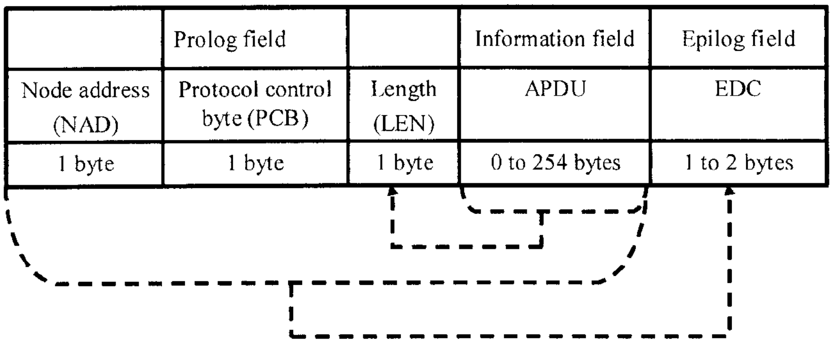
Before discussing the communication protocols in further detail, let us examine the TPDU and APDU data structures. Fig. 5.5 shows the TPDU data structure for a command with the transmission protocol T=0, which consists of a header and optionally a data section. The header has five fields: a class byte (CLA), an instruction byte (INS), and three parameter bytes (P1-P3), where parameter P3 is a length datum to indicate the length of the data byte transferred to or from the card. The data structure for a response with T=0 consists of an acknowledge byte (ACK), a flow control byte (NULL) to let the reader know that the card is still processing the command and is not yet ready to receive another command, a status return code (SW1), and optionally a return code (SW2) to indicate the amount of data (if the response contains data).



CLA	INS	P1	P2	P3	Data field

Fig. 5.5 TPDU command data structure with T=0

The transmission protocol T=1 is block-oriented. There are three types of blocks in the T=1 protocol: information block (I-block), receive ready block (R-block), and supervisory block (S-block). Each block contains two mandatory fields (prolog and epilog), and one optional field (information), as shown in Fig. 5.6. The prolog field consists of three bytes: node address byte (NAD), protocol control byte (PCB), and length byte (LEN). The information field contains the application layer's data (APDU) and it may be up to 254 bytes in length. The epilog field contains an error detection code with a length of either 1 or 2 bytes.



	Prolog field		Information field	Epilog field
Node address (NAD)	Protocol control byte (PCB)	Length (LEN)	APDU	EDC
1 byte	1 byte	1 byte	0 to 254 bytes	1 to 2 bytes

Fig. 5.6 T=1 transmission block structure

There are two APDU data structures: the command APDU structure and the response APDU structure. The command APDU structure consists of a header and a body, as shown in Fig. 5.7. The header includes CLA, INS, P1, and P2 fields. The body may be of variable length or it may be absent (when the data field is empty). The Lc field specifies the length of data sent to the card. The Le field indicates the length of data to be sent back from the card. There are four cases for the command APDU structure:

Case 1: No data is to be exchanged, and the command APDU structure only contains the header.
Case 2: No data is transferred *to* the card, but data is transferred *from* the card. The command APDU structure contains the header and the length (Le) of data returned from the card.
Case 3: Data is transferred *to* the card, but not *from* the card. The command APDU structure contains the header, the length (Lc) of data transferred to the card, and the data field.
Case 4: Data is transferred both to and from the card. The command APDU structure contains the header, the length (Lc) of data transferred to the card, the data field, and the length (Le) of data returned from the card.

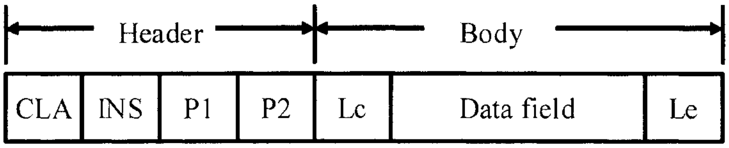


Fig. 5.7 Data structure of an APDU command

The response APDU structure consists of an optional body and a mandatory trailer, shown as in Fig. 5.8. The body contains the data field. The data length is determined in the Le field of the previous command APDU structure. The trailer contains two bytes, SW1 and SW2, which are the designated return codes for the response to the command.

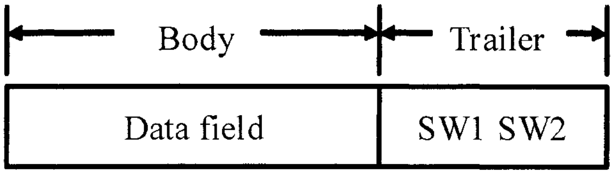


Fig. 5.8 Data structure of an APDU response

5.3 Java Card

A Java Card implementation of a smart card application runs programs written in a subset of the Java programming language, in byte-code form. (Java Card technology can also be applied to other resource-constrained devices.) Java Card defines a runtime environment supporting the smart card's memory, communication protocols, security, and application execution. It changes the landscape of the smart card world for the following reasons:

- **Platform independence**
  Applications written for one smart card platform can run on other platforms (from different vendors), provided that those platforms support Java Card technology.

- **Ability to run multiple applications**
  Downloadable Java byte-code enables multiple applications from multiple vendors to be run securely on a single smart card. For example, a single Java Card can be used as an electronic purse, an employee badge (for accessing buildings), a healthcare card, and a telephone card.

- **Ease of upgrades**
  Java Card technology allows the card issuer to: (a) upgrade existing applications on a card, and (b) download new, additional applications to a card.

- **Compatibility with existing standards**
  Java Card technology is compatible with existing smart card standards, such as ISO 7816 and EMV (Europay, MasterCard, Visa).

- **Security**
  The Java virtual machine implements Java-language security policies even though the Java Security Manager class is not supported by Java Card. This means that the level of access to all methods and instances of variables is strictly controlled. Java's "no pointers" feature prevents malicious programs from accessing data in memory.

- **Availability of sophisticated Java application development tools**
  There are a number of integrated Java development tools from leading software vendors such as Borland, IBM, Microsoft, Sun, and Symantec. Java Card developers can choose a tool to create and debug Java Card applications. This is in contrast to traditional smart card application development where a smart card application is coded in assembly language, compiled into machine code, and then burned into ROM. The traditional development method needs a relatively long time to develop and deploy a smart card application, and once the application is deployed it is hard to

make changes. With Java development tools, a Java Card application can be developed and deployed easily and quickly. Furthermore, the deployed Java Card applications can be easily upgraded.

- **Large and growing pool of experienced Java programmers**
  Due to Java's popularity, there is a large and growing pool of experienced Java programmers. Java programmers can easily become Java Card programmers; consequently, the cost of acquiring and training Java Card programmers is minimized.

Due to the resource constraints of smart cards, Java Card only supports some of the features of the Java language (e.g., small primitive data types, 1-D arrays, Java object-oriented features, Java packages, classes, interfaces, and exceptions), but preserves many benefits of the Java language, including productivity, security, robustness, tools, and portability. The Java Card virtual machine is split into two parts: one part running off-card, and the other part running on-card. The assumption is that many processing tasks that require significant resources or that do not have to be executed at runtime can be run on the off-card part of the Java Card virtual machine.

Java Card separates the smart card system and its applications, and uses a well-defined high-level API for application requests for system services and resources. Java Card technology defines a platform consisting of three parts:

- **Java Card virtual machine (JCVM)**
  The JCVM consists of two separate pieces: the Java Card interpreter and the Java Card converter.
  - **Interpreter**
    This is the on-card part of the Java Card virtual machine, providing runtime support for the Java language model. The interpreter executes Java byte-code instructions and Java applets, controls memory allocation and object creation, and enforces runtime security.
  - **Converter**
    This is the off-card part of the Java Card virtual machine. The converter loads and preprocesses all the Java class files that make up a Java package and converts the package to a converted applet (CAP) file. It also verifies the Java-class load images, checks for violations, initializes static variables, optimizes the byte-code, and allocates storage.
- **Java Card runtime environment (JCRE)**
  The JCRE consists of the Java Card system components that run inside a smart card, and serves as the operating system of the smart card. It manages card resources, executes Java applets, and ensures on-card system

and applet security. It is also responsible for network communication. JCRE has three layers, as shown in Fig. 5.9. The bottom layer contains the JCVM and the native methods that support the low-level communication protocols, memory management, and implementation of cryptographic functions. The middle layer contains system classes that manage transactions and communication, and control applet creation, selection, and deselection. The upper layer contains framework classes, industry-specific extensions, and the installer. The framework classes define the APIs that make the creation of an applet relatively easy. The industry-specific extensions are the add-on libraries supplied by specific industries or businesses to provide additional services. The installer is used for easily upgrading existing applications and for downloading new applications after the smart card has been issued.

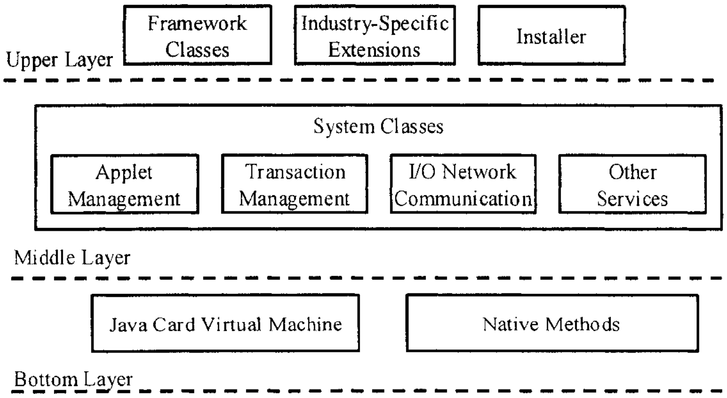


Fig. 5.9 Three-layered Java Card runtime environment

- **Java Card API**
  The Java Card API defines the calling conventions for programming smart card applications, by which the applications can access the JCRE and native methods. It specifies a subset of Java that is tailored for use in smart cards and other devices with limited memory. The Java Card API consists of four packages: three are core packages, and one is an extension package. They are java.lang, java.framework, javacard.security, and javacardx.crypto. These packages contain a set of customized compact classes supporting smart card standards, such as ISO 7816, and providing cryptographic services. The significance of the Java Card API is that it frees smart card developers from the development limitations (e.g., a proprietary assembly language) imposed by specific smart card manufacturers.

5.4 Smart Card Standards

In order for smart cards to be used en masse in the marketplace, interoperability between smart card systems from different vendors must be supported. Thus, there must be a standard upon which every vendor agrees. A standard is an open specification that lays down the rules, guidelines, or requirements that (a) have been proposed and agreed to, with the consensus of many interested parties, and (b) have been adopted by a recognized standards organization, such as the international organization for standardization (ISO) or the international electrotechnical commission (IEC).

Many standards organizations, such as ISO, IEC, European Telecommunications Standards Institute (ETSI), and American National Standards Institute (ANSI), have been actively involved in smart card standardization efforts. As a result, a set of standards on smart cards has been produced. Some of these standards are listed in Table 5.2. These standards play a key role in promoting the interoperability of smart card systems from different manufacturers and vendors.

Table 5.2 Selected smart card standards

Standard	Subject
ISO/IEC 7810	Physical characteristics
ISO/IEC 7811	Recording techniques: magnetic stripe & embossing
ISO/IEC 7812	Numbering system
ISO/IEC 7813	Financial transaction cards
ISO/IEC 7816	Contact cards
ISO 10373	Test methods
ISO 10536	Contactless cards
ISO 14443	Remote coupling communication cards

ISO/IEC 7810 defines the physical characteristics of smart cards, including visual and physical durability, embossing, and the location of a magnetic stripe. The information that identifies the cardholder and supports the transaction via the card may be conveyed via either a magnetic stripe, or a chip. ISO/IEC 7811 defines how to encode the information. ISO/IEC 7812 specifies how to construct the card identification number, which is up to 19 characters long and has three components: issuer ID number, individual account ID number, and check digit. ISO/IEC 7813 defines the location of embossed characters on the card. ISO/IEC 7816 is for contact smart cards, and specifies the following distinct parts:

- Physical characteristics (Part 1)
- Dimensions and locations of contacts (Part 2)

- Electronic signals and transmission protocols (Part 3)
    - Protocol type T=1, asynchronous half-duplex block transmission protocol (Part 3, Amendment 1)
    - Revision of protocol type selection (Part 3, Amendment 2)
- Inter-industry commands (Part 4)
- Numbering system and registration procedure for application identifiers (Part 5)
- Data elements for interchange (Part 6)
- Query language commands (Part 7)
- Security architecture (Part 8)
- Inter-industry enhanced commands (Part 9)
- Synchronous cards (Part 10)

ISO 10373 defines the test methods for smart cards. ISO 10536 covers contactless smart cards and defines a close-coupled card having a range up to 10 cm. It comprises the following parts:

- Physical characteristics (Part 1)
- Dimension and locations of the coupling elements (Part 2)
- Electronic signals and reset procedures (Part 3)
- Answer to reset and transmission protocols (Part 4)

ISO 14443 is a specification for contactless cards that changes the contact description to an antenna, and defines the protocol for communication over the air. It consists of the following four parts:

- Physical characteristics (Part 1)
- Radio frequency interface (Part 2)
- Transmission protocols (Part 3)
- Transmission security features (Part 4)

In addition to these ISO/IEC standards, there are smart card industry de facto standards and other regional standards. For example, the following standards define the operation of a smart card for various applications.

- EMV: This specification for payment systems based on ISO 7816 defines the content, structure, and programming of chip-based payment cards. It defines how smart cards exchange information with a payment terminal (e.g., PIN checking), and how security is enhanced by preventing the reading of certain low-level information.
- PC/SC: The PC/Smart Card architecture is an open architecture defined by various smart card and PC operating-system vendors including CP8 Transac, Schlumberger, Siemens Nixdorf, HP, and Microsoft. It defines a general-purpose architecture for multiple applications to share smart

card devices attached to a system through low-level device interfaces, device-independent APIs, and resource management.

- OpenCard: The Open Card Framework (OCF) specifications are open specifications allowing applications to be independent of the specific design of smart cards from different manufacturers. The difference between Open Card and Java Card is that Open Card runs Java on the host or terminal side, whereas Java Card runs (a subset of) Java on the smart card itself.
- ETS 300 608: This is a specification from the European Telecommunications Standards Institute defining a smaller-sized smart card (SIM card) to fit into GSM phones (GSM 11.11).

5.5 Smart Cards and Security

Smart cards are excellent vehicles for implementing security. They are a crucial part of security infrastructures. In the following sections, we examine the role of smart cards in key management, digital signatures, identification, authentication, and authorization.

5.5.1 Smart Cards in Key Management

Key management is essential for security, and is the hardest part of security. Secure electronic commerce applications rely on secure algorithms or protocols involving keys, and the key information must be kept secret. It is not easy to invent a new security algorithm/protocol that will be adopted widely for electronic commerce applications; it is even harder to keep key information secret.

There are millions of users of electronic commerce applications. It can be quite challenging to create, distribute, store, retrieve, and destroy keys for millions of users. Although an asymmetric cryptographic system, as discussed in previous chapters, can be used to solve the key management problem, a problem remains about how to keep one's private key confidential. Smart cards offer a solution to this problem. An individual's private key can be stored on a smart card with the aid of a PIN. The PIN can be used to generate a key to encrypt the private key. Then, if the smart card is lost or stolen, no one will be able to access the private key without knowing the PIN.

5.5.2 Smart Cards in Digital Signatures

A digital signature is a piece of data that is created with a signer's private signature key and is a function of the message being digitally signed. To generate a digital signature, a private signature key is required. As we discussed in the last section, smart card can be used for storing this private key.

A digital certificate is an electronic set of credentials for a signer, issued by a trusted authority called a certificate authority (CA). This confirms both the signer's identity and their public key. In some digital-signature implementations, a digital certificate of the signer is required as an appendix to the signed message. This makes the verification process simpler, since the certificate accompanies the message. In this case, it is no longer necessary to obtain the signer's digital certificate from an X.500 directory in a public key infrastructure. Smart cards can be used to store one's digital certificate in addition to one's private signature key.

5.5.3 Smart Cards in Identification

Identification is essential to secure electronic commerce. Smart cards can be used as a means of identification, in place of other forms of ID, such as passports. Various governments have expressed interest in national ID cards, including Malaysia, Hong Kong, and Singapore. Such cards can be used to securely store personal ID, healthcare information, and other data (including financial data), on a single smart card. For example, the Hong Kong Special Administration Region government plans to issue citizen identification cards, that is, smart cards containing personal identification information.

As mentioned in Section 5.5.1, if an individual's smart card containing encrypted or protected information is lost or stolen, no one else will be able to use it. In this regard, smart cards can be a better form of identification than passports. The use of a smart card for identification can provide more efficient processing of individuals at international border checkpoints.

The use of smart cards can also reduce fraud in healthcare and welfare systems. These two systems are well known for abuse and high costs (e.g., due to claims by ineligible recipients, or due to multiple claims in one or more jurisdictions). Banks and credit card companies can save millions of dollars in losses due to fraud, again by using smart cards as a form of cardholder identification.

5.5.4 Smart Cards in Authentication

As discussed in Chapters 2-3, authentication is the process by which an entity's identity is verified. Authentication is typically based one of the following three criteria:

- Something a person knows, such as a PIN or password
- Something a person possesses, such as a smart card

- Something a person uniquely has and cannot easily change, such as a fingerprint, iris image, facial image/structure, voice pattern, etc.

Smart cards can utilize all three of these criteria for authentication. First, a smart card can have a PIN that is only known to the owner of the card. Second, like a bank (ATM) card, each smart card has a unique serial number. Third, unlike a typical bank card, a smart card can store much more information, including information about fingerprints, iris images, etc., that serve to uniquely identify the cardholder (see Chapter 4). Such biometric information can be used to achieve a very high level of security.

Besides authentication via a PIN and the presence of the card, third-party authentication can be performed (either locally or remotely) using a private signature key stored on a smart card. To authenticate the cardholder, the cardholder's digital signature may be verified using a signature-verification process via a third party (i.e., a certificate authority). This third-party authentication is extremely useful for electronic commerce conducted over the Internet.

5.5.5 Smart Cards in Authorization

In electronic commerce, the authorization of a purchase or of a payment for the purchase is required, which can be carried out using smart cards. In the business-to-consumer (B2C) e-commerce applications, the *secure electronic transaction* (SET) protocol (see Chapter 10 for the detailed information about SET) can be used for authorization. In the business-to-business (B2B) e-commerce applications, if the dollar amount of the order is high, multiple levels of authorization may be required. For example, the order may have to be accompanied by the digital signature of the chief financial officer (CFO). To enable secure authorization, a smart card is an ideal tool because a given smart card is unique to the CFO, and no other individual will be able to create the CFO's digital signature. Hence, the likelihood of someone creating a bogus purchase order is minimized.

Smart cards can also be used to implement a hierarchical security scheme for access control, whereby certain individuals within a company are permitted to modify (i.e., add, update, or delete items from) a purchase requisition before it is sent for authorization. This gives authorized individuals the ability to override details on a purchase requisition. Digital certificates can be used to authenticate these individuals.

5.5.6 Summary of Smart Cards and Security

A smart card can be issued to an authorized person and be carried around by that person. The significance of smart cards is that they can be used to securely store the private keys, the digital certificates containing the public keys, and the cryptographic algorithms. Given that a private key never leaves the smart card, and the cryptographic algorithms on the card are used for security purposes, no third party can intercept the private key by listening to the communication between the card and the reader. In addition, the private key on the card can be protected using a PIN. This enables only the authorized person to make use of the private key for security purposes. The smart card system can even prevent further use of the card by locking out the card after a few unsuccessful PIN attempts.

5.6 Smart Card Applications

Smart cards have been widely adopted for many applications throughout the world, but especially in Europe and Asia Pacific. Typical smart card applications include:

- Electronic payment
- Access control
- Telecommunications
- Healthcare
- Transportation
- Identification

5.6.1 Electronic Payment

Smart cards play an important role in payment processing in electronic commerce. They can be used to store and process "value" or digital money, and they can be used to add an additional level of security to a credit card or debit card.

A *stored value* smart card cannot be reloaded, and is issued with some fixed amount of value or money (e.g., $20). As a user purchases goods or services with the card, the monetary value on the card is gradually decremented. Stored value cards have limited hardware functionality and do not contain a microprocessor. The card is decremented by a host application that interfaces with the smart card through a card reader.

An *electronic purse*, on the other hand, is a reloadable smart card. It contains a microprocessor, not only to perform monetary calculations but also to securely store the digital money, to authenticate the host application, and to perform secure communication with the host. A PIN can be used to "lock" the funds on a card to prevent other people from using the card. A Mondex card is an example of an electronic purse. Mondex digital money can even be exchanged between two smart cards belonging to family or friends, using a device called an electronic wallet.

To prevent a hacker from counterfeiting digital money, it is essential that the smart card guard against unauthorized access. Increases and decreases in monetary value must only take place with accepted host applications, using accepted protocols. Digital certificates are required used to authenticate the host and the smart card.

Not only can smart cards be used in credit card or debit card payment processing, they can be used to write digital signatures for electronic cheques. Electronic cheques are based upon a bank account debit system, and are paperless cheques that can be sent electronically from one entity to another. The receiving entity can endorse the cheque via another digital signature, and e-mail it to a bank.

Finally, we note that privacy is an integral part of payment processing. Smart cards can facilitate privacy through digital signatures or the use of anonymous digital money.

5.6.2 Access Control

Smart cards can be used to facilitate authorization to physical or logical sites and resources. For example, smart cards can be used in corporate, government, and military environments for physical access control to buildings, rooms, and parking lots. In addition, smart cards can be used for controlling access to, and operation of, designated physical assets, such as:

- Machines
- Vehicles
- Computer equipment
- Telecommunication equipment
- Laboratory research equipment
- Dangerous arms
- Other equipment

There is great potential in employing biometrics in ID cards for strict physical access control of military, government, and financial facilities and assets. Such applications became even more important following the September 11, 2001 terrorist attacks against the United States.

With respect to logical access, smart cards can serve as a form of identification for remote, online access to workstations, files, databases, and networks. Smart cards can be used to implement security using biometrics, without the need for a central, online database. In particular, they can replace many USERID/password scenarios with automated equivalents, and can provide a very high level of security. For situations where individuals are often working from different terminals, smart card solutions for network access are particularly attractive. Network security will become increasingly important for the Internet.

5.6.3 Telecommunications

Smart cards have been used in telecommunications for years. Typical applications include payment cards for public telephony, and subscriber identity modules (SIM) cards for GSM mobile communications.

Advantages of using smart cards for public telephony include reduced costs of operation since there is no need to collect cash, and theft deterrence (i.e., there is no money available to be stolen).

The use of SIM cards in mobile telephony has enhanced security of GSM because with SIM cards, user authentication, integrity, and confidentiality of voice and data can be provided.

5.6.4 Healthcare

Smart cards are used in healthcare in various ways, including facilitating registration/information in emergency-care situations. For example, in an emergency, a doctor other than the patient's regular physician can access the patient's health information (e.g., blood type, allergies, medicines, special needs, contact information).

Medical insurance companies like smart cards because smart cards can provide information about a patient's insurance eligibility and coverage. They can also be used in an electronic claim submission procedure since both insurance data and patient information can be read and verified from the smart card.

Smart cards are good vehicles for controlling healthcare costs by preventing fraud, especially in public healthcare systems where there may be no good way of verifying eligibility for medical services. Some states or provinces (e.g., British Columbia, Canada) have far more healthcare-benefit cards (not smart cards) in circulation than there are people in the population.

Electronic prescriptions with the physician's digital signature can be stored on smart cards, thereby reducing errors or misunderstandings, minimizing potential drug interactions, and reducing fraud (e.g., some patients visit many physicians, or forge prescriptions, in order to obtain drugs for resale). Healthcare professionals can also use smart cards to control access to unattended workstations in hospital wards.

5.6.5 Transportation

Smart cards have been successfully deployed in transportation applications in many cities, including Hong Kong and Shanghai in China. These applications include:

- Drivers' licenses
- Parking permits
- Taxi payments
- Local public transportation
- Train and air travel
- Electronic toll collection

Transportation is a smart card application that can reach a critical mass of people. In Section 5.7, we will take a close look at Octopus cards in the Hong Kong local transportation fare-collection system.

5.6.6 Identification

Smart cards are emerging as an excellent tool for personal identification in corporate, government, and university applications. Many organizations are using, or plan to use, smart cards as employee badges for multiple purposes. As mentioned previously, some nations are in the process of deploying smart cards as national identification cards.

In many university campuses, all-purpose student ID cards have been used for various purposes, including electronic payments for applications such as:

- Vending machines
- Laundry machines
- Photocopiers
- Meal payments in cafeterias

These student cards can also be used for identification or access in the following applications:

- Course registration
- Student union or club activities
- Libraries
- Athletic facilities
- Medical care

5.7 A Case Study in Smart Cards: Hong Kong's Octopus Card

5.7.1 The Rise of the Octopus Card

The history of the Octopus card started in June 1994 when Hong Kong's five major public transportation operators, namely, Mass Transit Railway Corp. (MTRC)[2], Kowloon-Canton Railway Corp. (KCRC)[3], Kowloon Motor Bus Company (KMB), Citybus Ltd., and the Hong Kong and Yaumatei Ferry systems (HKF), formed a joint venture company, Creative Star Limited (CSL), to develop an automated fare collection system based on contactless smart cards. The fare collection contract, valued at US$55 million, was awarded to ERG Australia Limited and its subsidiary AES Prodata, which subsequently awarded the contactless card portion of the contract to Sony and Mitsubishi Corporation. These contactless, reloadable smart cards, known as Octopus cards, were introduced to the general public in September 1997.

An estimated 10 million passenger journeys are made each day on Hong Kong's wide variety of public transportation services. According to a 1998 report by Industry Canada, the Creative Star Octopus System, when launched, was the largest, integrated, contactless, smart card, fare collection system in the world, and accounted for approximately US$13 million[4] in daily transactions.

Here is how the system works. Each of the operators' computer networks is linked to the Creative Star Clearing House system, which in turn apportions revenue to the operators and deposits funds into the appropriate bank accounts. In mid-2000, there were 6.5 million cards in use, with millions more to follow. Users have the ability to reload their cards with cash (HK$100 is a typical amount). Cardholders can reload their cards in any MTRC and KCRC station, as well as in any of the 368 7-Eleven convenience stores within Hong Kong. Octopus cards can also be used for other kinds of applications, such as purchasing food or merchandise, and even serving as an employee badge (Leong, 2000).

In 1998, Creative Star negotiated with Mondex and VisaCash to incorporate an electronic-purse function into its originally closed system, with the MTRC still owning a 67.8% stake in Creative Star.

[2] MTRC is the metropolitan underground railway system in Hong Kong.

[3] KCRC is an electric railway system that connects the Kowloon peninsula to the New Territories.

[4] At the time of writing, US$1 equals approximately HK$7.8.

5.7.2 Debit Cards in the Passenger Transportation Industry

Although various payment-processing applications already exist for smart cards, transit fare collection requires special considerations to ensure its success (Goldfinger, 1988). We itemize these criteria, modeling them as network goods, as follows.

Deployment and Apprehension

When smart cards were initially introduced for fare collection, there was some apprehension due to technology "hiccups", and the fact that passengers had to adjust to a new system. In those cases where an existing system was already in place (e.g., magnetic stripe cards), a smooth and short cutover was required to ensure the success of the smart card deployment. Some technical hiccups and customer apprehension had been experienced in previous smart card pilot studies (e.g., Mondex in Manhattan, New York City, in 1998, although that pilot did not involve the transportation sector). The bottom-line is that customers and vendors demand smooth rollouts; otherwise, they will lose confidence in the new technology and be more resistant to embracing it.

Co-operation Among Linked Fare Systems

In the case of passenger transportation applications, smart card systems often suffer from incompatibility with other fare systems at either the technology or the business level. At the technology level, vendors using incompatible technologies may develop different kinds of smart card systems that are adopted by different transportation providers. Sometimes, many smart card fare systems are used in one mode of transportation (e.g., buses), but not in another. Sometimes, due to different governing and policy-making bodies, the smart card systems are not integrated, even though there is a logical flow of passengers between two transportation systems (e.g., from trains to buses). And of course, customers are inconvenienced if they need to deal with multiple cards for similar applications.

Contactless Requirements

As mentioned earlier, the term "contactless" means that a transaction performed using a smart card does not require physical contact between the card and the card reader. In fact, many successful transportation-ticketing systems are contactless, and this is important for a number of reasons (Goldfinger, 1998). The user flow rate in some of these systems can be as high as one million or more per day (averaging 19 users per second). During peak hours, the flow rate can be two to three times the average hourly flow rate for the day. In order to handle such a volume, it is important to make sure that the processing time per user is as short as possible.

The critical delay for processing is often not due to transaction processing time, but due to human activity (i.e., locating or fetching the card, and then placing it into the reader). Slot-based cards require the user to physically place the card into a slot and this is the major source of delay for such systems, especially when customers are carrying bags or packages, and there is a queue of passengers boarding.

5.7.3 Analyzing the Success of the Octopus Card

When the Octopus card was introduced in 1997, there were already smart card systems in use, mainly in the form of debit systems introduced by major credit card companies, namely, VisaCash by Visa International, and Mondex by MasterCard. Both VisaCash and Mondex had two key competitive advantages: a large international customer base, and backing by two of the world's largest credit card companies. Both VisaCash and Mondex were in trials around the world. In Hong Kong, Mondex had just launched a trial of its smart card technology, and the results were positive. Given that the Octopus system was still in its infancy, traditional wisdom would have pointed to the demise of an unproven smart card system targeting a local application (i.e., passenger transportation), especially since a magnetic, debit card system was already in place. However, there were a number of factors that made the Octopus card a success, which we outline as follows.

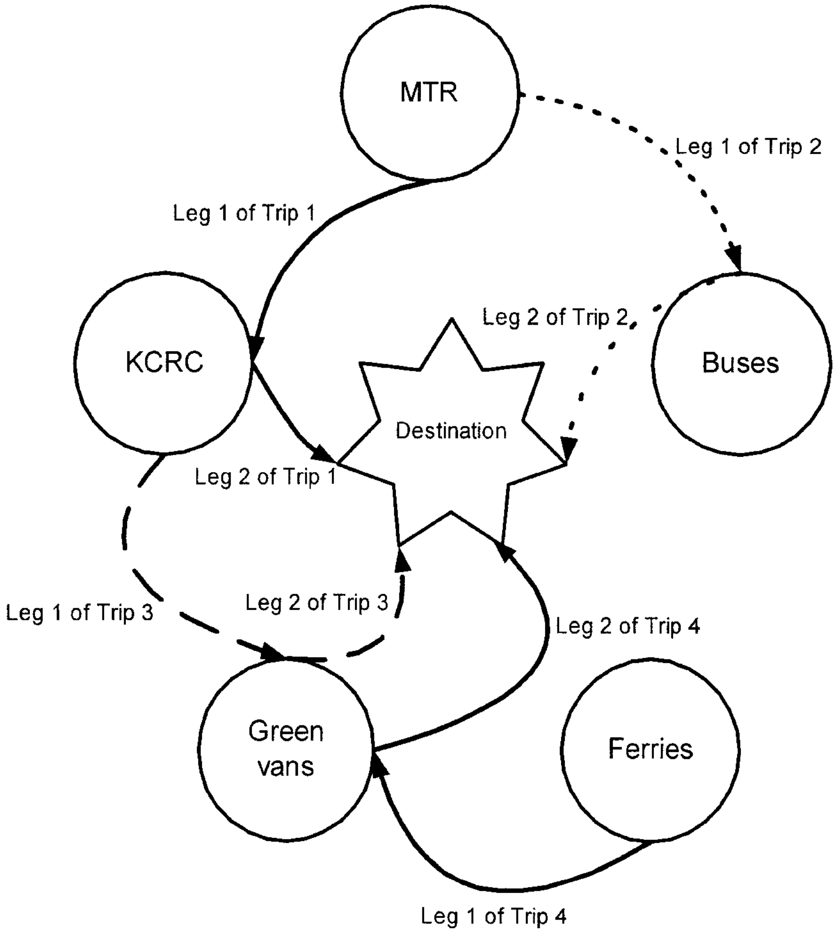


Fig. 5.10 Examples of types of multi-leg trips made in Hong Kong (Source: Poon & Chau, 2001)

A High-Density City with a Heavy Reliance on Public Transportation

Hong Kong is special in the sense that it has one of the highest population densities in the world, combined with a relatively low rate of private vehicle ownership. The highly congested road system and the high cost of car ownership mean that many Hong Kong people use public transportation for their day-to-day activities. The passenger-transportation sector in Hong Kong includes the MTRC, KCRC, bus companies, ferries, and other auxiliary transit systems, such as the "green vans" (a public, light bus system having designated routes). On average, an individual needs to make at least one return trip using the public transport system to go to work, attend school, etc. Given a population of 7.5 million people, if 70% of the population were doing one such trip per day, and spending on average HK$16.00 (approx. US$2.00) per return trip, then the daily transactions of the Octopus system would total approximately US$11 million. Due to the way the Hong Kong passenger transportation system is set up, passengers often need more than one mode of transportation; consequently, the daily transaction volume can be even higher.

Sector Wide Adoption with a Common Technological Standard

Although the use of a stored-value card for local transportation payments has been in place for over two decades in Hong Kong, the diffusion of such an application beyond the Mass Transit Railway Corporation (MTRC) and the Kowloon-Canton Railway Corporation (KCRC) was a recent development. The early stored-value card system used by the railway systems was based on magnetic stripes, and required insertion of the card into a reader slot at a turnstile. For all other modes of public transportation in Hong Kong (e.g., ferries, buses and mini-buses), payments were made by tendering the exact fare using coins into an on-board coin box.

Creative Star Limited was created and backed by the biggest passenger transportation companies in Hong Kong. These companies together operated more than 70% of the passenger transportation business in Hong Kong. Their unanimous adoption of the Octopus card generated an instant critical mass. A critical mass is an essential property for the flow of network goods. This also reflects a spirit of cooperation because the member companies jointly owning Creative Star were direct competitors in some cases.

It has been pointed out that if both the current and new systems coexisted, chances are that the old system would jeopardize the adoption of the new one because of the existing infrastructure and habitual usage (Goldfinger, 1998). In the case of the Octopus card system, the deployment was a quick conversion over a period of a few months, and users had no choice but to use the new system.

As this chapter was written, a sector-wide adoption situation had been achieved with about 40 passenger transportation companies accepting the Octopus card. Almost 7 million cards had been issued by early 2001, of which 1.4 million were sold to children under 18 years of age (Rader and Maghiros, 2001).

Captured Market with Reasons for Adoption

Although coin payment has been well accepted in Hong Kong's passenger transportation system, it has nevertheless been a hassle because of the "exact fare" rule that requires passengers to have the exact amount ready, usually in the form of coins. Getting change is difficult because there is often no money-change facility near the terminals or bus stops. Sometimes, the only way to get change is from nearby shops or other passengers, neither of which is a welcoming move.

The Octopus card, on the other hand, offers a number of conveniences, especially when considering the contactless property. For those people who have their Octopus cards buried under their belongings in a handbag, for example, there is no need to physically place the card into a reader because the card can be read where it is, providing the card is sufficiently close to the reader. The contactless nature of the card allows for fast scanning. In fact, transaction processing takes less than 300 milliseconds. The fact that there is a captured market of about 7 million cards, with constant use, means that there is a critical mass for this payment infrastructure. More importantly, the clearing and settlement of payments (via HSBC bank, at the time of writing) taking place between Creative Star and the transportation operators, take less time (i.e., less than 24 hours), compared to the considerably longer process involving coin boxes (Leong, 2000).

Fending Off Competition from VisaCash and MasterCard

When the Octopus card was launched, Visa International and MasterCard were both involved in debit-card trials in Hong Kong. Although both Visa and MasterCard had a much larger customer base and a long track record in the credit card industry, the captured market, contactless nature, and focused application have proven to be important criteria for success. Both VisaCash and Mondex were using contact-based cards, whose transaction times would have been at least as long as it takes for the card holder to place the card into the reader. In the Hong Kong passenger transportation industry, this process was too time-consuming, especially during rush hour. More than 3 million transactions take place per day, during an 18-hour period of operation. During rush hour, however, there might be up to 1.8 million transactions occurring, averaging about 600,000 per hour. Clearly, the contactless nature of the Octopus card and its short transaction time are key reasons for its success.

Expanding to be a Micro-Payment Provider

The original intent of the Octopus card was to provide a means of payment within the passenger transportation industry. In fact, under the original governance of the Hong Kong Monetary Authority (HKMA) Banking Ordinance, Octopus transactions were to be confined to the transportation and related sectors. However, Octopus can derive up to 15% of the equivalent core transactions from non-transportation sectors. In April 2000, Creative Star Limited was granted a deposit-taking licence by the HKMA, which broadened the card's scope of use (Leong, 2000). Octopus cards can be used for payments at shops in the fast-food sector, kiosks, phone booths, and soft-drink vending machines (Rader and Maghiros, 2001). They can be reloaded not just at MTRC and KCRC train stations, but also at 7-Eleven convenience stores using dedicated devices, and even at the point of transaction if one has an account with the Dah Sing Bank or Standard Chartered Bank. For example, if there are insufficient funds on the card at the time of usage, and the holder has an appropriate credit card from the bank, then the bank will automatically transfer/upload HK$250 to the Octopus card, without any special handling charge (Dah Sing Bank, 2002). This means that passengers can bypass sales counters and add-value machines. The above reasons make the Octopus card the most mobile-friendly micropayment system in Hong Kong.

5.7.4 Future Developments in the Octopus System

The Octopus card is in a strategic position to expand its market share, including non-transportation applications. Fig. 5.11 illustrates some of the potential paths for expansion.

All-Purpose Micropayment System

One potential development path is to have the Octopus card be an organizational charge card. In some organizations, there is a need to have an internal charge card to allow employees to use company resources on a pay-per-use basis. For example, in a university, students have to pay for different services such as photocopying and recreational facilities. The Octopus card is well positioned to take on the role of a university-wide charge card, thereby enabling students to use various resources and pay via an internal Octopus system.

Transaction Services Outsourcing

The strength of the Octopus system, besides its application in the passenger-transportation industry, is its transaction-clearing system. The high daily transaction volume means Octopus's clearing system rivals that of mid-size banks in Hong Kong (Rader and Maghiros, 2001). The Octopus system, in conjunction with HSBC's Hexagon system, can settle this volume of payments within 24 hours. This very efficient clearing system can be positioned as an outsourcer for other high-volume transaction environments, enabling the Octopus clearing system to become the universal backend for major transaction establishments.

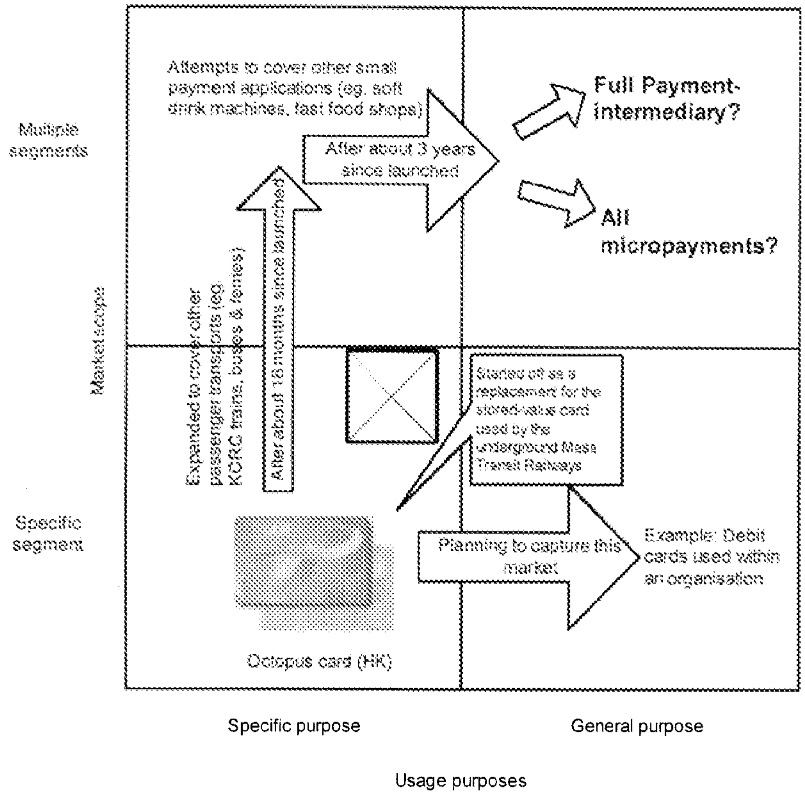

Multiple segments
Market scope
Specific segment
Attempts to cover other small payment applications (eg. soft drink machines, fast food shops)
After about 3 years since launched
Full Payment-intermediary?
All micropayments?
Expanded to cover other passenger transports (eg. KCRC trains, buses & ferries)
After about 18 months since launched
Started off as a replacement for the stored-value card used by the underground Mass Transit Railways
Planning to capture this market
Example: Debit cards used within an organisation
Octopus card (HK)
Specific purpose
General purpose
Usage purposes

Fig. 5.11 Strategic orientation of the Octopus card and its future (Source: Poon & Chau, 2001)

5.8 Summary

In this chapter, fundamentals of smart cards have been introduced, including smart card chips, readers, operating systems, and communication interface. Then, the information of Java Card has been provided. In addition, a set of smart card standards have been presented. The utilization of smart cards in implementing security and possible smart card applications have been discussed. Finally, a case study of Hong Kong's Octopus cards has been presented.

Smart cards, plastic cards with an embedded IC chip, are excellent vehicles for electronic payment and other applications, such as identification, access control, telecommunications, healthcare, and transportation. They are also crucial for implementing security. Currently, billions of smart cards are in use, and the market potential of smart cards is huge.

5.9 References

[5.1] Dah Sing Bank: http://www.dahsing.com.hk/etopup.htm

[5.2] W. Ford, M. S. Baum (2001) Secure electronic commerce (2nd ed.). Prentice-Hall, New York.

[5.3] C. Goldfinger (1998) Economics of financial applications of the smart card: a summary overview. http://www.fininter.net/Archives/fasc.htm

[5.4] M. Hendry (2001) Smart card security and applications (2nd ed.). Artech House, Boston London.

[5.5] Hong Kong Economic Times (2000) The largest eCommerce network in Hong Kong – the Octopus transaction system. September 27, 2000, IT2–IT3.

[5.6] W. Kou (1997) Networking security and standards. Kluwer, Boston Dordrecht London.

[5.7] E. Leong (2000) Octopus extends its reach. FinanceAsia.com, July 19, 2000.

[5.8] S. Poon, P. Y. K. Chau (2001) Octopus: the growing e-payment system in Hong Kong. Electronic Markets 11(2): 1–10.

[5.9] M. Rader, I. Maghiros (2001) Electronic Payment Systems Observatory Newsletter, No. 5. http://epso.jrc.es/newsletter.

[5.10] B. Schneier (2000) Secrets and lies: digital security in a networked world. Wiley, New York.

[5.11] Sun Microsystems (2002) Java Card platform security: technical white paper. http://java.sun.com/products/javacard

[5.12] The World Bank Group (2002) The Octopus system (presentation by the Hong Kong MTR Corporation). http://lnweb18.worldbank.org/External/lac.nsf/Sectors/Transport/0D8952

[5.13] U. Hansmann, M. S. Nicklous, et al. (2002) Smart card programming. Springer, Berlin Heidelberg New York.

6 Wireless Infrastructure

Weidong Kou

University of Hong Kong
Pokfulam Road, Hong Kong

6.1 Introduction

Wireless e-commerce (or mobile commerce) is projected to become a US$12.4 billion market by 2005 in Asia-Pacific, excluding Japan, according to International Data Corp (IDC). Mobile commerce applications such as mobile banking, email, wireless gaming, and stock trading already are available in the marketplace. For example, NTT DoCoMo's i-mode service in Japan, which provides email, web access, wireless banking, stock information service, flight information, online reservations, news and weather, yellow page service, fortune telling, online games, and digital content retrieval from its partners, in addition to regular cellular-phone functions. DoCoMo was formed in July 1992. It had sales of 4.6 trillion yens in fiscal 2000 year ended by March 31, 2001. It was reported that the subscriber number of the i-mode service exceeded 28 million as of October 2001. We see some countries, for example, Korea, where wireless subscription numbers exceed wired customers. A recent statistical report (October 2001) shows that China now has the largest hand-phone user base in the world, with a total of over 120 million users, or 10% penetration rate. In Hong Kong, over 5 million people out of a total of 7 million have a cellular phone. The penetration rates in European countries are also high. All this evidence shows that the growth of mobile commerce is phenomenal and its potential is huge.

Payment is essential for commerce transactions. Mobile commerce transactions also need to have payment in place. To enable the payment process in mobile commerce, we need to have a wireless infrastructure. In this chapter, we will examine various components of such a wireless infrastructure for e-payment and for mobile commerce in general, including wireless communication infrastructure, wireless and pervasive computing infrastructure including wireless and pervasive devices, wireless application protocol (WAP), and wireless security.

6.2 Wireless Communications Infrastructure

The Internet is the basis of the World Wide Web, and is the infrastructure that has taken the current form of e-commerce to center stage in the past few years. The Internet has been an interconnected computer network through cables since it was born more than thirty years ago. The rapid development of wireless technologies means the Internet going through a revolution. The Wireless Internet is emerging. Accessing information anytime and anywhere is becoming a reality. This change sets mobile commerce rolling. The essential part of the advances in wireless technologies is wireless communication infrastructure.

There are three main areas of the wireless communication infrastructure: the transmission and media access, the mobile network, and the mobile services. The transmission and medium access area covers wireless transmission technologies (such as multiplexing and modulation) and medium access technologies (such as TDMA and CDMA). The mobile network area addresses the network system architecture and protocols. The mobile services deal with voice and data services such as mobile-prepared services, mobile voice IP, and international roaming.

In wireless communication, radio transmission takes place via different frequency bands. It starts at several kilo-Hz. It can go as high as over one hundred mega Hz. As there exists interference in the radio transmission, and as radio frequencies are scare resources, the frequencies used for transmission are all regulated.

In order to ensure low interference between different senders, multiplexing schemes have been introduced in four dimensions: space, time, frequency, and code. The space-division multiplexing is a scheme to ensure that there is wide enough distance between senders to avoid interference due to radio transmissions from the different senders. The time-division multiplexing scheme allows senders to use the same frequency but at different times. The frequency division multiplexing scheme is to subdivide the frequency dimension into several non-overlapping frequency bands. Different senders are assigned to different frequency bands. The code-division multiplexing scheme is relatively new, and it resolves the interference problem by assigning senders to different codes, and the distances between these codes are wide enough to avoid the interference. As the many codes can be designed, the code-division multiplexing scheme offers much more flexibility than the space, time, and frequency division multiplexing schemes do.

In wireless networks there is a need to translate the binary bit stream into an analog signal first. This translation is referred to as digital modulation. There are three basic techniques for digital modulations: amplitude shift keying (ASK), frequency shift keying (FSK), and phase shift keying (PSK). In the ASK technique, the binary values, 1 and 0, are assigned to two different amplitudes. The FSK technique assigns the two binary values, 1 and 0, to two different frequencies. The PSK technique makes use of shifts in the phase of a signal to present the two binary values, 1 and 0, for example, shifting the phase by 180 degrees when the value of data changes. After digital modulation, wireless transmission requires an analog modulation, which is a technique to shift the center frequency of the baseband signal generated by the digital modulation up to the radio carrier frequency. There are three different analog modulation schemes: amplitude modulation (AM), frequency modulation (FM), and phase modulation (PM). These modulation schemes have been widely used, for example, AM and FM radios.

In wireless communication networks, how to allow a mobile-phone user to access the wireless networks is the problem that the medium-access control technology is meant to resolve. The typical algorithms for the medium-access control include space division multiple access (SDMA), frequency division multiple access (FDMA), time division multiple access (TDMA), and code division multiple access (CDMA). SDMA is a technology for allocating a separate space to mobile users in the wireless network. FDMA deals with allocating frequencies to transmission channels according to the frequency division multiplexing scheme. TDMA is to allocate certain time slots for wireless communications. CDMA is to separate different users through the codes used in the code division multiplexing.

The wireless communication systems are cellular, that is, they are designed as a network of cells. In the center of each cell, there is a base transceiver station (or, simply, a base station) that comprises radio equipment for transmitting and receiving radio signals, including antennas, signal processing, and amplifiers. Each cell has a coverage area. A mobile-phone user may move from one cell to another. This movement is called roaming. The process of switching such a user from one cell to another while the user is engaging a call is referred to as a handoff or handover.

The popular wireless communication systems include global system for mobile communications (GSM), general packet radio service (GPRS), and code division multiple access (CDMA) systems. Among them, the most popular one is the GSM system that has been used in more than 130 countries worldwide, including most countries in Europe and Asia, excluding Japan. GSM has initially been developed and deployed in Europe to provide a mobile phone system that offers the roaming service to users throughout Europe. Now, GSM supports the integration of different voice and data services and permits an easy system upgrade to higher data rates. GPRS was designed for data services by providing packet-mode transfer for applications, such as web requests and responses, and it promises to provide users with a high-capacity connection to the Internet. CDMA is a digital spread-spectrum system initially developed by a US-based company called Qualcomm, and it has been standardized by the Telecommunications Industry Association (TIA). The CDMA standard is known as Cellular IS-95.

An example of the wireless communication systems is shown in Fig. 6.1. A mobile station (MS) can be held by either a pedestrian or a motion vehicle. The MS communicates with a base transceiver station (BTS). The BTS is managed by a base station controller (BSC) which is connected to either a mobile service switching center (MSC) or a gateway mobile service switching center (GMSC). The MS information is usually stored in a home location register. The MS information can be static and dynamic. The static information includes the mobile subscriber ISDN number, subscribed services, and the authentication key information. The dynamic information includes the current location area of the MS. When an MS leaves the current location area, to localize a user in the worldwide network, this information needs to be stored and updated in a very dynamic database, which is called visitor location register (VLR). The VLR is associated with a MSC, storing the information of the MS who is currently visiting the location area associated to the MSC. The GMSC connects to the public switched telephone network (PSTN). The operation and maintenance center (OMC) monitors traffic and provides management functions such as subscriber and security management, and accounting and billing. The authentication center (AUC) is to protect mobile user identity and data transmission.

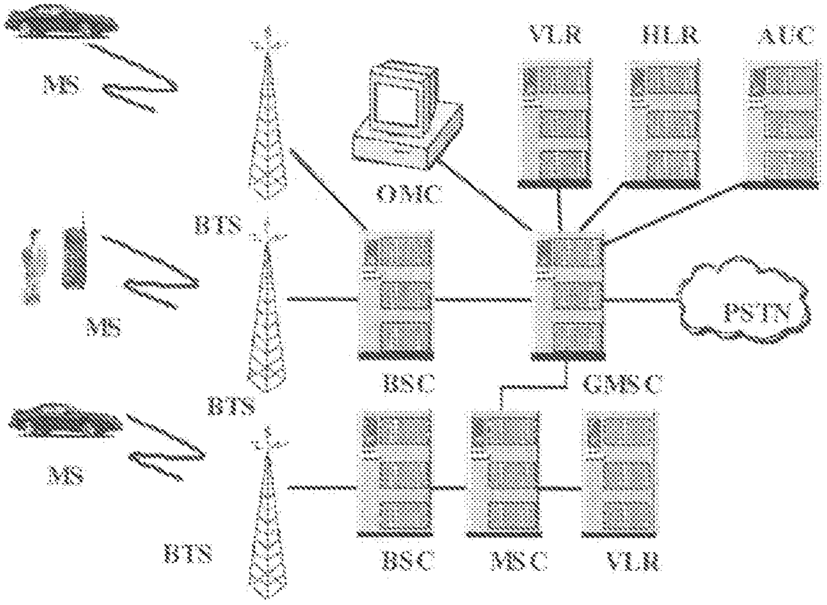


Fig. 6.1 An example of the wireless communication systems

By having the wireless communication systems in place, various mobile services have been provided to the mobile users, such as mobile voice and data services. The mobile data services include messaging services and wireless web services. The examples of the mobile messaging services are messaging through short message service (SMS), cell broadcast service (CBS), and unstructured supplementary services data (USSD). The examples of wireless web services include mobile commerce and mobile payment services.

6.3 Wireless Computing Infrastructure

The evolution of computing infrastructure has gone from the client-server infrastructure model, to the network computing infrastructure model. It is now moving toward the wireless computing infrastructure model. In the client-server computing infrastructure model shown in Fig. 6.2, many clients connect to a server. The server is a center of computing, to provide a variety of services that clients request. The client machines were equipped with dedicated client software. This model was very effective before the Internet was adopted.

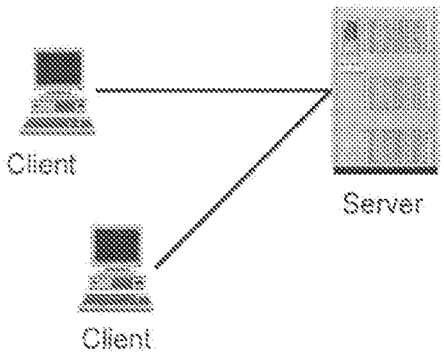


Fig. 6.2 Client-server computing infrastructure model

The Internet has changed the computing infrastructure. In the 1990s, many leading players in the computing industry looked into network computing. In the network computing infrastructure model shown in Fig. 6.3, the clients connect to the Internet, and the Internet connects a variety of servers. The client machines do not need special client software. Only standard Internet browser such as Netscape or Microsoft Internet Explorer is needed. This saves a huge effort for companies to develop different client software. Through the Internet, clients can access much wider applications than they used to. The network computing infrastructure model coupled with the revolutionized Internet, has indeed changed the computing industry, corporate IT infrastructure, and people's daily lives.

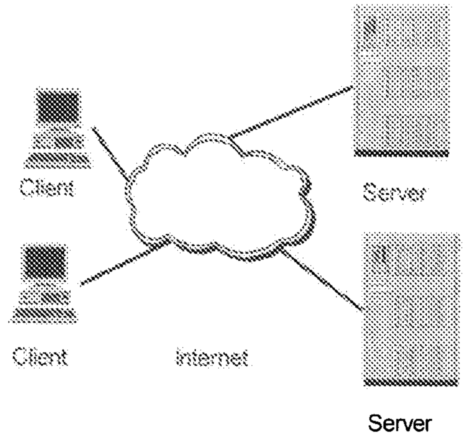


Fig. 6.3 Network computing infrastructure model

In the late 1990s, wireless communication was rapidly developed. Cellular phones and a variety of other wireless devices have become popular. The wireless communication infrastructure together with Internet technology makes another evolution step possible, resulting in the wireless computing infrastructure model shown in Fig. 6.4.

In addition to wireless communication infrastructure, the wireless computing-infrastructure model includes flexible and mobile devices, such as personal digital assistants (PDAs), mobile phones, pagers, hand-held organizers, and home-entertainment systems. These wireless devices connect to the Internet and provide quick access to many wireless applications. With enhanced security, electronic commerce transactions can also be conducted through these wireless devices. The differences between the network computing infrastructure model and the wireless computing infrastructure model are:

- In the wireless-computing-infrastructure model, client machines are no longer only the desktop PCs or laptop computers as those in the network computing infrastructure model. They can be any hand-held device with wireless communication capability.

- In the wireless-computing-infrastructure model, communication between clients and servers are no longer through wired lines as the case of the network computing infrastructure model. The communication is carried out through a wireless network and the Internet.
- In the wireless-computing-infrastructure model, there are two sets of communication protocols, one set is wireless protocols and the other set is the wired Internet protocols. This is different from the network computing infrastructure model in which, there is only wired Internet protocols.

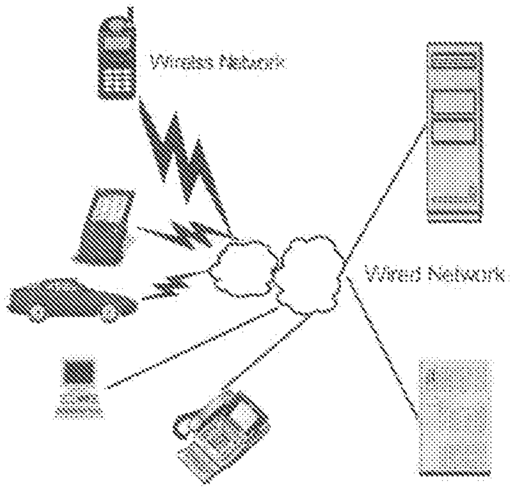


Fig. 6.4 Wireless computing infrastructure model

The challenges for the wireless-computing-infrastructure model include powerful wireless e-commerce applications for massive wireless users that are still to be developed and deployed, wireless-device capability-management systems, and personalization for different users, and the associated user management by the wireless service providers. Given the limitations that wireless devices have, the server software must be highly scalable and flexible. Mobility will also make the wireless applications be more interesting and challenging.

In the wireless-computing-infrastructure model, to make mobile commerce transactions successful, there are three very important principles: security, connectivity, and simplicity. The importance of the security model is obvious, as without the proper security protection of the consumer's financial account information and other private information, mobile commerce is not going to succeed. The connectivity loss during the mobile-commerce transaction will create the trustworthiness problem on mobile commerce for consumers. People will not accept that their mobile-commerce transactions (such as mobile payments) are aborted due to a broken connection. Given a limited capacity of a mobile device and the reliability of the wireless connection compared to that of the wired connection, it is easy to see that simplicity and reliability are important for completing a mobile-commerce transaction instantly.

6.4 Wireless Application Protocol

6.4.1 WAP Overview

The wireless application protocol (WAP) is a suite of emerging standards to enable mobile Internet applications. The WAP standards have been created as a result of the WAP Forum that was formed in June 1997 by Ericsson, Motorola, and Nokia. The WAP Forum is designed to assist the convergence of two fast-growing network technologies, namely, wireless communications and the Internet. The convergence is based on rapidly increasing numbers of mobile-phone users and the dramatic affect of e-business over the Internet. The combination of these two technologies will have a big impact on current e-business practice, and it will create huge market potential.

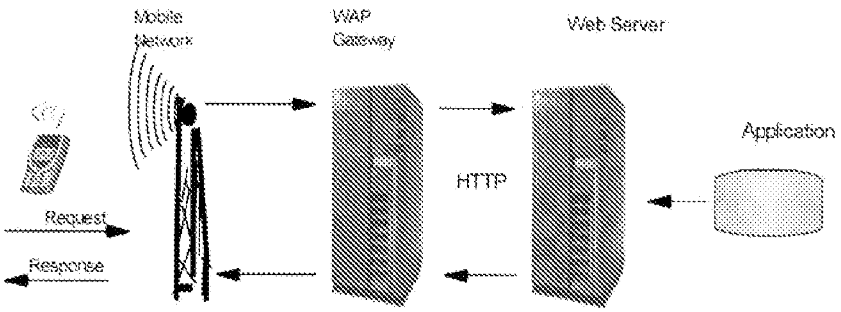


Fig. 6.5 The WAP architecture

The WAP standards consist of a variety of architecture components, including an application environment, scripting and markup languages, network protocols, and security features. These components and features together define how wireless data handsets communicate over the wireless network and how content and services are delivered. With the WAP standards, a wireless data handset can establish a connection to a WAP compliant wireless infrastructure, request and receive the content and services, and present the content and services to the end user. This WAP-compliant wireless infrastructure may include the handset, the server side infrastructure, such as the proxy server (WAP gateway), the web server, the application server, and the network operator (telecommunication company). The WAP architecture is shown in Fig. 6.5.

The WAP architecture can also be presented through the WAP protocol stack shown in Fig. 6.6. The WAP protocol stack covers the complete picture from bearers to applications. The bearers are the various wireless networks that WAP currently supports. The transport layer is an interface common to the underlying wireless network, and it provides a constant service to the upper layers in the WAP stack, such that the bearer services are transparent to the upper layers. In other words, with the transport layer, the specific network characteristics can be masked. The security layer provides security for the transport layer, based on the industry standard protocol, the transport layer security (TLS) protocol. The transaction layer provides a lightweight transaction-oriented protocol for mobile thin clients. The session layer provides the application layer with the capability to select connection-oriented or connectionless services. The application layer deals with a general-purpose environment for applications.

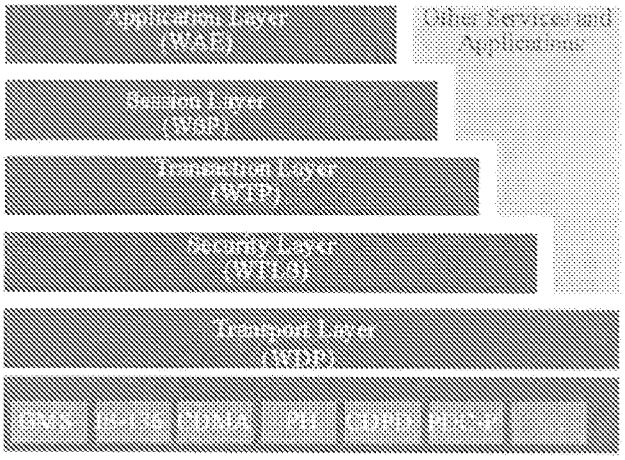


Fig. 6.6 The WAP protocol stack

The WAP protocols in Fig. 6.6 include wireless application environment (WAE), wireless session protocol (WSP), wireless transaction protocol (WTP), wireless transport layer security (WTLS), and wireless datagram protocol (WDP).

In the following sections, we discuss these protocols with a focus on WAE by providing more detailed information.

6.4.2 Wireless Application Environment

The wireless application environment (WAE) consists of a set of standards that collectively define a group of formats for wireless applications and downloadable content. WAE specifies an application framework for wireless devices, such as cellular phones, pagers, and PDAs. WAE has two logical layers, namely, user-agent layer and format-and-service layer. The components of the user-agent layer include browsers, phone books, message editors, and other items on the user device side, such as wireless telephony application (WTA) agent. The components of the format-and-service layer include common elements and formats accessible to the user agents, such as WML, WMLScript, and WAP binary XML content format (WBXML).

A WAP microbrowser has the following capabilities:

- Submission requests to the server
- Reception of responses from the server
- Converting and parsing the data
- Interpreters from WML and WMLScript files
- Ability to interact with the appropriate WAP layer
- Local cache and variable management
- Wireless session protocol processing
- Effective management of local hardware resources, such as RAM, ROM, small screen, and input and output

Wireless Markup Language

Wireless markup language (WML) is a language based on the extensible markup language (XML). WML is optimized for small screens and limited memory capacity, and it is for content intended for lightweight, wireless devices such as mobile phones and personal digital assistants (PDAs).

A WML document is called deck. A page of a WML document is called card. A deck consists of one or more cards. Each deck is identified by an individual URL address, similar to an HTML page. A WML deck requires a browser that will format the deck for the benefit of the user. The browser determines the final shape of the deck. Sometimes, people use the analogy of HTML to explain WML. In the analogy, a WML deck corresponds to an HTML page. However, there are differences between a WML deck and an HTML page. While each HTML file is a single viewable page, a WML deck may contain multiple cards, each of which is a separate viewable entity. WML files are stored as static text files on a server. During the transmission from the server to the browser, the WML files are encoded in binary format by the wireless connection gateway, and then sent to the browser. This is also different from HTML, where there is no need for such an encoding process.

WML contains commands for navigating in decks. Each WML command has two core attributes, namely, id and class. The id is the attribute for an individual name to the elements inside a deck, while the class is the attribute that links the element to one or several groups. A WML deck, at its most basic level, is constructed from a set of elements. Elements are identified by tags, which are enclosed in angle brackets. Each element must include a start tag (<el_tag>) and an end tag (</el_tag>). The content is included between the start and end tags. An empty element that has no content can be abbreviated by a single tag (<el_tag/>).

Because WML is based on the XML language, a WML document must follow the XML rule to contain the XML specified document type definition (DTD) at the beginning of the WML code, which is referred to as deck header or document prolog, as follows:

```
<?xml version=”1.0”?>
<!DOCTYPE wml PUBLIC “-//WAPFORUM//DTD WML 1.1//EN”
http://www.wapforum.org/DTD/wml_1.1.xml>
```

A deck is defined by the <wml> and </wml> tags that are required in every WML document. Within a deck, each card is defined by the <card> and </card> tags. Both <wml>...</wml> and <card>...</card> are formatting commands. The <wml>...</wml> commands summarize the deck. The <card>...</card> commands summarize the text, images, input fields, and any other objects of a card in the deck.

Cards are the basic units of WML, defining an interaction between mobile device and the user. Each card may contain three different groups of elements: content elements (such as text, tables, and images), tasks and events (such as <onevent>, <timer>, and <do>), and data entry (such as <input> and <select>).

WMLScript

WMLScript is a simple scripting language based on ECMAScript (ECMA-262 standard) with modifications to better support low-bandwidth communication and thin clients. WMLScript is part of the WAP application layer.

WMLScript complements the WML by adding simple formatting capabilities to make the user interfaces more readable, for example, the capabilities of checking the validity of user input and generating messages and dialog locally to reduce the need for expensive round-trip to show alerts. These capabilities are not supported by WML as the content of WML is static. WMLScript provides programmable functionality that can be used over narrowband communication links in clients with limited capabilities. With WMLScript, more advanced user interface functions can be supported and intelligence can be added to the client. WMLScript also provides access to the device and its peripheral functionality and reduces the amount of bandwidth that is needed for sending data back forth between the server and the client.

WMLScript is similar to JavaScript. For example, WMLScript includes a number of operators such as assignment and arithmetic operators, which are similar to those in JavaScript. However, there are major differences between WMLScript and JavaScript. First, WML contains references to the URL address of a WMLScript function, whereas JavaScript functions are normally embedded in the HTML code. Second, WMLScript must be compiled into binary WMLScript code prior to its execution in a WAP device, while there is no such requirement for JavaScript.

Although WMLScript is based on ECMAScript as we mentioned before, there are differences between WMLScript and ECMAScript. First, like JavaScript, ECMAScript is not encoded in a binary form while WMLScript has to be. Second, to form WMLScript, many advanced features of the ECMAScript language have been dropped to make WMLScript smaller, and easier to compile into binary WMLScript code.

WMLScript syntactically resembles C language. It has basic types, variables, expressions, and statements. Unlike C, WMLScript cannot be used to write standalone applications. There is no built-in support for reading and writing files. Because it is an interpreted language, scripts or functions can run only in the presence of an interpreter, which is supplied as part of the WAP user agent. WMLScript is a weakly typed and object-based language, in which variables must be declared before they can be used in expression. In WMLScript, there is no main program or routine. Functions are created to perform specific tasks and they are invoked through a WML call. When a WMLScript function is invoked, the WAP gateway accesses the source code, compiles it into binary WMLScript code, and then sends the execution function to the WAP user agent. WMLScript code is written in normal text files with the file extension "wmls."

Each WMLScript file contains at least one function. Each function is composed of statements that perform the appropriate processing. The structure of a WMLScript function is as follows:

```
extern function function_xyz (parameter list)
{ // start of the statements
      statement_1;
      statement_2;
```

```
        statement_n;
    }// end of the statements
```

With this structure and the file extension "xmls," a simple WMLScript example to set a first day of the week, which is included in the file named "setday.xmls," is listed as follows:

```
    extern function SetDay(givenDay)
    {
        if (givenDay > 0 && givenDay <= 7) {
                var newDay =  givenDay;
        }
        else {
                newDay = 1;
        }
        return newDay;
    }
```

To invoke a WMLScript function, a reference to the WMLScript function must be included in a WML document. The call will be routed from the WAP browser through the WAP gateway to the server. The server then sends the binary WMLScript code to the WAP browser. The WAP browser has an interpreter that is able to execute WMLScript programs in their binary format. Using our example, the reference to the WMLScript can be as simple as follows:

```
    <do type="ACCEPT" label="Set Day">
    <!--Calling the WMLScript function: -->
        <go href="setday.xmls#SetDay($(givenDay))"/>
    </do>
```

Wireless Telephony Application Interface and Wireless Telephony Applications

One of the major mobile services is voice. How can we set up a call or receive an incoming call using a WAP enabled mobile device? This is the problem that Wireless Telephony Application Interface (WTAI) addresses. WTAI is designed to allow wireless network operators access the telephony features of WAP device. Through either a WML deck/card or WMLScript, using the WTAI function libraries, a mobile phone call can be set up and an incoming call can be received. In addition, text messages can be sent or received, and phonebook entries can be manipulated on the WAP device.

Wireless telephony applications (WTA) is a collection of telephony-specific extensions for call and feature control mechanisms that make advanced mobile network services available to the mobile users. It provides a bridge between wireless telephony and data. The WTA applications can use the privileged WTAI.

From the architecture point of view, a WTA server communicates with the WAP gateway to deliver and manage telephony services; on the client side, there is a WTA framework which has three components as follows:

1. User agent: This agent supports the WTAI libraries, renders WML, and executes WMLScripts.
2. Repository: It provides persistent client-side storage for wireless telephony applications.
3. Event Handling: This deals with incoming-call and call-connected events to be delivered to a wireless telephony application for processing, which may also invoke WMLScript library interfaces to initiate and control telephony operations.

Wireless telephony supports in WAP make WAP suitable for creating mobile applications through voice services. The compact form, encryption, and error handling capabilities of WAP enable critical wireless payment transactions.

WBXML

WAP Binary XML Content Format (WBXML) is defined in the Binary XML Content Format Specification in the WAP standard set. This format is a compact binary representation of the XML. The main purpose is to reduce the transmission size of XML documents on narrowband communication channels.

A binary XML document is composed of a sequence of elements and each element may have zero or more attributes. The element structure of XML is preserved while the format encodes the parsed physical form of an XML document. This allows user agents to skip elements and data that are not understood. In terms of encoding, a tokenized structure is used to encode an XML document. The network byte order is big-endian, that is, the most significant byte is transmitted first. Within a byte, bit-order is also big-endian, namely, the most significant bit first.

6.4.3 Wireless Session Protocol

The Wireless session protocol (WSP) is a protocol family in the WAP architecture, which provides the WAP Application Layer with a consistent interface for session services. WSP establishes a session between the client and the WAP gateway to provide content transfer: the client makes a request, and then the server answers with a reply through the WAP gateway. WSP supports the efficient operation of a WAP micro-browser running on the client device with limited capacity and communicating over a low-bandwidth wireless network. The WSP browsing applications are based on the HTTP 1.1 standard, and incorporated with additional features that are not included in the HTTP protocol, for example, the connection to the server shall not be lost when a mobile user is moving, resulting in a change from one base station to another. The other additional features that WSP supports include:

- **Binary encoding**:
  Given the low bandwidth of the wireless network, the efficient binary encoding of the content to be transferred is necessary for mobile Internet applications.

- **Data push functionality**:
  Data push functionality is not supported in the HTTP protocol. A push is what is performed when a WSP server transfers the data to a mobile client without a preceding request from the client. WSP supports three push mechanisms for data transfer, namely, a confirmed data push within an existing session context, a non-confirmed data push within an existing session context, and a non-confirmed data push without an existing session context.

- **Capability negotiation**:
  Mobile clients and servers can negotiate various parameters for the session establishments, for example, maximum outstanding requests and protocol options.

- **Session suspend/resume**:
  It allows a mobile user to switch off and on the mobile device and to continue operation at the exact point where the device was switched off.

WSP offers two different services, namely, the connection-oriented service and the connectionless service. The connection-oriented service has the full capabilities of WSP. It operates on top of the wireless transaction protocol (WTP), supports session establishment, method invocation, push messages, suspend, resume and session termination. The connectionless service is suitable for these situations where high reliability is not required or the overhead of session establishment and release can be avoided. It supports only basic request-reply and push, and does not rely on WTP.

6.4.4 Wireless Transaction Protocol

The wireless transaction protocol operates on top of a secure or insecure datagram service. WTP introduces the notion of a transaction that is defined as a request with its response. This transaction model is well suited for web content requests and responses. It does not handle stream-based applications (such as telnet) well.

WTP is responsible for delivering the improved reliability over datagram service between the mobile device and the server by transmitting acknowledge messages to confirm the receipt of data and by retransmitting data that has not been acknowledged within a suitable timeout period.

WTP supports an abort function through a primitive error handling. If an error occurs, such as the connection being broken down, the transaction is aborted.

WTP is message-oriented and it provides three different types of transaction services, namely, unreliable one-way, reliable one-way, and reliable two-way transactions. The transaction type is set by the initiator and it is contained in the service request message sent to the responder. The unreliable one-way transactions are stateless and cannot be aborted. The responder does not acknowledge the message from the initiator. The reliable one-way transactions provide a reliable datagram service that enables the applications to provide reliable push service. The reliable two-way transactions provide the reliable request/response transaction services.

6.4.5 Wireless Transport Layer Security

The wireless transport layer security (WTLS) protocol is a security protocol based on the *transport layer security protocol* (TLS) [6.10] (see Section 6.7). TLS is a derivative of the *secure sockets layer* (SSL), a widely used security protocol for Internet applications and payment over the Internet (for more information about SSL, see Section 11.6 of Chapter 11). WTLS has been optimized for the wireless communication environment. It operates above the transport protocol layer.

WTLS is flexible due to its modular design. Depending on the required security level, it can be decided whether WTLS is used or not. WTLS provides data integrity, data confidentiality, authentication, and denial-of-service protection. The data integrity is to ensure that data sent between a mobile station and a wireless application server is unchanged and uncorrupted. The data confidentiality is to ensure that data transmitted between the mobile station and the wireless application server is private to the sender and the receiver, and it is not going to be understood by any hackers. The authentication is to check the identity of the mobile station and the wireless application server. The denial-of-service protection is to prevent the upper protocol layers from the denial-of-service attacks by detecting and rejecting data that is replayed or not successfully verified.

6.4.6 Wireless Datagram Protocol

The wireless datagram protocol (WDP) in the WAP architecture specifies how different existing bearer services should be used to provide a consistent service to the upper layers. WDP is used to hide the differences between the underlying bearer networks. WDP layer operates above the bearer services and provides a consistent interface to the WTLS layer.

Different bearers have different characteristics. The bearer services include short message, circuit-switched data and packet data services. Since WAP is designed to operate over the bearer services, and since the bearers offer different types of quality of service with respect to throughput, error rate, and delays, the WDP is designed to adapt the transport layer to specific features of the underlying bearers. The adaptation results in a family of protocols in the WDP layer, dealing with each supported bearer network protocol. When a message is transmitted through WAP stack, depending on the underlying bearer network, a different WDP protocol may be used. For example, for an IP bearer, the user datagram protocol (UDP) must be adopted as the WDP protocol, and for a short message service (SMS) bearer, the use of the source and destination port numbers becomes mandatory.

6.4.7 WAP Gateway

A WAP gateway as shown in Fig. 6.7 is a proxy server that sits between the mobile network and the Internet. The purpose of this proxy server is to translate between HTTP and WSP. The reason for the translation is that the web server connected to the Internet only understands the HTTP protocol while the WAP-enabled mobile client only understands the WSP. The WAP gateway also converts a HTML file into a WML document that is designed for small-screen devices. In addition, the WAP gateway compiles the WML page into binary WML which is more suitable for the mobile client. The WAP gateway is transparent to both the mobile client and the web server.

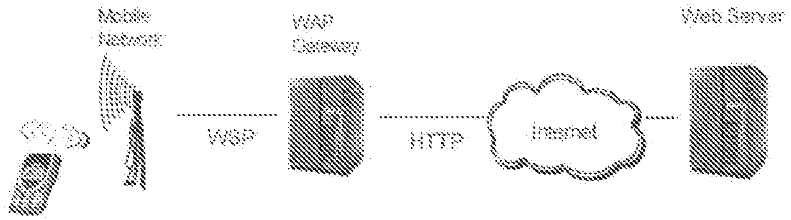


Fig. 6.7 WAP gateway

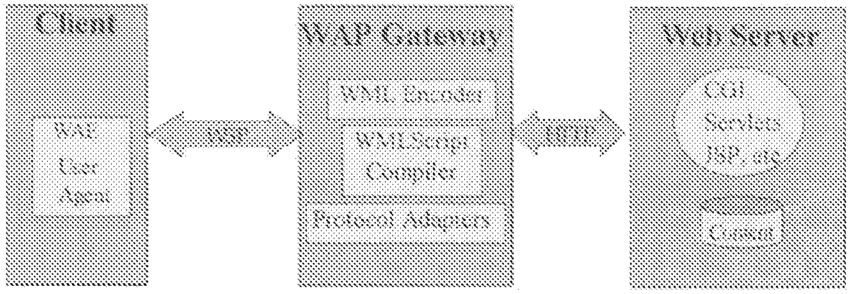


Fig. 6.8 WAP model

Fig. 6.8 shows the WAP model using the WAP gateway. How the WAP gateway processes of a typical request for a document can be illustrated as follows:

1. The mobile user makes a request for a specific document using the WAP phone.
2. The WAE user agent on the WAP phone encodes the request and sends it to the WAP gateway.
3. The WAP gateway decodes and parses the encoded request.
4. The WAP gateway sends a HTTP request for the document.
5. The web server answers with a response to the WAP gateway.
6. The WAP gateway parses and encodes the response.
7. If the content-type is WML then the gateway compiles it into binary WML.
8. The WAP gateway sends the encoded response to the WAP phone.
9. The WAE user agent on the WAP phone interprets and presents the document to the mobile user.

6.5 Wireless Security

Wireless security is becoming more and more important as transaction-based mobile commerce applications (such as mobile payment, banking, and buying stock via cellular phones or handheld devices) take off.

The basic security needs for mobile commerce are similar to that for electronic commerce over the wired Internet, such as authentication, confidentiality, nonrepudiation, and data integrity. However, implementing them in the wireless world is more difficult than it is to implement them in the wired world. This is simply because the limitations that wireless have, including limited bandwidth, high latency, and unstable connections. In addition, limited battery power and limited processing power that the wireless devices have also make the sophisticated security algorithms difficult to run on these devices.

As we discussed in the previous section, WAP does specify an SSL like security protocol, namely, wireless transport layer security (WTLS). However, there are some drawbacks in WTLS. First, WTLS only provides security protection from the mobile client to the WAP gateway where the wireless communication ends. In the wired Internet environment, when a web client (web browser) starts an SSL session with web server, the web client and web server are communicated directly, and the end-to-end security protection is provided through the SSL session. This means when one sends a credit card number over SSL, only the receiving web server will be able to receive it. The situation is different in the WTLS. The credit card number will be securely protected between the mobile device and the WAP gateway. It will be in the clear form at the WAP gateway. Then, an SSL session will be established between the WAP gateway and the Web server for securely transmitting the credit card number over the Internet. This means that there is no end-to-end security protection for the wireless transactions since there is a potential security hole in the WAP gateway. Second, the CCITT X509 certificate is too large for the mobile phones, and the limitations of the processing power and battery for the wireless devices make it difficult to perform the sophisticated computation of the public-key encryption. In summary, WAP security has two issues: (1) there is no end-to-end security protection, and (2) there is a lack of certificates for mobile devices.

People are currently addressing these two security issues. As a result, simplified certificates have been defined for mobile devices. The research on how to use currently available mobile devices to perform the computation of public-key encryption is ongoing. For example, elliptic curve cryptography (ECC) requires far fewer resources and it looks very promising for wide deployment to CPU-starved wireless devices.

6.6 Summary

The convergence of wireless technologies and the e-commerce over the Internet lead to emerging and fast growth of mobile commerce. As the result, mobile commerce and mobile payment have attracted more and more attention of the academic researchers and business leaders. Being able to conduct e-commerce and make payment anywhere and anytime is becoming reality. However, because of the limitations that wireless have, conducting e-commerce and making payment in the wireless world is more difficult than in the wired world. Understanding the wireless infrastructure that the wireless applications rely on is important for developing and deploying such applications.

In this chapter, we discussed the wireless infrastructure for mobile payment and fore mobile commerce in general, including wireless communication infrastructure, wireless computing infrastructure, wireless application protocol, and wireless security.

6.7 Appendix

Overview of the Transport Layer Security

The transport layer security (TLS) [6.10] is a protocol that provides privacy and data integrity between two communicating applications. The TLS is application protocol independent, that is, higher-level protocols can layer on top of the TLS protocol transparently. The TLS protocol is composed of two layers:

- **TLS record protocol**: This protocol provides connection security and is used for encapsulation of various higher-level protocols, such as the TLS handshake protocol to be discussed below. It has the following two basic properties.
  - The connection is private. Data encryption is used for ensuring the communication privacy, and is based on symmetric cryptographic algorithms, such as DES or RC4. The keys for symmetric encryption are generated uniquely for each connection and are based on a secret negotiated by another protocol (e.g., the TLS handshake protocol). The record protocol can also be used without encryption.
  - The connection is reliable. A message integrity check based on a keyed MAC is used for protecting message transport. Secure hash functions, such as SHA and MD5, are used for MAC computations. In the case of that another protocol uses the record protocol and negotiates security parameters, the record protocol can operate without a MAC.
- **TLS handshake protocol**: This protocol allows the server and client to authenticate each other, and negotiate an encryption algorithm and cryptographic keys. It has the following three basic properties.
  - The authentication between the server and client can be based on a public-key cryptographic algorithm, such as RSA or DSS. Although

the authentication can be mutual, the mutual authentication is optional. Generally speaking, one-way authentication is required.

- It is secure for the negotiation of a shared secret between the server and client.
- The negotiation is reliable.

Because the TSL is a derivative of SSL, the actual handshake exchanges are similar to that of SSL. The descriptions of the main SSL exchanges can be found in Section 11.6 of Chapter 11.

6.8 References

[6.1] WAP. http://www.ini.cmu.edu/netbil.

[6.2] Wireless Application Protocol Forum Ltd. (1999) Official wireless application protocol. Wiley, New York.

[6.3] S. Mann, S. Sbihli (2000) The wireless application protocol. Wiley, New York.

[6.4] S. Singhal, et al. (2001) The wireless application protocol. Addison-Wesley, New York.

[6.5] J. Schiller (2000) Mobile communications. Addison-Wesley, New York.

[6.6] U. Hansmann, et al. (2001) Pervasive computing handbook. Springer, Berlin Heidelberg New York.

[6.7] C. Sharma (2001) Wireless Internet enterprise applications. Wiley, New York.

[6.8] Y. B. Lin, I. Chlamtac (2001) Wireless and mobile network architectures. Wiley, New York.

[6.9] A. Dornan (2001) The essential guide to wireless communications applications. Prentice-Hall, New York.

[6.10] T. Dierks, C. Allen (1999) The TLS protocol version 1.0. http://www.ietf.org/rfc/rfc2246.txt.

7 Payment Agents

Amitabha Das

School of Computer Engineering
Nanyang Technological University, Singapore

7.1 Introduction

In a broad sense, a software agent is a computer program that acts autonomously on behalf of a person or organization. Software-agent technology seems able to provide attractive solutions in the field of electronic commerce. An agent-based architecture for electronic commerce allows the creation of a virtual marketplace in which a number of autonomous or semi-autonomous agents trade goods and services. The introduction of software agents acting on behalf of end-consumers could reduce the effort required from users when conducting electronic commerce transactions, by automating a variety of activities. The personalized, continuously running autonomous nature of agents makes them well suited for mediating consumer behavior with respect to information filtering and retrieval, personalized evaluations, complex coordination, and time-based interactions. Agents are able to examine a large number of products before making a decision to buy or sell. This not only eliminates the need to manually collect information about products but also allows the negotiation of an optimal deal with the various sellers of a good.

7.1.1 Agent-Based Electronic Commerce Systems

During the hay day of the dotcoms, several agent-based e-commerce systems came into existence, e.g., PersonaLogic, Firefly, BargainFinder, and Jango. All of them disappeared within a short while. PersonaLogic allowed users to specify product features and used a constraint satisfaction algorithm to filter through the product space to retrieve an ordered set of products. Firefly used an automated collaborative filtering method to rate and recommend products to shoppers. BargainFinder and Jango were systems that could take a product name as the input, obtain price information from other websites, and perform a price comparison.

More advanced systems in which buyer and seller agents cooperate to constitute a virtual market have also been developed in academic institutions (e.g., [7.1-7.6]). Among them Kasbah [7.1] is a multi-agent system where agents filter through ads on behalf of their owners and find those that their users might be interested in. The agents then proceed to negotiate to buy and sell items. MAGMA [7.6] is a prototype of a virtual marketplace system, which consists of multiple trader agents, an advertising server, and a bank. Trader agents are responsible for buying and selling goods. They also handle the price negotiations. Advertising server provides a classified advertisement service that includes search and retrieval of ads by category. Bank provides a set of basic banking services that includes checking accounts, lines of credit, and electronic cash.

7.1.2 Use of Agents for Payment

In none of the agent-based e-commerce systems discussed above are the agents used for executing the actual transaction involving transfer of money. Besides, in all the above systems, the agents are immobile. They do not support mobile users, and for activities that require a large number of interactions between remote agents they leave much to be optimized.

However, when the system is based only on static agents that reside in sites controlled entirely by their owners, there is no real reason for not allowing the agents to execute the payment operations. The security concerns in this case are not different from those when the payment involves manual intervention by the transacting parties. In spite of that, so far, the introduction of a system in which the agents carry out autonomously the entire process of e-commerce transaction starting from information gathering to the completion of the transaction has not materialized. The real reason for this is perhaps the lack of confidence in the competence of the agents to take decisions that are intelligent enough.

The security concerns, however, change drastically when it comes to the introduction of *mobile agents*. A mobile agent is a program that represents a user and can migrate autonomously from node to node in a computer network to perform some computation on behalf of the user. It is not bound to the system where it begins execution. It can suspend its execution at an arbitrary point, transport itself to another node, and resume execution there.

The mobile-agent paradigm offers several advantages compared to traditional approaches, such as a reduction in communication costs, better support of asynchronous interactions, enhanced flexibility in the process of software distribution, and the offer of increased performance and robustness. The use of mobile agents has been particularly promising in the fields of information retrieval, network management, electronic commerce, and mobile computing.

Information-retrieval applications often download and process large amounts of information from the server over the network while generating a comparatively small amount of result data. This can be supported much more efficiently if a mobile agent representing a query moves to the server where the data are actually stored, rather than having to move all of the data across the network for filtering. Then vendors can set up online shops with products or services for sale. Mobile agents can help customers locate the best offerings, can negotiate deals, and can even conclude transactions on behalf of their owners.

Finally, an important application of mobile agents concerns mobile computing. A portable computer's network connectivity is often achieved through low-bandwidth wireless links, hence it is likely to be slow. Besides, to minimize power consumption and transmission costs, users will not want to remain online while some complicated query is handled on their behalf by the fixed computing resources. Mobile agents offer a promising way of achieving this: users simply submit mobile agents that embody their queries and log off, waiting for the agents to deposit their results, ready to be picked up at a later time.

7.2 Security Implications of Mobile-Agent-Based Systems

The fact that mobile agents can and do execute in hosts other than the ones controlled by their owners gives rise to a number of security concerns that do not exist in the case of static agents. The most important and the most difficult-to-handle security threat arises from the fact that the third-party host has complete access and observability of the code of the mobile agents. As a result, it is extremely easy for a malicious host to either spy on confidential information, or tamper with the execution of the mobile agents.

This appears to be a severe limitation on the applicability of mobile agents in tasks involving confidentiality or security, such as electronic payment operation. However, a number of possible remedies have been suggested to overcome this problem. We will briefly examine these before we embark on the task of designing a secure payment protocol for mobile agents in untrusted host environments.

7.3 Security Techniques Protecting Mobile Agents

Methods that protect an agent against attacks can be categorized into those that prevent attacks and those which detect attacks. The detection methods use cryptographic and other techniques to detect tampering with the code and/or data carried by the mobile agent. The detection techniques are useless in preventing an attack, such as the theft of confidential information, and only serve to help in post mortem analysis. In the context of payment protocols, such methods can come in handy in detecting whether the payment has been redirected maliciously to an unintended recipient.

On the other hand, the prevention techniques are more relevant in thwarting attacks. Some of the prevention techniques proposed in the literature are *sliding encryption* [7.7], trail obscuring, code obfuscation [7.8], and computing with an encrypted function [7.9-7.10]. These are briefly discussed below.

7.3.1 Methods for Protecting Mobile Agents

Sliding Encryption

A mobile agent uses this technique [7.7] to encrypt acquired data by using a public key. The key is public, so theft is not an issue. Decryption can only be performed with the corresponding private key. The mobile agent uses sliding encryption to hide what it is carrying, so potentially malicious hosts that the mobile agent visits cannot steal any data.

This technique is applicable when mobile agents gather information in small chunks from multiple sources and it is necessary to prevent any host other than the source of a given piece of information from seeing it. As an example, consider a scenario where a mobile agent is collecting product information from multiple vendor sites and accumulating the information in its buffer. When it visits a host, the host can potentially see all the information the mobile agent is carrying and can possibly tamper with it for its own commercial gain or to affect the operations of the predecessor nodes.

A straightforward use of public key cryptography can result in substantial storage overhead for an agent. As an example, suppose that the agent is required to collect 4 bytes of data from each of 1024 different sites. If it uses a 128-byte public key, then it must encrypt the 4 bytes of data collected from a node into at least a 128-byte ciphertext before it moves on to a subsequent node. Consequently, in the end, the agent must have the capacity to carry 128 Kbytes of ciphertext which contains only 4 Kbytes worth of plaintext.

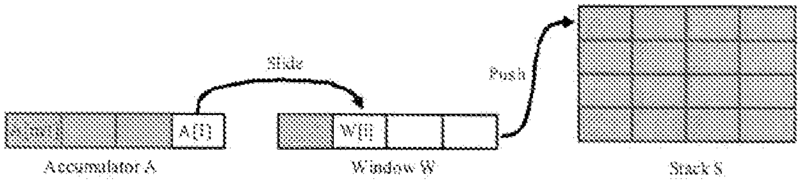


Fig. 7.1 Data structures used for sliding encryption

The technique of sliding encryption [7.7] helps to reduce the size of the ciphertext substantially by using the method of chain ciphering (see Fig. 7.1). The scheme described is based on the RSA [7.11] public-key encryption algorithm, but it is general enough to accommodate any other public-key encryption algorithm. The essentials of the scheme are described below.

Assume that the granule of plaintext that is collected from each site is of a small fixed length u. This is concatenated with randomly generated v bytes to construct a word of size $v+u$ in which the random word occupies the upper-order bytes. Let us call this composite word X which is of length $t = u+v$. The length of the RSA public key used is m. Both m and t are powers of 2, and $m >> t$.

The data structures used by the agent for sliding encryption include an accumulator A of m bytes, an m-byte window W, and a stack S, each stack element $S[i]$ being m bytes long. The accumulator is divided into m/t entities, each t bytes long. $A[1]$ contains the least-significant bytes of A, and $A[m/t]$ contains the most-significant bytes. We will denote the public-key encryption function by E() and the corresponding private-key decryption by D(). The functions E() and D() include an uneven Feistel-like preprocessing and postprocessing as in [7.12].

The mechanism works as follows. Initially the stack is empty, and the accumulator is initialized to a random non-zero positive integer K. After the piece of information is collected at the first site, the composite word X1 is formed by concatenating it with the randomly generated v bytes. Then A[1] is replaced by X1, and the resultant content of the whole accumulator is encrypted using E(). A[1] now contains the lowest-order bytes of the ciphertext. Then we set $W[m/t] = A[1]$. The remaining part of the ciphertext in the accumulator is carried unchanged and serves as a link in the chain ciphering process.

In the next node, the composite word X2 is similarly formed from the information picked up at that node, then A[1] is replaced by X2. The modified content of the accumulator is now encrypted using E(), and again the lowest bytes of the ciphertext, which now occupy A[1], slide into $W[(m/t)-1]$.

After all the *m/t* slots in the window W are full, it is pushed onto the stack S, and the sliding restarts from slot $W[m/t]$.

After all the nodes are visited and the mobile agent returns to its owner, the decryption process starts. This makes use of the private key and the process simply reverses the encryption steps sequentially, retrieving the hidden pieces of information in reverse order.

Trail Obscuring

This method depends on changing a mobile agent binary image to make it hard to identify by pattern matching. A mobile agent attempts to obscure its path through the network by constantly modifying its own binary image so that it cannot be identified as the same mobile agent by different hosts which are colluding in an attempt to track the mobile agent. This works in a situation where anonymity is required, such as an anonymous monetary donation or auction bid. It may also aid in surviving malicious hosts trying to stop specific behavior that can be identified by analyzing the mobile agent's path.

One important component of traceability of a mobile agent is its state information. If a group of adversary nodes compare the state information of a mobile agent captured in the snapshots taken by them, it can be possible to determine the order in which the nodes were traversed. Therefore, to thwart such attacks, the state information associated with an agent must be concealed. As an example, if sliding encryption is used as described above, then the state information will consist of the accumulator values, the window values, the stack, the stack pointer, and the index to the next location in the window where the next value of A[1] will slide into. In this case, one has to devise ways to conceal the real state using various techniques. For details the reader is referred to [7.7].

However, trail obscuring is not a foolproof method. A major problem of this approach is that mobile agents cannot encrypt and decrypt themselves, because, if they could, then any host could also do the same as it too will have access to the decryption key. Suppose that a subset of all the nodes visited by the mobile agent colludes to trace the agent's itinerary by taking snapshots of the agent while it was visiting each node of this subset. Under such circumstances, it is impossible to make an agent completely untraceable if all the adversaries are connected directly, and the agent cannot modify itself without being caught just before moving out of an adversary node.

Code Obfuscation

This method was discussed in [7.8]. Most of the above security problems can be solved if the host is not able to determine the relation between single lines of code and their semantics and the relation between memory bits and the semantics of data elements, respectively. A host can of course modify code, data and control flow anyway, but not with a computed effect. For a host this results in three choices:

- Host can execute the agent undisturbed.
- Host can execute the agent by switching some bits, not knowing about the effect on the execution.
- Host can take the agent without executing it.

An attacker needs a certain amount of time to read the data, understand the code and, thereafter, manipulate both in a meaningful way. The basic idea of the approach described now is simply not to give them enough time to do this. According to [7.8] this can be achieved by a combination of code mess-up and a limited lifetime of code and data.

With the employment of code mess-up techniques, Hohl has developed a non-cryptographic agent protection scheme that is built up like any cryptographic mechanism: readable input (i.e., code and data) is transformed to an unreadable form by a mechanism that cannot be inverted easily with the current knowledge.

Code mess-up does cost something, both in terms of speed and of space, and the processing model is more complex due to expiration aspects. Therefore, this scheme should be mainly used for agents that need to be protected, e.g., because they carry money or other sensitive data. The global usage of this mechanism even for nonsensitive applications may be too expensive, but because a code mess-up infrastructure is needed only for protected agents, agents of both protection levels can exist and interact in parallel. Hohl claims that it is possible to practically protect agents from malicious hosts by using code mess-up techniques. However, future work has to prove this claim.

Computing with Encrypted Function (CEF)

This method of concealing the computations of an agent from its host is proposed in [7.9] and [7.10]. Instead of using the more general term *program*, the authors differentiate between a function and the program that implements it. Thus, the goal is to encrypt functions such that their transformation can again be implemented as programs. The resulting program will consist of cleartext instructions that a processor or interpreter understands. What the processor will not be able to understand is the "program's function." With the requirements of mobile agents in mind, we can state the problem that we want to solve, as follows.

> Alice has an algorithm to compute a function *f*. Bob has an input x and is willing to compute $f(x)$ for her, but Alice wants Bob to learn nothing sub-

stantial about $f(\cdot)$. Moreover, Bob should not need to interact with Alice during the computation of $f(x)$. To let Alice and Bob work together in the way described above, we assume that a function f can be transformed (encrypted) to some other function $E(f)$. The encryption hides the function f and may or may not also contain the encryption of the output data. We let the notation $P(f)$ stand for the program that implements the function f. In this protocol Alice does not send to Bob the program $P(f)$ for the plain function f but the program $P(E(f))$ for the encrypted function $E(f)$. Bob only learns about the program $P(E(f))$ that he has to apply to his input x and the result of this computation that he has to return to Alice. The simple protocol for noninteractive computing with encrypted functions looks like this:

(1) Alice encrypts $f(x)$.
(2) Alice creates a program $P(E(f))$ which implements $E(f)$.
(3) Alice sends $P(E(f))$ to Bob.
(4) Bob executes $P(E(f))$ on x.
(5) Bob sends $P(E(f))(x)$ to Alice.
(6) Alice decrypts $P(E(f))(x)$ and obtains $f(x)$.

Noninteractive computing with encrypted functions is a challenge for cryptography. The challenge is to find encryption schemes for arbitrary functions. The authors of [7.9] identified some specific function classes (i.e., polynomials and rational functions) for which they could find encrypting transformations.

Their approach of studying algebraic homomorphic encryption schemes (HES) yields a first and simple scheme for CEF. However, they leave it open whether the CEF approach is applicable to arbitrary functions, that is, they don't even claim to have achieved a complete solution for the case of all polynomials. However, within the restricted setting of polynomials and rational functions they can prove first positive results that falsify the "general belief on mobile code vulnerability" for nontrivial cases.

7.4 Secure Payment Protocols Using Mobile Agents in an Untrusted Host Environment

With the above background we are now in a position to explore ways to design payment protocols for mobile agents that offer security against malicious hosts. In order to do that we need to define precisely the context in which the mobile agents operate and the specific threats of attacks they face. The following two sections define these parameters.

7.4.1 Model for the Mobile-Agent-Based E-Commerce Environment

There are mainly four parties in the payment system [7.13-7.14]: a bank B, a customer U, a merchant M, and a *trusted third party* (TTP). A TTP is an impartial entity that is trusted by both the customer and the merchant and whose testimony is accepted in a court of law as valid evidence. In addition, we assume the existence of a trusted certification authority that can certify the validity of the public keys of the different parties. Both customer U and Merchant M have accounts with the bank B. An electronic payment system consists of protocols that allow customer U to make a payment to the merchant M. The customer's site can create some buyer agents to do the information gathering, negotiation, and payment for him. The merchant's site can create some seller agents to interact with the customer or buyer agent. The bank can create one or more bank agents that can interact with the buyer agent and seller agent, providing some services, such as creating accounts, withdrawing and depositing money, transfering money, etc. TTP is used to provide a non-repudiation service in case any party should deny sending or receiving information in the protocol.

The buyer agents for the customer are hosted by a network of mobile agent hosts (MA hosts). These hosts provide a resident and executive environment for all these buyer agents.

In order to make the payment, the mobile agents have to communicate and transfer messages to one another. So the system must provide a mechanism for finding the current location of an agent and the MA host. All the entities, such as the customer U, Merchant M, Bank B, TTP, and all the MA hosts are assigned unique names. So that an agent can specify its desired destination when it migrates, a name server is provided which maps a symbolic name to the current location of the named entity. The locations of the customer, merchant, Bank, TTP, and MA hosts do not change often, whereas the location of a mobile agent is more likely to change. Whenever a mobile agent arrives or migrates to a new host, it should contact and inform the name server regarding its change of address, so that it can be located afterwards.

Threat Model

In this model, one or more agents representing a customer need to transfer sensitive payment information to the merchant site. The customer agents are hosted by a network of MA hosts. Both the MA hosts and the merchant site can be malicious. The following is a list of assumptions regarding the nature of the potential attacks in the scenario described above:

A1. At the most *m-1* malicious hosts may collude to steal sensitive information from the customer agent.

A2. The mobile agents travel through secure channels, i.e., no one other than the intended recipient can gain any information by eavesdropping on the communication channels.

A3. The merchant may deny that he has received the electronic money, and spend the money as his own later.

A4. The merchant site host works independently and does not collude with malicious MA hosts to cheat the customer.

A5. There is a "trusted third party" which is honest, and both the customer and the merchant trust that the TTP will execute its role correctly. The TTP has no role to play in the protection of the customer agents against malicious MA hosts.

7.4.2 A Secure Payment Protocol

Since electronic payment necessarily involves processing and transfer of confidential information, mobile payment agents are extremely vulnerable to attacks by malicious hosts. The CEF technique described above can be an effective tool to protect mobile agents from malicious hosts, but its applicability is quite restricted because of the limited current knowledge of the functions. An alternative is to ensure that at no point in time is the confidential information completely accessible to the untrusted hosts. This approach is adopted in the following protocol by making use of secret sharing schemes. The basic protocol described below [7.13-7.14] is based on Shamir's secret sharing scheme [7.15] and can use any digital cash scheme such as Chaum's digital cash [7.16] or the more efficient e-cash scheme of Stefan Brands [7.17-7.18]. But it can be adapted to many other payment methods and can be based on other secret sharing techniques as well.

7.4.3 Main Phases of the Protocol

There are four phases in the payment protocol:

1. **Withdrawal phase:** In this phase the customer withdraws electronic cash from their bank. The specifics of this phase will depend on the e-cash

scheme being adopted, and the subsequent phases are largely independent of this phase. At a conceptual level, it suffices to view this phase as generating some tokens which we call e-cash.

2. **Distribution phase:** In this phase, the customer encrypts the e-cash using a secret key and divides the secret key into several small shares using an (m, n) threshold scheme.

3. **Payment phase:** In this phase, m mobile agents work together and each passes its share of the secret key to the merchant's site. The merchant then reconstructs the secret key and deciphers the e-cash. Payment by e-cash typically involves a token transfer phase followed by an authentication phase in which the payee poses a challenge and the payer responds with appropriate values. The authentication phase can be handled by a single agent. For the sake of simplicity, we will confine our attention to the phase involving the transfer of the token only.

4. **Verification and transfer phase:** The merchant signs the e-cash deciphered in the payment phase and forwards it to bank B. The bank verifies that the e-cash submitted is not fake or duplicated and credits the amount to the merchant's account.

The core Secure Payment protocol consists of the Distribution Phase and the Payment Phase. In what follows we describe these phases in detail. But before we do so, the notation used is explained.

Notation:

- M: The merchant (or payee).
- s: Secret key for encrypting e-cash.
- C: The e-cash to be transferred as payment.
- $e_s(m)$: Message m symmetrically encrypted using secret key s.
- $d_s(g)$: Encrypted message g decrypted using secret key s.
- $(m)^k$: Message m asymmetrically encrypted with a public/private key k.
- $S_X(m)$: Message m digitally signed by X.
- Share$_i$: The share of the secret carried by the ith agent.
- $H(\cdot)$: One-way collision-free hash function. We will denote $H(\text{Share}_i)$ as $H(i)$.
- *Lagent*: A leader agent used to organize the transfer of the shares, any of the mobile agents carrying a share of the secret can be a leader agent
- P_x and V_x : The public/private key pair of party x.

- F_{REC}, F_{SUB}, F_{CON}: Flags used to identify the steps of transferring the shares in the protocol. They indicate the intended purpose of a (signed) message. F_{REC}, F_{SUB}, and F_{CON} indicate that the objective of the step is transferring a receipt, submitting a document, and confirming the receipt of a document, respectively.
- TTP: On-line trusted third party providing security services accessible to the public.

The steps of the two core phases are described below.

7.4.4 Distribution Phase

The steps of this phase are as follows:

1. **Distribute the secret keys using a secret sharing scheme**

The customer distributes s using the secret sharing scheme, an (m, n) threshold scheme. For this purpose an arbitrary polynomial of degree $m-1$ is generated:

$$f(x) = a_0 + a_1 x + a_2 x^2 + \cdots + a_{m-1} x^{m-1}, \text{where } a_0 = s\ .$$

The coefficients $a_1, a_2, \ldots a_{m-1}$ are chosen randomly from the set $Z_p = \{0, 1, \ldots, p-1\}$ and are kept secret and discarded after the shadows are handed out. p is a prime larger than the number of possible shadows (shares), and the largest possible secret. All arithmetic is done modulo p.

The n shadows are obtained by evaluating the polynomial at n random different points, $y_i = f(x_i)$. The values of $x_i (i = 1, \ldots, n)$ are made public whereas $y_i (i = 1, \ldots, n)$ are kept secret and act as the n shares $\text{share}_i (i = 1, \ldots, n)$.

It is easy to see that when any m shadows come together, linear algebra can be used to solve for the coefficients of the polynomial, including the constant term, $a_0 = s$.

2. **Compute n message digests for n shares**

A hash function is used to compute $H(i) = H(\text{share}_i)$, $i = 1, \ldots, n$.

3. Prepare *n* mobile agents

The customer creates *n* agents, each agent assigned a unique name Agent$_i$ $(i = 1,\dots,n)$, and will carry the encrypted e-cash $e_s(C)$, a share of the secret *s*, and a set of *n* ordered pairs $\{x_j, H(j)\}$, $j = 1,\dots,n$.

4. Dispatch *n* mobile agents to *n* hosts

The *n* agents are dispatched to *n* distinct hosts. Each agent will carry the following information:

$$\text{Agent}_i, e_s(C), \text{share}_i, \{\text{Agent}_j, x_j, H(j), j = 1,\dots,n\}.$$

5. Register the location of these *n* mobile agents

After dispatching a mobile agent, the customer's site should register at the name server for the new location of the dispatched mobile agent. The name server maintains a hash map, each record of which is a pair, the agent name and the location. The location consists of a hostname and a port number. It can be resolved using the domain name server (DNS). The agent name is unique, so that the customer and other mobile agents can communicate with it or control it while it travels on its itinerary. So the content of the hash map is:

$$\{\text{Agent}_i, \text{Location (Hostname : PortNumber)}\}.$$

7.4.5 Payment Phase

The payment phase begins when one of the mobile agents of the customer initiates the payment process after receiving the payment order from the merchant. This agent will be designated as the leader agent or *Lagent*. The payment order (PO) consists of a number of components, as given below:

$$PO = S_M(Tid, Lagent, M, Goods\_desc, Amount, Time)$$

where

Tid = unique identifier for the transaction being carried out,
$Goods\_desc$ = description of the goods being purchased,
$Amount$ = the amount to be paid to the merchant,
$Time$ = the time when the payment order is made,
$S_M(message)$ = indicates that the message is signed by *M*.

The *Lagent* will randomly select the other $m-1$ mobile agents and send the payment order signed by the merchant. After the other $m-1$ mobile agents have verified the payment order, they will send their shares to the merchant. The merchant sends a signed acknowledgment to *Lagent* after the merchant has received $m-1$ shares. The *Lagent* then sends the last share to a TTP from which the merchant collects it.

The steps of this phase are explained below.

1. Initialization

The *Lagent* randomly selects $m-1$ Agents from the information it carries, finds the locations of these $m-1$ mobile agents from the *name server*, and sends the following information to the selected mobile agents: the message digest of its share $\{x_j, H(j)\}$, the merchant identity *M*, and the payment order signed by *M*.

Other mobile agents can authenticate *Lagent* using the message digest. They verify the payment order using the merchant's public key. If anything inconsistent is found, the payment is stopped and the problem is reported to the owner.

2. Other *m-1* mobile agents send shares to the merchant

All the selected mobile agents send their shares of the secret key to the merchant:

$$Agent_i \rightarrow M : (Tid, Share_i, x_i)^{P_M} \text{ .}$$

3. Merchant sends the acknowledgement to *Lagent*

After $m-1$ shares have been received by the merchant, the merchant computes the message digests of the $m-1$ shares using the same hash function $H(\cdot)$. Then the merchant sends the following message as a receipt to *Lagent*:

$$M \rightarrow Lagent : S_M\{F_{REC}, Tid, M, Lagent, m-1 \text{ pairs } (x_i, H(i))\} \text{ .}$$

4. *Lagent* sends its share as the last share to the TTP

Lagent verifies that each share received by the merchant is valid by comparing each message digest in the receipt with the one carried by itself. After that *Lagent* sends the *m* th (or last) share to the TTP.

$$Lagent \rightarrow TTP : (F_{SUB}, Tid, Lagent, M, e_s(C), Share_m, x_m)^{P_{TTP}} \text{ .}$$

5. *M* and *Lagent* retrieve the confirmed message from the TTP

Both *Lagent* and M have to retrieve the confirmed message from the TTP as part of the non-repudiation evidence required in a dispute. It is assumed that even in the case of network failures, both parties will eventually be able to retrieve the message from the TTP.

$$M \leftrightarrow TTP : (F_{CON}, Tid, Lagent, M, e_s(C), Share_m, x_m)^{V_{TTP}} ,$$

$$Lagent \leftrightarrow TTP : (F_{CON}, Tid, Lagent, M, Share_m, x_m)^{V_{TTP}} .$$

The two-sided arrow ($\leftrightarrow$) indicates that the transfer is initiated by the recipient through an ftp call.

6. Reconstruction of the secret key

Once the merchant gets all *m* shares, they reconstruct the secret key *s* using the Lagrange interpolation formula:

$$s = a_0 = \sum_{j=1}^{m} Share_j \prod_{1 \le k \le m, k \ne j} \frac{x_k}{x_k - x_j} . \qquad (1)$$

7. Payment

The merchant decrypts the e-cash using $C = d_s(e_s(C))$, signs the e-cash with the merchant's private key V_M , and forwards it to the bank.

7.4.6 Correctness of the Protocol

We show that the secure payment protocol (SPP) presented above provides adequate protection under the threat model presented in the previous section. We do so through a couple of simple claims.

Claim 1. The protocol SPP ensures that the e-cash is protected against spying/stealing by $m-1$ or fewer malicious MA hosts.

Proof. The e-cash is protected by encryption and it requires at least *m* agents to reconstruct the encryption key. Since all the transfers take place through secure channels, and at no point of the protocol has any one agent access to more than one share, this property is trivially guaranteed provided that the agent itinerary ensures that no MA host is ever visited by more than $m-1$ agents.

Note that the protocol will fail if there is collusion between the merchant and the host of *Lagent*.

Since the merchant receives $m-1$ shares from the other $m-1$ agents, the merchant simply needs to pass them to the host of the *Lagent*, who then uses that information to extract the e-cash.

Claim 2. The protocol produces evidence to support non-repudiation for both the customer and the merchant.

Proof. The nonrepudiable evidence is generated at steps 3, 4, and 5 of the payment phase. At step 3, the merchant signs and sends message digests of the $m-1$ shares already received by them. If these digests are not all valid, the *Lagent* will not complete the payment. If they are valid, they serve as evidence of M's receipt of the $m-1$ shares.

In step 4, the *Lagent* passes the last share as well as the encrypted e-cash to the TTP. This message is protected by encryption using the public key of the TTP. This ensures that the merchant cannot spy on the last share from this message.

This message need not be signed by *Lagent* as the TTP is trusted. (Otherwise it could have passed the message clandestinely to the merchant, later corrupting the data and producing untenable non-repudiation evidence.)

In step 5, both M and *Lagent* retrieve the message using ftp, which serves as the non-repudiable evidence of transfer of the last share as well as the e-cash.

In summary, at the end of this protocol, if *Lagent* wants to prove that the shares have been received, it presents

$$S_M(F_{REC}, Tid, M, Lagent, m-1 \text{ pairs } \{x_i, H(i)\})$$

and $$(F_{CON}, Tid, Lagent, M, e_s(C))$$

to the judge. The first piece of evidence confirms that M received the $m-1$ shares, and the second piece confirms that the last share was deposited with the TTP, which means that the merchant has access to it.

7.4.7 Efficiency of the Protocol

The message complexity of the protocol can be computed as follows. In the initialization phase, the *Lagent* sends $m-1$ messages of $O(1)$ length to $m-1$ participating agents. Each of the agents transfers its share to the merchant, using altogether $m-1$ messages of $O(1)$ length.

The merchant sends the single acknowledgement message of length $O(m-1)$ in step 3. In steps 4 and 5, three messages are transmitted, each of length $O(1)$. Thus altogether, the complexity of the messages communicated is $3O(m\text{-}1)+\ 3O(1)$, whereas the total number of messages transferred is $3(m\text{-}1)+3 = 3m$.

The parameter m can be viewed as a measure of untrustworthiness of the host network. Therefore, it can be said that the cost of protection increases linearly with the number of untrustworthy hosts in the network. The secret key is distributed to n shares, m of them is enough to reconstruct the secret, $n \geq m$.

The larger m and n are, the more secure and reliable the protocol is. But at the same time, the cost increases. To select m and n one needs to find a good balance among the safety, reliability, cost, and efficiency of the protocol.

It is worth noting here that the protocol involves the TTP only for the transfer of the last share. So the TTP remains unaffected by the choice of m.

7.4.8 Limitations of the Protocol

The protocol discussed above is a very basic one and has several drawbacks that must be addressed effectively before it can be put to practical use. First of all, since the secret key is revealed at the end of each payment operation, the protocol requires a different key to be used for each payment transaction.

Second, since the value of the e-cash token is fixed at creation, this protocol cannot be used in cases where the amount to be paid is not known before the creation of the mobile agents.

Third, there are important additional security issues that need to be addressed. For example, if m or more of the mobile agents pass through any given untrusted host in the course of their nomadic lifetime, that host can gather all the information necessary to reconstruct the secret. To avoid this possibility, one can either preplan the itineraries of all the mobile agents or the mobile agents need to consult a central controlling agent for clearance to move to an intended site.

Another possible attack can be mounted in which a malicious host that hosts any one agent creates a fake merchant identity and makes up a payment order in which it makes itself the recipient site. However, any payment protocol is vulnerable to such an attack, and the only effective way to address this is by making the certification process more reliable.

7.4.9 An Electronic-Check-Based Payment Protocol

The constraints imposed by the SSP protocol presented above, namely the single use of secret keys and the predetermined amount of the payment can be removed with some modifications if the requirement for anonymity is given up. In what follows, we describe a modified version of the SSP protocol that uses electronic checks rather than e-cash and thus no longer provides anonymity to the customer.

The exposure of the secret key at the end of each transaction can be avoided by using a homomorphic secret sharing scheme [7.19]. We can partition the private key in a way that each mobile agent can partially encrypt or sign the message without revealing their share to the combiner. After the last partial signature, the document is completely signed with the shared private key, and none of the shareholders learns about any other shares. The combiner can get the whole signature after each mobile agent has signed without knowing the private key. This concept is explored further in [7.20].

Let g be an encryption function, if g is homomorphic it satisfies the following equation:

$$g(k_1 + k_2) = g(k_1) \times g(k_2) \,. \tag{2}$$

If $k_1 + k_2$ is the encryption key, then a threshold cryptographic system may be constructed [7.21]. As an example, in computing an RSA signature one computes $g_h(s) = h^s \bmod n$, where $h = H(message)$ is the message digest, s is the secret key, $n = p \times q$ is the public modulus, and p and q are two distinct large primes. From (1), and with careful choice of p and q, Shamir's scheme satisfies the property

$$s = \sum_{i \in Q} (\text{constant}_{i,Q} \times s_i) \,, \tag{3}$$

where Q is a quorum subset of the set of all participants, called A, and $|Q| = m$ is the threshold value. From (1), the terms $\text{constant}_{i,Q}$ can be obtained from the known values of x_i as follows:

$$\text{constant}_{i,Q} = \prod_{i,k \in Q, k \neq i} \frac{x_k}{x_k - x_i}. \tag{4}$$

Hence, when combined with a homomorphic g, one obtains:

$$\begin{aligned} g_h(s) &= g_h \sum_{i \in Q} \text{constant}_{i,Q} \times s_i \\ &= \prod_{i \in Q} g_h(\text{constant}_{i,Q} \times s_i) \\ &= \prod_{i \in Q} g_h(s_i)^{\text{constant}_{i,Q}} \end{aligned} \tag{5}$$

Thus, an RSA signature can be computed using partial signatures, yet both the private key s and the various shadows remain secret even after combining the shares.

The payment protocol based on e-check is described briefly in the following steps:

Step 1. The customer first distributes her private key s using a polynomial of degree m-1, and gets n shadows s_i by evaluating the polynomial at n different points x_i. The customer creates n mobile agents with $\{x_i, s_i\}$ and dispatches them to n different hosts through secure channels.

Step 2. Whenever any of the customer agents, say *agent*$_j$, wishes to make a payment against a properly authenticated payment order sent by the relevant merchant, it creates an e-check C of an appropriate amount, computes $h = H(C)$ and then a partial signature $g_h(s_j)$. The agent sends $\{C, g_h(s_j), x_j\}$ along with proper identifiers to the merchant through a secure channel. At the same time it sends the payment order and a copy of the check C to randomly selected m-1 other customer agents.

Step 3. All the *m-1* customer agents that receive the above message generate their own partial signature $g_h(s_i)$ and send $\{g_h(s_i), x_i\}$ along with proper identifiers to the merchant through a secure channel.

Step 4. After collecting *m* partial signatures, the merchant combines them to obtain the fully signed e-check $g_h(s)$ using the formula given in (5).

Note that a TTP can be involved if a non-repudiation service is needed. In this case, the leader agent will send its partially signed check through the TTP instead of sending it directly to the merchant and a protocol similar to the one described using e-cash can be used to ensure proper documentation.

It may be noted that since this protocol involves the signature of the customer, it does not support the anonymity of the customer.

7.4.10 Possibility of Combining Anonymity with Reuse of Secret Key by Payment Agent

To protect a payment agent from malicious usurpation, one must ensure that no agent ever has complete knowledge of a validated instrument of payment, such as cash or a signed check. As we have seen in the preceding section, using a threshold encryption scheme, a set of mobile agents can generate an e-check of arbitrary amount independently of any intervention by the owner and can complete payment in a secure manner. However, this process requires the customer agents to give up anonymity. The question is whether it is possible to combine the flexibility of making payments in arbitrary denominations a multiple number of times using a single secret key while retaining anonymity. With the available cryptographic techniques that seems to be impossible at the time being. Whether it is possible at all is debatable, and it is best to withhold any conclusion until it is proven formally either way.

7.5 Summary

In this chapter, we have addressed the problem of protecting sensitive information carried by mobile agents from malicious hosts, and proposed two payment protocols using Shamir's secret sharing scheme. One of the protocols is based on electronic cash that allows the customer to carry out transactions anonymously. But it imposes two constraints, namely, the amount to be paid must be predetermined, and secondly for each payment transaction a new secret key needs to be used. The second protocol, which is based on electronic check payment, removes these constraints at the cost of anonymity. The protocols guarantee protection of confidential data, such as electronic cash, against concerted attack by a known maximum number of malicious hosts. In addition, by making optional use of a TTP in a minimal way, it produces non-repudiable evidence of transfer of funds from the customer to the merchant.

7.6 References

[7.1] A. Chavez, P. Maes (1996) Kasbah: an agent marketplace for buying and selling goods. In: Proceedings of the First International Conference on the Practical Application of Intelligent Agents and Multi-agent Technology.

[7.2] J. G. Lee, J. Y. Kang, E. S. Lee (1997) ICOMA: an open infrastructure for agent-based intelligent electronic commerce on the Internet. In: Proceedings of the International Conference on Parallel and Distributed Systems.

[7.3] P. Maes, R. H. Guttman, A. G. Moukas (1999) Agents that buy and sell: transforming commerce as we know it. Comm ACM (March Issue).

[7.4] A. Moukas, R. Guttman, P. Maes (1998) Agent-mediated electronic commerce: an {MIT} media laboratory perspective. In: Proceedings of ICEC Conference, 1998.

[7.5] M. Tsvetovatyy, M. Gini (1996) Toward a virtual marketplace: architectures and strategies. In: Proceedings of the First International Conference on the Practical Application of Intelligent Agent and Multi-agent Technology (PAAM'96), Blackpool, 1996.

[7.6] M. Tsvetovatyy, M. Gini, B. Mobasher, Z. Wieckowski (1997) MAGMA: an agent-based virtual market for electronic commerce. J Appl Artificial Intelligence.

[7.7] A. Young, M. Yung (1997) Encryption tools for mobile agents: sliding encryption. In E. Biham (ed.) Fast software encryption – FSE'97, LNCS 1267. Springer, Berlin Heidelberg New York.

[7.8] F. Hohl (1997) An approach to solve the problem of a malicious host. Report No. 1997, Universität Stuttgart, Fakultät Informatik.

[7.9] T. Sander, C. Tschudin (1997) Towards mobile cryptography. Technical Report, International Computer Science Institute, Berkeley.

[7.10] T. Sander, C. Tschudin (1997) Protecting mobile agents against malicious hosts. In: Mobile agent security, Springer, Berlin Heidelberg New York.

[7.11] M. Bellare, P. Rogaway (1994) Optimal asymmetric encryption. In: Eurocrypt 94, LNCS 950. Springer, Berlin Heidelberg New York.

[7.12] R. Rivest, A. Shamir, L. Adleman (1978) A method for obtaining digital signatures and public key cryptosystems. Commun ACM 21(2): 120–126.

[7.13] A. Das, G. Yao (2001) A secure payment protocol using mobile agents in an untrusted host environment. In: W. Kou, et al. (eds.) Electronic commerce technologies – ISEC 2001, LNCS 2040. Springer, Berlin Heidelberg New York.

[7.14] G. Yao (2001) Security mechanisms for mobile agent-based e-commerce systems. Master's Thesis, Nanyang Technological University.

[7.15] A. Shamir (1979) How to share a secret. Commun ACM 22:612–613.

[7.16] D. Chaum (1989) Online cash checks. In: Proceedings of Advances in Cryptography–Eurocrypt'89, LNCS 434. Springer, Berlin Heidelberg New York.

[7.17] S. A. Brands (1993) An efficient off-line electronic cash system based on the representation problem. Technical Report CSR9323, Computer Science Department, CWI, US.

[7.18] S. Brands (1994) Untraceable off-line cash in wallet with observers. In: Advances in Cryptology–CRYPTO'93. Springer, Berlin Heidelberg New York.

[7.19] J. C. Benaloh (1997) Secret sharing homomorphisms: keeping shares of a secret secret. In: A. Odlyzko (ed.), Advances in Cryptology, Proc. of Crypto'86, Santa Barbara, CA, US, Aug. 1987.

[7.20] Y. Desmedt (1997) Some recent research aspects of threshold cryptography. In: E. Okamoto, et al. (eds.), Information security, LNCS 1396. Springer, Berlin Heidelberg New York.

[7.21] N. Jacobson (1985) Basic algebra, I.W.H. Freeman, New York.

8 Digital Cash

Yi Mu[1], Vijay Varadharajan[1], and Khanh Quoc Nguyen[2]

[1] Department of Computing, Macquarie University,
Sydney, Australia

[2] Gemplus Technologies Asia,
12 Ayer Rajah Crescent, Singapore

8.1 Introduction

A digital-cash system normally consists of clients, vendors, and a bank. Any legitimate client can obtain a valid digital coin[1] from a bank and anonymously send the coin to a vendor. The vendor later deposits the coin to the bank. Because of the anonymity of the client, the bank can validate the coin but cannot link the coin to the information used in the coin-issuing process. The bank and the vendor cannot trace transactions made by the client.

The first digital-cash scheme was proposed by Chuam [8.1]. The transaction untraceability proposed in Chaum's digital-cash is based on the use of zero-knowledge proofs, which is computationally expensive and is not efficient enough for any real applications. Some subsequent works [8.2, 8.3, 8.6, 8.7] have achieved various improvements on Chaum's scheme. In particular, protocols proposed by Brands [8.2] and Ferguson [8.6] achieve transaction untraceability without requiring zero-knowledge proofs.

There are several forms of digital-cash. Besides the normal digital-cash, there are two additional catalogs: divisible digital-cash and fair digital-cash.

The first divisible digital-cash scheme was proposed by Okamoto [8.7], and then two more efficient methods were proposed [8.8, 8.9]. The divisible digital-cash scheme allows a user to divide a digital coin into several even

[1] For convenience, we use "digital coin" to represent a monetary unit of digital cash.

pieces that can be used as normal digital coins. Fair digital-cash schemes are applied to restricting unconditional privacy of clients. In a normal digital-cash system, there is no any mechanism for banks and vendors to identify a client in a transaction without breaking the underlying number theoretic assumptions. This protection, which is desirable from client's viewpoint, is a major concern for law enforcement agencies. It was pointed out in [8.16, 8.31] that anonymous e-cash can be a "safe haven" for criminal activities that include money laundering, illegal purchases, perfect blackmailing and other attacks. This prevents the deployment of anonymous digital-cash cash systems in a large scale, where such attacks and many others are often expected. In a fair digital-cash system, a designated trusted third party can compute the identity of a client when necessary.

In this chapter, we will introduce three typical digital-cash schemes: the digital-cash scheme proposed by Brands [8.2], and the digital-cash and the fair digital-cash scheme proposed by Nguyen et. al. [8.14, 8.14].

8.2 Security Requirements for Digital Cash

Digital-cash must not be illegally forgeable and cannot be double spent. In the meanwhile, digital-cash must also have such properties as providing anonymity to clients and untraceability to digital coins. In general, Digital-cash should have the following security properties:

- Unforgeability. This is the basic requirement for digital-cash. Digital cash must not be able to be forged in a polynomial time frame.
- Untraceability. Once a digital coin is issued, it is not traceable by the bank and any other parties.
- Anonymity. The identity of the digital-cash owner should not be revealed. That is, given a digital coin, any other parties cannot find the identity of the owner.
- Double-spending detection. A digital coin can be used only once, bounded by its monetary value. Any attempt at duplication of a coin or double uses of a coin can be detected by the bank that has signed the coin.
- Fairness. In fair digital-cash, the anonymity of a coin owner is conditional. There is a trusted third party who can find the identity of the coin owner when the coin has been illegally used.

8.3 Brands' Digital-Cash Scheme

Brands proposed the first digital-cash scheme without using cut-and-choose, therefore it is much more efficient than Chaum's original scheme. In this section, we introduce Brands' scheme. Like Chaum's scheme, Brands' digital-cash scheme has four phases: opening an account, withdrawal, payment, and deposit. For simplicity of the following presentation, we have slightly modified the scheme without compromising its security.

8.3.1 The Setup

We first give some general notations to be used in Brands' digital-cash scheme.

- p: a large prime
- q: an integer satisfying $q|p-1$
- $\mathbb{Z}_p^*$: a multiplicative group of prime order q satisfying $q|p-1$
- $\mathbb{G}_q$: a multiplicative group of prime order q and $\mathbb{G}_q \subset \mathbb{Z}_p^*$
- $\mathbb{Z}_q$: a finite field of size q
- $\mathcal{H}(.)$: a strong correlation-free one-way hash function $\mathcal{H}(.) \in \mathbb{Z}_q$

The system consists of clients, merchants, and a bank. We denote by C a client, M a merchant, and B a bank.

B chooses $(g_1, g_2, g_3) \in_R \mathbb{G}_q^3$ as generators and a number $x \in_R \mathbb{Z}_q$ as its private key, and then computes its public key $z = g^x \bmod p$. For simplicity, we will omit modulo p in the following protocols.

8.3.2 Opening an Account

To open an account, C and B follow the following steps:

1. The client C needs to identify himself to the bank B, when opening an account by, for example, showing his passport to B.
2. C selects a secret random number $U \in_R \mathbb{Z}_q$ and computes $I = g_1^U$ where $g_1^U g_2 \neq 1$, then sends I to B.
3. B stores C's identification information along with I. I will be used as C's account number.

The security of U is based on the difficulty of solving discrete logarithm. In other words, if C spends a digital coin related to I once, his identity U cannot be computed within a polynomial time frame.

8.3.3 The Withdrawal Protocol

To withdraw a coin from B, C needs first identify himself to B and prove his ownership of the account. The following withdrawal protocol is then performed (also see Fig. 8.1).

1. B generates a number $w \in_R \mathbb{Z}_q$ and sends $a = g^w$ and $b = (Ig_2)^w$ to C.
2. C generates three numbers $s, x_1, x_2 \in_R \mathbb{Z}_q$ and computes $A = (Ig_2)^s$, $B = g_1^{x_1} g_2^{x_2}$, and $z' = z^s$. C also generates two numbers $u, v \in_R \mathbb{Z}_q$ and uses them to compute $a' = a^u g^v$ and $b' = b^{su} A^v$. C then computes the challenge $c' = \mathcal{H}(A, B, z', a', b')$, and sends the blinded challenge $c = c'/u \bmod q$ to B.
3. B sends the response $r = cx + w \bmod q$ to C, and debits the account of C.

C accepts iff $g^r = h^c a$ and $(Ig_2)^r = z^c b$. If this verification holds, C computes $r' = ru + v \bmod q$. The withdrawn coin consists of (A, B, z', a', b', r').

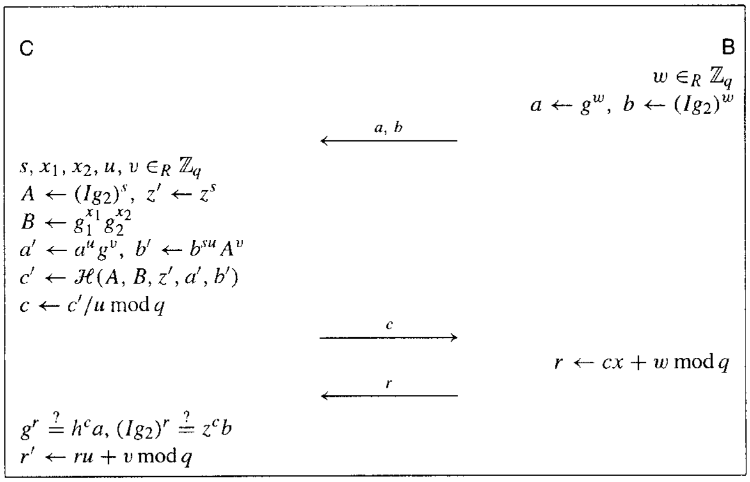

C
B
$w \in_R \mathbb{Z}_q$
$a \leftarrow g^w, \ b \leftarrow (Ig_2)^w$
a, b
$s, x_1, x_2, u, v \in_R \mathbb{Z}_q$
$A \leftarrow (Ig_2)^s, \ z' \leftarrow z^s$
$B \leftarrow g_1^{x_1} g_2^{x_2}$
$a' \leftarrow a^u g^v, \ b' \leftarrow b^{su} A^v$
$c' \leftarrow \mathcal{H}(A, B, z', a', b')$
$c \leftarrow c'/u \bmod q$
c
$r \leftarrow cx + w \bmod q$
r
$g^r \overset{?}{=} h^c a, \ (Ig_2)^r \overset{?}{=} z^c b$
$r' \leftarrow ru + v \bmod q$

Fig. 8.1 The coin withdrawal protocol

8.3.4 The Payment Protocol

When C wants to spend his coin at V, the following protocol is performed.

1. C sends A, B, and Sign(A, B) to V, where Sign(A, B), which represents B's signature on a pair $(A, B) \in (G_q)^2$, is the tuple (A, B, z', a', b', r').
2. V computes a challenge $d = \mathcal{H}_0(A, B, ID_V, Date/Time)$ and sends d to C.
3. Upon receipt of c, C computes the responses $r_1 = du_1s + x_1 \bmod q$ and $r_2 = ds + x_2 \bmod q$, and then sends them to C.

V accepts the coin iff Sign(A, B) is valid and $g_1^{r_1} g_2^{r_2} = A^d B$.

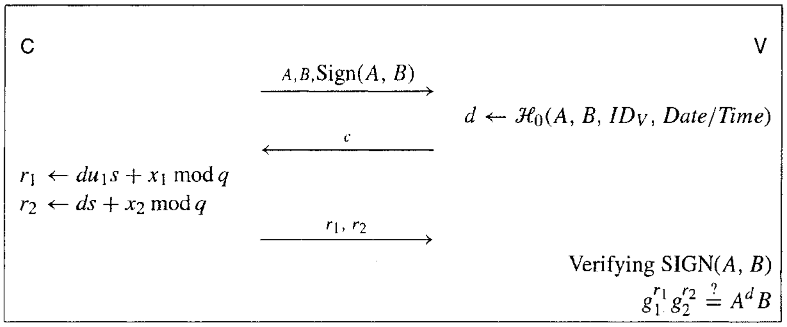

C
V
A, B,Sign(A, B)
$d \leftarrow \mathcal{H}_0(A, B, ID_V, Date/Time)$
c
$r_1 \leftarrow du_1s + x_1 \bmod q$
$r_2 \leftarrow ds + x_2 \bmod q$
r_1, r_2
Verifying SIGN(A, B)
$g_1^{r_1} g_2^{r_2} \overset{?}{=} A^d B$

Fig. 8.2 The payment protocol

8.3.5 The Deposit Protocol

V sends the payment transcript of a coin to B. B needs to check if the coin has been stored before. If not, B stores (A, Date/Time, r_1, r_2) in its database as being deposited by V, and credits the account of V. However, if the coin has been spent before, a double-spending fraud must have occurred. A double-spending can be easily detected, since the challenges are different. B can obtain a triplet (c, r_1, r_2) from the new transcript and a triplet (c', r_1', r_2') from the deposited information. B can compute $I = g_1^{(r_1 - r_1')/(r_2 - r_2')}$ and then search its database for this account number.

8.4 One-Response Digital Cash

Current offline digital-cash systems [8.1, 8.4, 8.6] tend to provide double-spending detection, client anonymity, and transaction untraceability. However, as there is always a trade-off between double-spending detection and transaction untraceability, the computational cost is often high. Even in the most efficient systems [8.4, 8.6], many discrete exponential computations are required for each digital monetary unit in order to achieve the untraceability. To design an electronic-payment system that allows small payment amounts, heavy use of discrete exponential computations must be avoided. In fact, this requirement makes all current offline cash systems economically infeasible.

In this section, we introduce an efficient approach [8.13] to offline digital-cash schemes that makes small payment amounts possible. This scheme maintains the basic features in digital-cash: client anonymity and double-spending detection. The computational efficiency achieved in this scheme is due to the use of one-way hash functions that make clients perform only *one* major computation and perform no discrete exponential computations in the payment phase. This feature leads to a significant improvement in computational efficiency in contrast to all previously proposed schemes.

8.4.1 Schnorr's One-Time Signature Scheme

Schnorr's one-time signature scheme [8.11] is used for this digital-cash scheme. Let p and q be prime such that q is a prime factor of $p-1$. Let $g \in \mathbb{Z}_p^*$ be the multiplicative primitive, where $g \neq 1$ and $g^q \equiv 1$. Again, we omit the modulo p in our presentation.

To generate a particular pair of private key and public key, a user (say, Alice) chooses a random number s as her private key, $0 < s < q$. Alice then computes her public key v as $v = g^{-s}$.

To sign a message m, Alice picks a random number $r \in \mathbb{Z}_q$ and does the following computations:

$$\begin{aligned} x &= g^r \\ c &= \mathcal{H}(m \| x) \\ y &= (r + sc) \bmod q, \end{aligned}$$

where $\mathcal{H}(.)$ is a suitable collision-free one-way hash function. The signature on the message m is the pair (c, y). To verify the signature, we check: $x \overset{?}{=} g^y v^c$ and check if c is equal to $h(m \| x)$.

8.4.2 The One-Response Digital-Cash Protocol

We assume that each *coin* in this scheme represents a monetary unit. The face value of each coin is decided by the bank. We denote by C_i a coin with an abstract face value c_i. We also assume that the bank has a RSA public/secret key (e, d) with the composite modulo n of the product of two large prime numbers, q_1, q_2, and a number g such that $g^q \equiv 1$ and $gcd(g-1, n) = 1$. The values of g, p, q, n, and e are public.

Account opening phase. When C wishes to open an account at B, after identifying himself to B, C uses a zero-knowledge process to obtain a blind-signature from B on $h(g^U)$ as $(h(g^U))^d \bmod n$. U is constructed as $U = I \| k$ $(0 < U < q)$ by C, where $\|$ denotes a concatenation of bits, k is a random number, and I is the client identity registered with the bank (also referred as the client's bank account number). The bank should not have any knowledge about the value of k and consequently the value of U. There have been several such zero-knowledge processes[2] described in the literature; see, for instance, [8.12, 8.10].

The length of I and k should be fixed, at least 80 bits each, so that given g^U, it is feasible to obtain I. After the account's opening phase, the client has an anonymous bank certificate *Cert* as $(h(g^U))^d \bmod n$. This certificate would remain anonymous as long as nobody is able to compute U. Extracting U from *Cert* is infeasible unless the client double-spends under the discrete logarithms assumption (Further discussions will be given later.) After the account-opening process, C stores *Cert* and $g_C = g^U$.

Withdrawal phase. Before withdrawing any money from the bank, the client C proves his ownership of I to B. If the client wishes to withdraw k coins, he chooses a random number c_k and computes $c_i = h(c_{i+1})$ for $\forall i \in \{1, \ldots, k-1\}$. For each c_i, C uses a blind signature technique[8.5] to withdraw an anonymous coin from B using the following protocol (see Fig. 8.3):

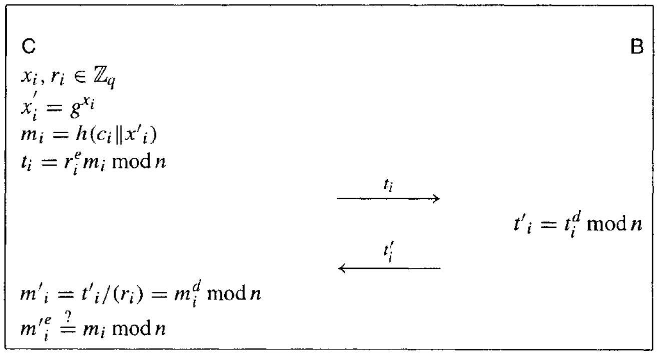

C
B
$x_i, r_i \in \mathbb{Z}_q$
$x_i^{'} = g^{x_i}$
$m_i = h(c_i \| x'_i)$
$t_i = r_i^e m_i \bmod n$
t_i
$t'_i = t_i^d \bmod n$
t_i'
$m'_i = t'_i/(r_i) = m_i^d \bmod n$
$m'^e_i \overset{?}{=} m_i \bmod n$

Fig. 8.3 The withdrawal protocol

[2] A cut-and-choose method has to be used in the proof. To avoid it, a trusted third party may be included for verifying correctness of $h(g^U)$ and signs it prior to the bank's signing.

1. C generates a random number $x_i \in_R \mathbb{Z}_q$, and computes: $x_i' = g^{x_i}, m_i = h(c_i \| x'_i)$.

2. C then uses blind signature technique [8.5] to obtain a bank signature on m_i by choosing a blind factor r_i and sending $t_i = r_i^e m_i \bmod n$ to B. B signs the value of t_i and returns the signature as t_i'. The client then removes the blind factor r_i to obtain the bank blind signature $m_i' = t_i'/r_i = m_i^d \bmod n$.

For each signature, the bank deducts the client's account by an equivalent value of a coin. After the withdrawal, C has each coin C_i with a face value of c_i in the form of $[h(c_i \| x_i')]^d \bmod n$. It is unforgeable unless the factorisation of n is known.[8.1] For each coin C_i, C stores $[c_i, x_i, x_i', m_i']$.

Payment phase. When the client wants to spend the coin chain $C_1, C_2, \ldots, C_n$ to V, he must spend them in the order $C_1, C_2, \ldots, C_n$.

Without the loss of generality, we assume that C has already spent all the coins $C_0, C_1, \ldots, C_{i-1}$ in some previous payments. Now if C wishes to pay some coins to V, C must send the coins to V in the exact sequence $C_i, C_{i+1}, \cdots, C_j, \cdots$ according to the following process:

- For the first coin C_i (see also Figure 8.4):

    1. V generates a random challenge a and sends it to C. This challenge should be unique for each transaction. For example, it can be computed as $a = h(\mathsf{C} \| \mathsf{V} \| \mathit{Date} \| \mathit{Time})$.
    2. C computes the response $b = x_i - Ua \bmod q$ for the challenge a and sends it along with $(\mathit{Cert}, g_{\mathsf{C}}, b, c_i, x_i', m_i')$ to V. The response b is also considered as Schnorr's one-time signature on the message a, where x_i is a one-time value.

    V accepts the coin if and only if *Cert* and m_i' are valid bank signatures on g_{C} and $\mathcal{H}(c_i \| x_i')$, respectively, and $g_{\mathsf{C}}^a g^b = x_i'$.

- For every coin C_j, thereafter (see also Fig. 8.5):

    C sends (x_j, c_j, m_j') to V. V accepts the coin C_j if and only if $h(c_j) = c_{j-1}$ (where c_{j-1} was obtained from the previous coin) and m_j' is a valid bank signature on $h(c_j \| g^{x_j})$.

For the sake of convenience, let us name the first coin C_i as *signed coin* and all the other coins C_j as *normal coins*.

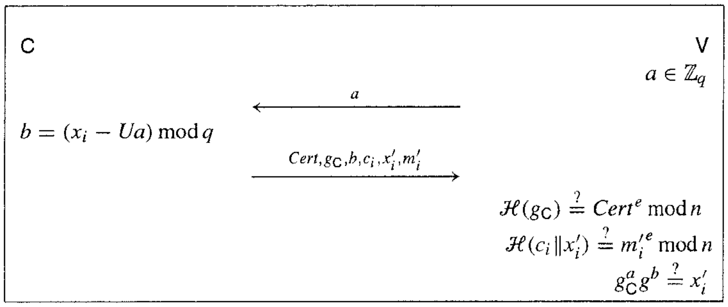

C
V
$a \in \mathbb{Z}_q$
a
$b = (x_i - Ua) \bmod q$
$Cert, g_C, b, c_i, x'_i, m'_i$
$\mathcal{H}(g_C) \stackrel{?}{=} Cert^e \bmod n$
$\mathcal{H}(c_i \| x'_i) \stackrel{?}{=} m'^e_i \bmod n$
$g^a_C g^b \stackrel{?}{=} x'_i$

Fig. 8.4 The payment protocol for the first coin

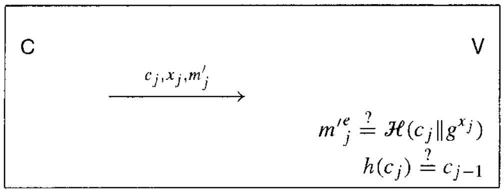

C
V
c_j, x_j, m'_j
$m'^e_j \stackrel{?}{=} \mathcal{H}(c_j \| g^{x_j})$
$h(c_j) \stackrel{?}{=} c_{j-1}$

Fig. 8.5 The payment protocol

Deposit phase. In deposit phase, V deposits all the received coins at B by sending $(Cert, g_C, a, b, c_i, x'_i, m'_i)$ for each signed coin and (c_j, x_j, m'_j) for each normal coin. B goes through exactly the same verification process as V did in the payment phase. If everything is OK, B pays V an equivalent amount of money and stores (a, b, c_i) for the first coin, (c_j, x_j) for each other coin in its coin database.

8.4.3 Discussion

In this section, we will closely examine security and efficiency features of the system, including double-spending detection, client anonymity, and efficiency.

Double spending. Double-spending occurs when C double spends some coins in the hope that B cannot detect the identity. In our protocol, double-spending is detected as follows:

When C double spends some coins, for the first double-spent coin C_i, it must be a signed coin in at least one transaction. So there are only two possibilities: C_i is spent as either a signed coin in the both transactions or as a signed coin in one transaction and as a normal coin in another transaction.

- Double spend a coin as signed coins: C spends C_i as a signed coin twice, i.e., for two different challenges a and a', B therefore has $b = x_i - Ua \mod q$ and $b' = x_i - Ua' \mod q$. B can easily find U by computing:

$$U = \frac{b - b'}{a' - a} \mod q$$

- Spend a coin as a signed coin and as a normal coin: C spends C_i twice, once as a normal coin and the other as a signed coin. B therefore has a and $x_i - Ua$ from the signed coin and x_i from the normal coin. This information is sufficient to compute U.

So in either case, the value U can be computed. After obtaining U, B extracts I and matches it with the client's ID stored in its database. Once, a match is found, B asks C to reveal the value U incorporated in his *Cert*. If this value matches the value U obtained by B from the first double-spent coin, C must have double spent the coin. The evidence is *undeniable* because U is client's secret information, which is infeasible for anyone else to compute unless the client had double spent a coin.

Anonymity. Client anonymity is protected unconditionally in this scheme. The zero-knowledge process used in the account's opening phase completely hides the identity of the client. The bank will not be able to link *Cert* to C's ID, once *Cert* is issued. On the other hand, our coins are blindly signed by the bank so the bank cannot trace any particular coin to any particular client.

During the payment process, the client only has to show *Cert*, which is an anonymous certificate and reveals $x_i - Ua \mod q$ for each signed coin and x_j for each normal coin. For two different coins, as their corresponding (x_i, x_j) are chosen at random, they are different and unlinkable. Having only a, B cannot obtain U from $x_i - Ua \mod q$ (since x_i is chosen at random).

Efficiency. The account's opening phase is a one-off process, so even though the zero-knowledge process is inefficient, it will not affect the efficiency of the system for any transaction later on.

The withdrawal phase is very efficient. To withdraw a coin, ignoring the number of hash operations, C computes only two exponentiations. The number of discrete exponentiations required in Chaum's [8.1], Ferguson's [8.6], Brands' [8.4] protocols are forty, seventeen, and ten, respectively. In contrast to these schemes, this protocol needs only two multiplication operations.

In the payment protocol, for the whole transaction, the client only has to compute a single response, i.e., $b = x_i - Ua \mod q$. This is far more efficient than all off-line digital-cash schemes known todate, especially as the response message does not involve any discrete exponential computation. Moreover, the vendor, in the payment phase, does not need to perform any complicated verification. In fact, the vendor only has to verify one RSA signature per coin plus a certification *Cert* and a Schnorr's one-time signature for each transaction.

Hence the protocol is much more efficient than other existing digital-cash schemes such as those in [8.1, 8.4, 8.6, 8.12].

8.5 Fair Digital Cash

Brickell, Gemmell, and Kravitz [8.16] proposed an escrowed digital-cash scheme to control unconditional privacy of clients, often known as *fair digital-cash.* The main feature of fair digital-cash is the existence of a trusted authority or a revocation authority that can revoke the anonymity of any given coin. A different and more efficient scheme was later proposed by Camenisch, Piveteau, and Stadler [8.27]. Both schemes require the revocation authority to be actively involved in every withdrawal and thus are not desirable.

Frankel, et. al. [8.25] and Camenisch, et. al. [8.20] respectively proposed a fair digital-cash scheme employing an off-line revocation authority. The advantage of this approach is that the revocation authority is not involved in any payment transaction. When needed, the revocation authority can be called upon to identify the owner of a coin or a transaction. The most efficient schemes to date are those by Davida, Frankel, Tsiounis, and Yung [8.24] and by Camenisch, Maurer, and Stadler [8.20]. Both of these two schemes are constructed from Brands' anonymous e-cash scheme.

In this section, we will not describe all fair-digital-cash schemes, whereas we introduce the concept of fair-digital-cash by using a typical model that uses an off-line revocation authority [8.28]. This fair-digital-cash scheme is based on Nyberg–Rueppel digital signature scheme and thus poses as an alternative to Schnorr-based fair-digital-cash schemes.

In a fair e-cash scheme, there are the following parties, a bank B, a trusted authority T, vendors, and clients. We denote by V a vendor and by C a client.

A fair e-cash scheme consists of five basic protocols, three of them are the same as in anonymous e-cash, i.e., a withdrawal protocol, a payment protocol, and a deposit protocol. The two additional protocols are conducted between B and T, *owner-tracing* and *coin-tracing* protocols.

- In the owner-tracing protocol, B gives to T the view of a deposit protocol and T returns a string that contains some specific information which allows B to identify the owner of the coin.
- In the coin-tracing protocol, B gives to T the view of a withdrawal protocol and T returns some specific information that allows B to identify the coin in the deposit phase.

These two additional protocols provide the revocation capacity and the protection against certain types of attacks. For instance, the owner-tracing protocol allows the authorities to identify the origin of dubious coins and thus eliminates money laundering. The coin-tracing protocol allows the authorities to find the destination of dubious coins and thus eliminates blackmailing.

In the following, we will first introduce a digital-cash scheme based on Nyberg–Rueppel digital-signature scheme[8.21] and then convert it to a fair version.

8.5.1 Nyberg-Rueppel Digital-Signature Scheme

The key generation protocol works as follows. Let p be a large prime and q be equal to $p-1$ or a large integer factor of $p-1$. Also let g be a random generator of $\mathbb{G}_q \subset \mathbb{Z}_p^*$. Each signer chooses $x \in_R \mathbb{Z}_q^*$ and computes $h = g^x \bmod p$, where (x, h) is his secret and secret-public key pair. Again, we omit the modulo p in our presentation.

To sign a message $m \in \mathbb{Z}_p$, the signer selects a random number $w \in \mathbb{Z}_q$ and computes r and s as

$$r = mg^w \text{ and } s = xr + w \bmod q.$$

The pair (r, s) is the signature of the message m. To verify the signature, we check

$$m = g^{-s}h^r r.$$

This signature scheme can be converted into a blind version using the protocol given in Fig. 8.6.

The pair (r', s') is a blind signature on message m. The correctness of the signature is shown as follows:

$$\begin{aligned} g^{-s'}h^{r'}r' &= mg^{-s\beta-\alpha+xr'+w\beta+\alpha} \\ &= mg^{-m'x\beta-w\beta+r'x+w\beta} \\ &= mg^{xr'-xm'\beta} = m. \end{aligned}$$

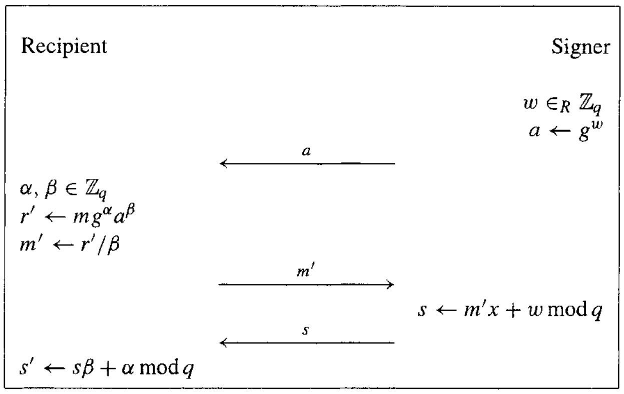

Recipient
Signer
$w \in_R \mathbb{Z}_q$
$a \leftarrow g^w$
a
$\alpha, \beta \in \mathbb{Z}_q$
$r' \leftarrow mg^{\alpha}a^{\beta}$
$m' \leftarrow r'/\beta$
m'
$s \leftarrow m'x + w \bmod q$
s
$s' \leftarrow s\beta + \alpha \bmod q$

Fig. 8.6 Blind Nyberg–Rueppel digital signature scheme

The blindness holds because if α and β are chosen at random, r' and m are uniformly distributed in their respective domains. As r' and m uniquely identify s', (r', s', m) is a random triplet and independent of the signer's view.

No apparent security weakness of this protocol is known. Some security proofs of this protocol have been discussed in [8.19, 8.30]. Particularly, [8.30] shows that the view of the signer in the protocol and the signature are statistically independent, i.e., generated signatures are witness-indistinguishable.

8.5.2 The Digital Cash Scheme

In this section, we take a look at a previously proposed digital-cash scheme, which is based on the blind Nyberg–Rueppel digital-signature scheme. [8.28]

The setup protocol. On inputting a security parameter k, the bank B runs a key generation algorithm to generate:

- a large prime p and a large number q such that $q|p-1$,
- three generators g, g_1 and g_2 of the unique subgroup $\mathbb{G}_q$ of the multiplicative group $\mathbb{Z}_p^*$,
- a randomly chosen collision-intractable hash function $\mathcal{H}(.)$ of polynomial size of k that maps its inputs to $\mathbb{Z}_q$,
- a random number $x \in \mathbb{Z}_q$ and
- $h_1 = g_1^x$ and $h_2 = g_2^x$.

B has now the secret key x and the public tuple $(p, q, g_1, g_2, h, h_1, h_2, \mathcal{H}(.))$ respectively.

The account setup. The setup phase is similar to those we studied previously in this chapter. To set up an account at B, C chooses $U \neq 0 \in_R \mathbb{G}_q$ at random and calculates $I = g_1^U$. B regards $I \neq 1$ as C's account identification and sends $z = (Ig_2)^x$ to C. Note that I is the unique link to the user's real name, while U is unknown to the bank. U can be computed by the bank only when the user double spends a coin.

The withdrawal protocol. The withdrawal protocol between C and B is given in Fig. 8.7.

C		B
$t, x_1, x_2 \in_R \mathbb{Z}_q^*$ $a' \leftarrow (Ig_2)^t$ $z' \leftarrow z^t$ $A \leftarrow g_1^{x_1} g_2^{x_2}$ $Z \leftarrow h_1^{x_1} h_2^{x_2}$		$w \in_R \mathbb{Z}_q$ $a \leftarrow (Ig_2)^w$
	$\xleftarrow{\quad a \quad}$	
$m \leftarrow \mathcal{H}(A, Z, a', z')$ $\alpha, \beta \in_R \mathbb{Z}_q^*$ $r' \leftarrow ma'^{\alpha} a^{t\beta}$ $m' \leftarrow r'/\beta \bmod q$		
	$\xrightarrow{\quad m' \quad}$	
		$s \leftarrow m'x + w \bmod q$
	$\xleftarrow{\quad s \quad}$	
$s' \leftarrow s\beta + \alpha \bmod q$		

Fig. 8.7 The withdrawal protocol

The blind Nyberg–Rueppel signature scheme is essential for the anonymity of clients. Note that the base used in the protocol is not the fixed base g of the signer public key, but the base $(Ig_2)^t$ for a random number t chosen by the user. At the end of the withdrawal protocol, the client should receive the blind Nyberg–Rueppel signature Sign$(A, Z) = (A, Z, z', a', r', s')$, which is verified using the equation

$$\mathcal{H}(A, Z, a', z') = a'^{-s'} z'^{r'} r'.$$

It is important to verify that the secret key used in the signature generation is the secret key x of the bank. Otherwise, the user can create such a signature using any secret key. This verification is described in the payment protocol.

The payment protocol. The payment protocol is run between C and V. The payment of a coin (A, Z, z', a', r', s') is described in Fig. 8.8. As for the proof of equality of discrete logarithms, for a random challenge c if

$$g_1^{r_1} g_2^{r_2} \overset{?}{=} a'^c A,$$
$$h_1^{r_1} h_2^{r_2} \overset{?}{=} z'^c Z,$$

we must have $\log_{a'} z' = \log_{g_1} h_1$. This shows that the bank's secret key $x = \log_{g_1} h_1$ was used in the generation of $\mathrm{Sign}(A, Z)$.

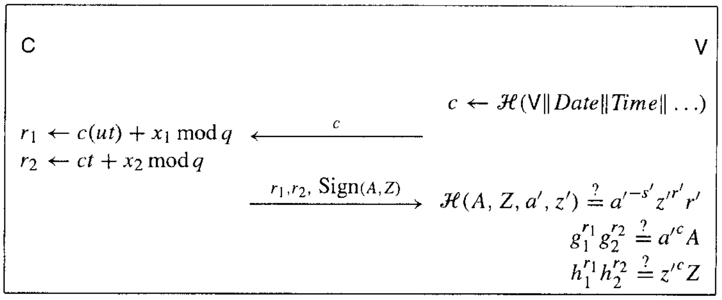

C
V
$c \leftarrow \mathcal{H}(\mathsf{V}\|Date\|Time\|\ldots)$
$r_1 \leftarrow c(ut) + x_1 \bmod q$
$\xleftarrow{c}$
$r_2 \leftarrow ct + x_2 \bmod q$
$\xrightarrow{r_1, r_2,\ \mathrm{Sign}(A,Z)}$
$\mathcal{H}(A, Z, a', z') \overset{?}{=} a'^{-s'} z'^{r'} r'$
$g_1^{r_1} g_2^{r_2} \overset{?}{=} a'^c A$
$h_1^{r_1} h_2^{r_2} \overset{?}{=} z'^c Z$

Fig. 8.8 The payment protocol

The deposit protocol. V can deposit the coin $\mathrm{Sign}(A, Z)$ at any suitable time. The deposit procedure is to send the transcript of the payment to B. B verifies the payment procedure and accepts the coin if V follows the procedure correctly and the coin satisfies all verifications as checked in the payment phase.

8.5.3 The Fair-Digital-Cash Scheme

We now convert the digital-cash scheme described previously in this section into a fair version.

The setup. The setup phase is similar to the original scheme. Here, we give only the difference. Let $x \in \mathbb{Z}_q$ be the secret key of B. The public data of B consists of tuple: $(p, q, g_1, g_2, g_3, h_1 = g_1^x, h_2 = g_2^x)$, where $(g_1, g_2, g_3) \in (\mathbb{Z}_p^*)^3$. To set up an account at B, C chooses $U \neq 0 \in_R \mathbb{G}_q$ at random and calculates $I = g_1^U$. B regards $I \neq 1$ as C's account identification and sends $z = (I g_2 g_3)^x$ to C.

The trusted authority T's secret key is $\tau \in \mathbb{Z}_q$ and his public key is the doublet $(h_{T1} = g_1^\tau, h_{T2} = g_2^\tau)$.

The withdrawal protocol. The withdrawal protocol is executed between C and B. Formally, the protocol is given in Fig. 8.9.

C		B
$t, x_1, x_2 \in_R \mathbb{Z}_q^*$ $d \leftarrow I g_2 g_3^{t^{-1}}$ $e \leftarrow h_{T1}^{Ut} h_{T_2}^{t}$		$w \in_R \mathbb{Z}_q$
	$\xrightarrow{d, e, g_3^{t^{-1}}}$	
	$\mathsf{C} \Rightarrow \mathsf{B} : PK\{(U, t) : d = g_1^U g_2 g_3^{1/t} \wedge e = h_{T1}^{Ut} h_{T2}^{t}\}$	
$a' \leftarrow d^t$ $A_1 \leftarrow g_1^{x_1}$ $A_2 \leftarrow g_2^{x_2}$ $Z \leftarrow h_1^{x_1} h_2^{x_2}$		$a \leftarrow d^w$ $z \leftarrow (I g_2 g_3^{t^{-1}})^x$
	$\xleftarrow{a, z}$	
$z' \leftarrow z^t$ $m \leftarrow \mathcal{H}(A_1, A_2, Z, a', z')$ $\alpha, \beta \in_R \mathbb{Z}_q^*$ $r' \leftarrow m a'^{\alpha} a^{t\beta}$ $m' \leftarrow r'/\beta \bmod q$		
	$\xrightarrow{m'}$	
		$s \leftarrow m'x + w \bmod q$
	$\xleftarrow{s}$	
$s' \leftarrow s\beta + \alpha \bmod q$		

Fig. 8.9 The withdrawal protocol

The withdrawal protocol is basically the blind Nyberg–Rueppel digital signature scheme but the base is $d^t = (Ig_2)^t g_3$. At the end of the withdrawal protocol, the user should receive the blind Nyberg–Rueppel signature

$$\mathrm{Sign}(A_1, A_2, Z) = (A_1, A_2, Z, z', a', r', s')$$

which is verified using the equation

$$\mathcal{H}(A_1, A_2, Z, a', z') = a'^{-s'} z'^{r'} r'.$$

Besides $(A_1, A_2, Z, z', a', r', s')$, the bank also stores the value e, given below, in the coin database for referencing.

In this protocol, the value A is split into two values A_1 and A_2. This is necessary to achieve owner tracing. The extra computation of (d, e) and the associate proof of knowledge

$$PK\{(U,t) : d = g_1^U g_2 g_3^{1/t} \wedge e = h_{T1}^{Ut} h_{T2}^t\},$$

will be used to achieve coin tracing, where we meant that the prover proves his knowledge on (U, t) from the given (d, e) without revealing the value of (U, t). This notation will also be used later on. The details of the proofs can be found in the Appendix.

The payment protocol. The payment protocol is run between C and V. The payment of a coin $(A_1, A_2, Z, z', a', r', s')$ is described in Figure 8.10. The payment protocol is similar to the payment protocol of the original scheme, whereas C now splits a' into $a_1 = I^t = g_1^{Ut}$ and $a_2 = g_2^t$ and proves to V that

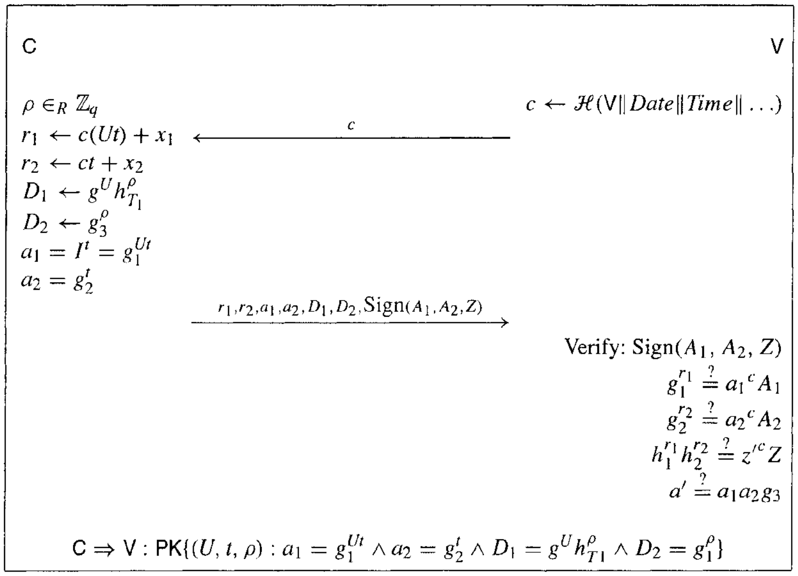

C
V
$\rho \in_R \mathbb{Z}_q$
$c \leftarrow \mathcal{H}(\mathsf{V}\|Date\|Time\|\ldots)$
c
$r_1 \leftarrow c(Ut) + x_1$
$r_2 \leftarrow ct + x_2$
$D_1 \leftarrow g^U h_{T_1}^{\rho}$
$D_2 \leftarrow g_3^{\rho}$
$a_1 = I^t = g_1^{Ut}$
$a_2 = g_2^t$
$r_1, r_2, a_1, a_2, D_1, D_2, \mathrm{Sign}(A_1, A_2, Z)$
Verify: $\mathrm{Sign}(A_1, A_2, Z)$
$g_1^{r_1} \overset{?}{=} a_1{}^c A_1$
$g_2^{r_2} \overset{?}{=} a_2{}^c A_2$
$h_1^{r_1} h_2^{r_2} \overset{?}{=} z'^c Z$
$a' \overset{?}{=} a_1 a_2 g_3$
$\mathsf{C} \Rightarrow \mathsf{V} : \mathsf{PK}\{(U, t, \rho) : a_1 = g_1^{Ut} \wedge a_2 = g_2^t \wedge D_1 = g^U h_{T1}^{\rho} \wedge D_2 = g_1^{\rho}\}$

Figure 8.10 The payment protocol

$$\mathsf{PK}\{(U, t, \rho) : a_1 = g_1^{Ut} \wedge a_2 = g_2^t \wedge D_1 = g^U h_{T1}^{\rho} \wedge D_2 = g_1^{\rho}\}.$$

This proof is used to achieve user tracing.

The deposit protocol. V can deposit the coin $\mathrm{Sign}(A_1, A_2, Z)$ at any suitable time. The deposit procedure is to send the transcript of the payment to B. B verifies the payment procedure and accepts the coin if V follows the procedure correctly and the coin satisfies all verifications as checked in the payment phase.

The completeness of the scheme is straightforward. As the scheme is developed using our proposed anonymous-digital-cash scheme, it is easy to show that this fair-digital-cash scheme satisfies all security requirements of the anonymous e-cash scheme, e.g., unforgeability. It remains to show that user tracing and coin tracing can be satisfied.

Anonymity revocation. There are two possible anonymity controls in this scheme. One is to identify the user in a payment transaction and the other is to identify the history, i.e., the life cycle of a coin. The former is referred to as user tracing and the latter is referred as coin tracing. In practice, T should only run these protocols under a court order. Formally, the user-tracing and coin-tracing protocols work as follows:

Client Tracing

To identify the client in a payment transaction, B brings (D_1, D_2) to T who then computes

$$D_1/D_2^{\tau} = g^U h_{T1}^{\rho}/g_1^{\tau\rho} = g^U,$$

which identifies the client.

The soundness of this protocol is due to the proof of knowledge

$$\mathsf{PK}\{(U, t, \rho) : a_1 = g_1^{Ut} \wedge a_2 = g_2^t \wedge D_1 = g^U h_{T1}^{\rho} \wedge D_2 = g_1^{\rho}\},$$

which shows g^U is the plaintext corresponding to the ElGamal ciphertext (D_1, D_2) encrypted using T's public key for the client secret information U. In the client-tracing protocol, T simply decrypts the ciphertext and returns g^U which identifies the client.

Note that this procedure is not possible for other parties as only T can decrypt a ciphertext encrypted using T's public key.

Coin Tracing

Identifying a coin history can be done in two different ways. One is to identify the coin payment for a given coin withdrawal and the other is to identify the coin withdrawal for a given coin payment.

In the later case, B sends to T the payment transcript. Then T computes the value

$$a'/{g_3}^{\tau} = ((Ig_2)^t)^{\tau} = g_1^{Ut\tau} g_2^{t\tau} = h_{T1}^{ut} h_{T2}^{t} = e,$$

and sends the value e back to B. The anonymity revocation is done by searching for the computed value e in the coin withdrawal reference database.

In the former case, B sends to T the withdrawal reference e. Then T computes and sends to B the value

$$e^{1/\tau} g_3 = h_{T1}^{Ut/\tau} h_{T2}^{t/\tau} g_3 = g_1^{Ut} g_2^{t} g_3 = a'.$$

Now, the anonymity revocation is done by matching this computed value a' with the value a' in every deposited coin.

8.6 Summary

There are various digital-cash protocols having been proposed in past two decades. We can refer them to as three forms: normal digital-cash (e.g., [8.1, 8.2]), divisible digital-cash [8.7], and fair digital-cash (e.g., [8.14]). In this chapter, we have described three typical digital-cash schemes: the digital-cash scheme proposed by Brands[8.2], and the digital-cash and the fair digital-cash scheme proposed by Nguyen et. al. [8.14] We hope that these schemes give the reader an overall picture of digital-cash.

8.7 Appendix

This section gives protocols for proving the knowledge of various discrete logarithms. Some of these protocols presented in this section are borrowed from Camenisch [8.17], which gives a rigorous treatment of proofs of knowledge about discrete logarithms. The interactive versions of these protocols are known to be witness-indistinguishable and proofs of knowledge. The reader is also referred to [8.15, 8.18, 8.26, 8.29] for detailed discussions of these protocols and other variations.

In the following, we assume that $g, h_1, h_2, g_1, \ldots, g_m \in \mathbb{G}_q$ $(\subset \mathbb{Z}_p^*)$ are generators of order q such that computing a representation of any generator with respect to other generators is infeasible.

Proving the Knowledge of Discrete Logarithms. A proof of knowledge of the discrete logarithm proves the knowledge of the secret number $x \in \mathbb{Z}_q$ from $y = g^x$. This proof is actually part of the Schnorr identification scheme. Following the the notations of [8.17, 8.18], we denote this protocol as

$$\mathsf{PK}\{(\alpha) : y = g^{\alpha}\}.$$

The proof is straightforward. We omit it here.

Proving the knowledge of a Representation. The proof of knowledge of a representation proves the knowledge of a representation of y to the bases $g_1, \ldots, g_m$, which is denoted by

$$\mathsf{PK}\{(x_1,\ldots,x_m) : y = \prod_{i=1}^{m} g_i^{x_i}\}.$$

This proof is first introduced in [8.22] and is given in Fig. 8.11.

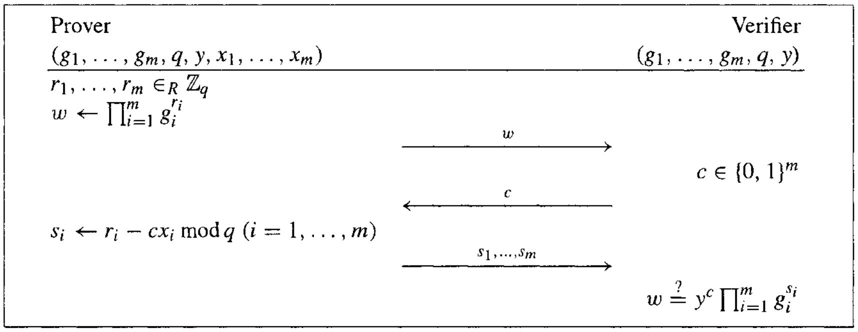

Prover
$(g_1,\ldots,g_m,q,y,x_1,\ldots,x_m)$
Verifier
$(g_1,\ldots,g_m,q,y)$
$r_1,\ldots,r_m \in_R \mathbb{Z}_q$
$w \leftarrow \prod_{i=1}^{m} g_i^{r_i}$
$\xrightarrow{w}$
$c \in \{0,1\}^m$
$\xleftarrow{c}$
$s_i \leftarrow r_i - cx_i \bmod q \; (i = 1,\ldots,m)$
$\xrightarrow{s_1,\ldots,s_m}$
$w \overset{?}{=} y^c \prod_{i=1}^{m} g_i^{s_i}$

Fig. 8.11 A proof of representation of y to the bases $g_1,\ldots,g_m$

The correctness of this protocol is due to

$$y^c \prod_{i=1}^{m} g_i^{s_i} = \prod_{i=1}^{m} g_i^{cx_i} \prod_{i=1}^{m} g_i^{s_i} = \prod_{i=1}^{m} g_i^{cx_i+s_i} = \prod_{i=1}^{m} g_i^{r_i} = w.$$

The soundness is due to the fact that given a same value w, if the prover can answer two different challenges c and c' correctly, the knowledge extractor obtains two sets of $(c, s_1,\ldots,s_m)$ and $(c', s_1',\ldots,s_m')$ and extract the secret x_i as:

$$x_i = \frac{s_i - s_i'}{c' - c} \bmod q.$$

The zero-knowledge holds because a honest verifier can construct a valid view by choosing $s_1,\ldots,s_m$ and c at random and computing $w = y^c \prod_{i=1}^{m} g_i^{s_i}$.

Proving the Equality of Discrete Logarithms. This proof proves not only the knowledge of secret keys but also certain relations among them. In the most simplest form, it is a proof of knowledge and of equality of discrete logarithm of y_1 to the base g_1 and y_2 to the base g_2. This proof was first introduced by Chaum and Pedersen in [8.23]. Let us denote this protocol by:

$$\mathsf{PK}\{(\alpha) : y_1 = g_1^{\alpha} \wedge y_2 = g_2^{\alpha}\}.$$

The intuition is to run the proof of knowledge of discrete logarithm of y_1 to the base g_1 and of discrete logarithm of y_2 to the base g_2, and then $\log_{g_1} y_1 = \log_{g_2} y_2$ only if the prover can return the same answer in both cases for a random challenge chosen by the verifier. Formally, this protocol is given in Fig. 8.12.

Prover		Verifier
$(g_1, g_2, q, y_1, y_2, x)$		(g_1, g_2, q, y_1, y_2)
$r \in_R \mathbb{Z}_q$ $w_1 \leftarrow g_1^r$ $w_2 \leftarrow g_2^r$		
	$\xrightarrow{w_1, w_2}$	
		$c \in \{0, 1\}^k$
	$\xleftarrow{c}$	
$s \leftarrow r - cx \bmod q$		
	$\xrightarrow{s}$	
		$w_i \overset{?}{=} y_i^c g_i^s \; (i = 1, 2)$

Fig. 8.12 The proof of $\log_{g_1}(y_1) \equiv \log_{g_2}(y_2)$

It is trivial to extend the proof system of equality of discrete logarithms to a proof system of equality of representations. One of such proof is the proof of knowledge of representation of y_1 and y_2 to the bases (g_1, h_1) and (g_2, h_2), respectively and that the representation of y_1 to g_1 and y_2 to g_2 are equal. This protocol, which is denoted by

$$\mathsf{PK}\{(\alpha, \beta_1, \beta_2) : y_1 = g_1^{\alpha} h_1^{\beta_1} \wedge y_2 = g_2^{\alpha} h_2^{\beta_2}\},$$

is described in Fig. 8.13. This proof introduced in [8.23] is the basic building block for many blind digital signatures and anonymous-digital-cash schemes.

Prover		Verifier
$(g_1, g_2, h_1, h_2, q, y_1, y_2, \alpha, \beta_1, \beta_2)$		$(g_1, g_2, h_1, h_2, q, y_1, y_2)$
$r, \rho_1, \rho_2 \in_R \mathbb{Z}_q$ $w_1 \leftarrow g_1^r h_1^{\rho_1}$ $w_2 \leftarrow g_2^r h_2^{\rho_2}$		
	$\xrightarrow{w_1, w_2}$	
		$c \in \{0, 1\}^k$
	$\xleftarrow{c}$	
$s \leftarrow r - c\alpha \bmod q$ $\sigma_i \leftarrow \rho_i - c\beta_i \; (i = 1, 2)$		
	$\xrightarrow{s, \sigma_1, \sigma_2}$	
		$w_i \overset{?}{=} y_i^c g_i^s h_i^{\sigma_i} \; (i = 1, 2)$

Fig. 8.13 The proof of equality of representations

Proving knowledge of inverse of discrete logarithms. In this protocol, we give the proof for: given $y_1 = g_1^t$ and $y_2 = g_2^{t^{-1}}$, Proving that $(\log_{g_1} y_1)^{-1} = \log_{g_2} y_2$. The protocol is given in Figure 8.14.

$$\begin{array}{lcr}
\text{Prover} & & \text{Verifier} \\
b \in_R \mathbb{Z}_q & & \\
a_1 = g_1^b, \quad a_2 = y_2^b & & \\
c = \mathcal{H}(y_1, y_2, a_1, a_2) & & \\
w = ct + b & \xrightarrow{a_1, a_2, c, w} & \\
 & & g_1^w \overset{?}{=} y_1^c a_1 \\
 & & y_2^w \overset{?}{=} g_2^c a_2
\end{array}$$

Fig. 8.14 A proof of equality of inverse of discrete logs

8.8 References

[8.1] D. Chaum, A. Fiat, M. Naor (1988) Untraceable electronic cash. In: Advances in Cryptology – CRYPTO'88. Springer, Berlin Heidelberg New York, pp. 319–327.

[8.2] S. Brands (1993) Untraceable off-line cash in wallet with observers. In: Advances in Cryptology – CRYPTO'93. Springer, Berlin Heidelberg New York, pp. 302–318.

[8.3] L. A. M. Schoenmakers (1995) An efficient electronic payment system withstanding parallel attacks. Technical Report CS-R9522, CWI.

[8.4] S. Brands (1993) Untraceable off-line cash in wallet with observers. In: Advances of Cryptology – CRYPTO'93. Springer, Berlin Heidelberg New York, pp. 302–318.

[8.5] D. Chaum (1985) Security without identification: transaction systems to make Big Brother obsolete. Commun ACM 28(10):1030–1044.

[8.6] N. T. Ferguson (1993) Single term off-line coins. In: Advances in Cryptology – EUROCRYPT'93. Springer, Berlin Heidelberg New York, pp. 318–328.

[8.7] T. Okamoto, K. Ohta (1991) Universal electronic cash. In: Advances in Cryptology – CRYPTO'91. Springer, Berlin Heidelberg New York, pp.324–337.

[8.8] T. Eng, T. Okamoto (1994) Single-term divisible electronic coins. In: Advances in Cryptology – EUROCRYPT'94. Springer,Berlin Heidelberg New York, pp. 306–319.

[8.9] T. Okamoto (1995): An efficient divisible electronic cash scheme. In: Advances in Cryptology – CRYPTO'95. Springer, Berlin Heidelberg New York, pp. 438–451.

[8.10] W. Mao (1996) Blind certification of public keys and off-line electronic cash. Technical Report HPL-96-71, HP Laboratories.

[8.11] C. Schnorr (1989) Efficient identification and signatures for smart cards. In: Advances in Cryptology – CRYPTO'89, LNCS 435. Springer, Berlin Heidelberg New York, pp. 239–252.

[8.12] Y. Yacobi (1994) An efficient off-line cash. In: Advances in Cryptology – Asiacrypt'94. Springer, Berlin Heidelberg New York.

[8.13] K. Q. Nguyen, Y. Mu, V. Varadharajan (1997) One-response off-line digital coins. In: Proceedings of Fourth Annual Workshop on Selected Areas in Cryptography (SAC'97), Canada, 1997, pp. 244–251.

[8.14] Y. Mu, K. Q. Nguyen, V. Varadharajan (2001) A fair electronic cash scheme. In: W. Kou, et al. (eds.) Electronic commerce technologies – ISEC2001, LNCS 2040. Springer, Berlin Heidelberg New York, pp. 20–32.

[8.15] S. Brands (1997) Rapid demonstration of linear relations connected by boolean operators. In: Advances in Cryptology – Eurocrypt'97, LNCS 1223. Springer, Berlin Heidelberg New York, pp. 318–333.

[8.16] E. F. Brickell, P. Gemmell, D. Kravitz (1995) Trustee-based tracing extensions to anonymous cash and the making of anonymous change. In: Symposium on Distributed Algorithms. Albuquerque, NM. http://www.cs.sandia.gov/psgemme/

[8.17] J. Camenisch (1998) Group signature schemes and payment systems based on the discrete logarithm problems. Ph.D. thesis, Swiss Federal Institute of Technology, Zurich.

[8.18] J. Camenisch, M. Michels (1999) Proving in zero-knowledge that a number is the product of two safe primes. Technical Report RS-98-29, BRICS. An abstract version appeared in Proceeding of Eurocrypt'99, LNCS 1592. Springer, Berlin Heidelberg New York, pp. 106–121.

[8.19] J. Camenisch, J. M. Piveteau, M. Stadler (1994) Blind signatures based on the discrete logarithm problem. In: Advances in Cryptology – Eurocrypt'94, LNCS 950. Springer, Berlin Heidelberg New York, pp. 428–432.

[8.20] M. Camenisch, U. Maurer, M. Stadler (1996) Digital payment systems with passive anonymity-revoking trustees. In: ESORICS'96, LNCS 1146. Springer, Berlin Heidelberg New York, pp. 33–43.

[8.21] K. Nyberg, R. A. Rueppel (1995) Message recovery for signature schemes based on the discrete logarithm problem. In: Advances in Cryptology – Eurocrypt'94, LNCS 950. Springer, Berlin Heidelberg New York, pp. 182-193.

[8.22] D. Chaum, J. Evertse, J. Graaf (1988) An improved protocol for demonstrating possession of discrete logarithms and some generalizations. In: Advances in Cryptology – EUROCRYPT'87, LNCS 304. Springer, Berlin Heidelberg New York, pp. 127–141.

[8.23] R. Cramer, T. P. Pedersen (1993) Improved privacy in wallets with observers. In: Advances in Cryptology – Eurocrypt'93, LNCS 765. Springer, Berlin Heidelberg New York, pp. 329–343.

[8.24] G. Davida, Y. Frankel, Y. Tsiounis, M. Yung (1997): Anonymity control in e-cash. In: Proceedings of First Financial Cryptography conference, LNCS 1318. Springer, Berlin Heidelberg New York.

[8.25] Y. Frankel, Y. Tsiounis, M. Yung (1996) Indirect discourse proofs: achieving fair off-line e-cash. In: Advances in Cryptology – ASIACRYPT'96, LNCS 1163. Springer, Berlin Heidelberg New York, pp. 286–300.

[8.26] E. Fujisaki, T. Okamoto (1997) Statistical zero-knowledge protocols to prove modular polynomial relation.In: Advances in Cryptology – CRYPTO'97, LNCS 1294. Springer, Berlin Heidelberg New York, pp. 16–30.

[8.27] J. Piveteau, J. Camenisch, M. Stadler (1994) An efficient payment system protecting privacy. In: Computer Security – ESORICS'94, LNCS 875. Springer, Berlin Heidelberg New York, pp. 207–215.

[8.28] K. Q. Nguyen, Y. Mu, V. Varadharajan (1997) A new digital cash scheme based on blind Nyberg-Rueppel digital signature. In: Information Security Workshop, LNCS 1396. Springer, Berlin Heidelberg New York, pp. 312–320.

[8.29] T. Okamoto (1995) An efficient divisible electronic cash scheme. In: Advances in Cryptology – CRYPTO'95, LNCS 963. Springer, Berlin Heidelberg New York, pp. 439–451.

[8.30] M. Stadler (1996) Cryptographic protocols for revocable privacy. Ph.D. thesis, Swiss Federal Institute of Technology, Zurich.

[8.31] B. Solms, D. Naccache (1992) On blind signatures and perfect crimes. Computer and Security 11(6): 581–583.

9 Digital Checks

Bo Yang

National Key Laboratory for ISN
Xidian University, Xi'an, China

9.1 Introduction

In electronic commerce, there is a need for a check-like payment system where funds are transferred from the payer's bank account to the payee's bank account at the time the transaction takes place. From the bank's point of view, it would be desirable to use existing interbank funds-transfer networks as much as possible. This chapter will introduce the foundational concept of digital check and two important electronic-check systems: NetBill and NetCheque.

9.2 Digital Check Concept

9.2.1 Digital Check's Basic Elements

As with its paper counterpart, the digital check will contain an instruction to the payer's bank to make a payment of a specified amount to an identified payee. The fact that the check is in electronic form and is being conveyed across computer networks should allow more flexibility in the handling of the check. New services can be provided, such as the ability to immediately verify funds availability. Allowing digital-signature validation can enhance security, and check payments can more easily be integrated into electronic ordering and billing processes.

The concept of digital checking can be described using Fig. 9.1. There are five parties in the system: the customer, the merchant, the consumer's bank, the merchant's bank, and clearing house, in which the clearing house processes checks among different banks. The functions described for a clearing house may be handled by a separate entity or by an existing banking system. For simplicity, we have not included the online malls.

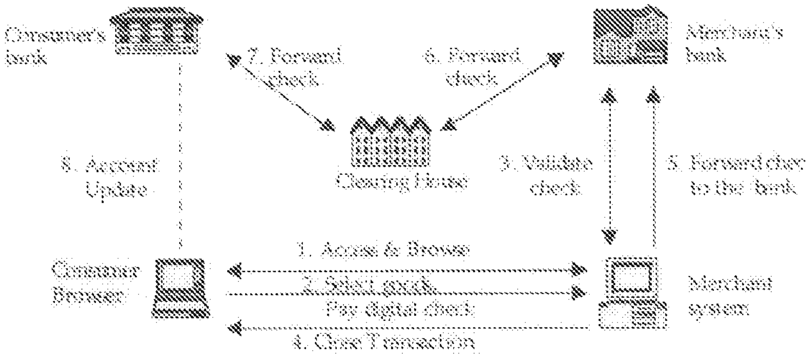


Fig. 9.1 A digital check system

The consumer uses a web browser that has access to various web servers over the Internet. The consumer views various shopping malls and storefronts at the browser. The browser has provisions for displaying the digital check formats. The banks process digital checks which are similar to paper checks.

A complete digital check transaction may consist of several basic steps outlined next. These steps are executed in three distinct and optionally separate phases. In the first phase, the consumer makes a purchase. In the second phase, the merchant sends the digital checks to its bank for redemption. In the third phase, the merchant's bank approaches the clearing house or the consumer's bank to cash the digital checks.

Phase 1: Purchasing goods

1. The consumer accesses the merchant server, and the merchant server presents its goods to the consumer.

2. The consumer selects the goods and purchases them by sending a digital check to the merchant. The check can be transported in some kind of secure envelope; the form of this envelope is outside the architecture and could be sent in a secure email or in an encrypted interactive dialogue between the two parties.

3. The merchant may validate the digital check with its bank for payment authorization, and endorses the check.

4. Assuming the check is validated.

Phase 2: Depositing checks at the merchant's bank

5. The merchant electronically forwards the checks to its bank. This action takes place at the discretion of the merchant.

Phase 3: Clearing the checks among the banks

6. The merchant's bank forwards the digital checks to the clearing house for cashing. The processing is identical to that undergone by any paper check today. This means that the banks involved would clear the check using the normal *automated clearing house* (ACH) or *electronic check presentment* (ECP) methods.

7. The clearing house works with the consumer's bank, clears the check, and transfers money to the merchant's bank, which updates the merchant's account.

8. At a later time, the consumer's bank updates the consumer with the withdrawal information.

9.2.2 Security schemes for digital checks

The security requirements for digital checks consist of authenticating the digital check, supplying the originator's public key to receiver, and securely storing the originator's private key.

Authenticity of Digital Checks

The digital check may consist of a document that is signed by the consumer's private key. The receiver (the merchant or the merchant's bank) uses the payer's public key to decrypt the digital signature. This assures the receiver that the sender indeed signed the check. It also provides for non-repudiation, such that the payer cannot deny issuing the check since it is signed by the payer's private key (that only the payer is expected to possess).

Additionally, the digital check also may require the digital signatures of the originator's bank. This step will assure the receiver that the check is written on a valid bank account. The receiver (or receiver's bank) can validate the authenticity of the originator's bank by using the public key of the originator's bank.

For large sums of money, additional security requirements may be levied.

Delivering Public Keys

The originator as well as the originator's bank must provide their public keys to the receiver. Attaching their X.509 certificates to the digital checks can provide the public keys. These certificates may use certificate chains including the signatures of the root CA. The public key of the root CA should be well publicized to avoid fraud.

Storage of Private Keys

To avoid fraud, the consumer's private key needs to be securely stored and made available to the consumer. This can be achieved by providing a smart card that the consumer can carry.

Cashier's Checks

Finally, a cashier's check may be issued by a bank as follows. The check is created by a bank and is signed using the bank's private key. The originating bank includes its certificate with the digital check. The receiving bank uses the originating bank's public key to decrypt the digital signature. In this way, the receiving bank is assured that the cashier check indeed was originated by the name of the bank indicated on the check. It also provides the receiving bank with non-repudiation such that the originating bank cannot deny issuing this check since it is signed by the originating bank's private key (that only the originating bank is expected to possess).

9.2.3 Benefits and concerns

Compared to paper checks and other forms of payments, digital checking provides the following advantages:

- **Time saved**:
  Digital checks can be issued without needing to fill out, mail, or deliver checks. It also saves time in processing the checks. With paper checks, the merchant collects all the checks and deposits them at the merchant's bank. With digital checks, the merchant instantly can forward checks to the bank and get them credited to their account. As such, digital checks can greatly reduce the time from the moment a consumer writes a check to the time when the merchant receives the deposit.

- **Deduced paper handling cost**:
  There is no need for long lines at the banks on the first day of the month, or for long lines of students paying their tuition at the university. Corre-

spondingly, it reduces the bank employees' effort to receive the checks, process them, and mail the cancelled checks to the consumers.

- **Reduction in bounced checks**:
  Digital checking can be designed in such a way that the merchant can get authorization from the customer's bank before accepting the digital check. Digital checks can be used to give gifts or make payments without the fear of being lost or stolen. If a check is stolen, the receiver can request the payer to stop the payment. On the other hand, digital cash is exposed to theft and other risks.

9.3 NetBill

NetBill is a payment system for the selling and delivery of low-priced information goods. A customer, represented by a client computer, wishes to buy information from a merchant's sever. A account server (the NetBill server), maintains accounts for both customers and merchants, linked to conventional financial institutions, A NetBill transaction transfers information goods from merchant to customer, debiting the customer's NetBill account and crediting the merchant's account for the value of the goods. When necessary, funds in a customer's NetBill account can be replenished from a bank or credit card; similarly, funds in a merchant's NetBill account are made available by depositing them in the merchant's bank account. NetBill acts as an aggregator to combine many small transactions into larger conventional transactions, amortizing conventional overhead fees.

The transfer of information goods consists of delivering bits to the customer. Users may be charged on a per-item basis, by a subscription allowing unlimited access, or by a number of other pricing models.

9.3.1 The NetBill Transaction Model

The NetBill transaction model involves three parties: the customer, the merchant, and the NetBill transaction server. A transaction involves three phases: price negotiation, goods delivery, and payment. For information goods, which can be delivered over the network, the NetBill protocol links goods delivery and payment into a single atomic transaction.

In a NetBill transaction, the customer and merchant interact with each other in the first two phases; the NetBill server is not involved until the payment phase, when the merchant submits a transaction request. The customer contacts the NetBill server directly only in the case of communications failure or when requesting administrative functions. Fig. 9.2 shows the relationships among parties in a NetBill transaction.

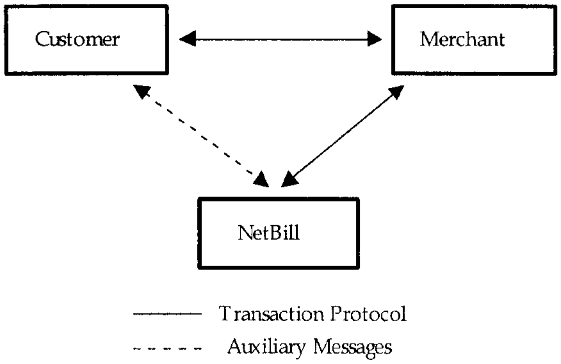


Fig. 9.2 Parties in a NetBill transaction

Transaction Objectives

NetBill transaction can obtain the following set of objectives.

a) Only authorized customers can charge against a NetBill account.
b) The customer and merchant must agree on the item to be purchased and the price to be charged.
c) A customer can optionally protect his identity from merchants.
d) Customers and merchants are provided with proof of transaction results from NetBill.

In addition, NetBill can also obtain the following objectives to support price negotiation and goods delivery.

e) There is an offer and acceptance negotiation phase between customer and merchant.
f) A customer may present credentials identifying them as entitled to special pricing or treatment.
g) A customer receives the information goods he purchases if and only if they are charged (and thus the merchant is paid) for the goods.
h) A customer may need approval from a fourth (access control) party before the NetBill server will allow a transaction.

Finally, as a general objective for all phases of the purchase process, the following objective can be added:

i) The privacy and integrity of communications is protected from observation or alteration by external parties.

To achieve these goals, the NetBill protocol provides for strong authentication and privacy, atomic payment and delivery protocol, and a flexible access control system.

In the price- negotiation phase, the customer presents evidence of their identity, and (optionally) supplemental credentials, and requests a price quote on an item. The customer may also include a bid for the item. The merchant responds with a price offer.

In the second phase, the customer accepts or declines the offer. In the case of information goods, acceptance constitutes an order for network delivery. The merchant provisionally delivers the goods, under encryption, but withholds the key.

Key delivery is linked to completion of the third phase, the payment protocol. In this phase, the customer constructs and digitally signs an *electronic payment order* (EPO), and sends it to the merchant. The merchant appends the key to the EPO and endorses (digitally signs) the EPO, forwarding it to the NetBill server. The NetBill server returns a digitally signed receipt, which includes the key, to the merchant, who forwards a copy to the customer.

9.3.2 The Transaction Protocol

We use the notation $X \Rightarrow Y$ to indicate that X sends the specified message to Y. The basic protocol involves three phases that can be divided into eight steps, where C, M, and N represent respectively customer, merchant and NetBill.

1. $C \Rightarrow M$ Price request
2. $M \Rightarrow C$ Price quote
3. $C \Rightarrow M$ Goods request
4. $M \Rightarrow C$ Goods, encrypted with a key K
5. $C \Rightarrow M$ Signed Electronic Payment Order
6. $M \Rightarrow N$ Endorsed EPO (including K)
7. $N \Rightarrow M$ Signed result (including K)
8. $M \Rightarrow C$ Signed result (including K)

The Price-Request Phase

The price-request phase consists of step 1 and step 2, which present a request/response message pair in which the customer requests a price quote of the merchant. The customer presents an identifying ticket (the identity presented may be a pseudonym) to the merchant, along with some optional credentials establishing their membership in groups which may make their eligible for a discount.

The customer passes parameters indicating a request for the disposition of the transaction. The merchant, on receiving the request for a quotation, determines a price for the user and returns a quotation.

Step 1 and 2 may be repeated as needed until customer and merchant can agree on a price.

The Goods-Delivery Phase

Once the customer and merchant have negotiated a price for the goods in question, the customer directs the merchant to deliver the goods in step 3.

In step 4, the merchant generates a unique symmetric cipher key *K*, encrypts the goods using this key and sends the encrypted goods to the customer, along with a cryptographic checksum computed on the encrypted goods, so that the customer will immediately detect any discrepancy before proceeding. The merchant also sends an *electronic payment order ID* (EPOID), with the goods. The EPOID is a globally unique identifier that will be used in the NetBill server's database to uniquely identify this transaction. It consists of three fields: a field identifying the merchant, a timestamp marking the time at the end of goods delivery, and a serial number to guarantee uniqueness.

The specification that the EPOID must be globaly unique is used to prevent replay attacks, in which unscrupulous merchants reuse customers' old signed payment instructions. The time stamp portion of the EPOID is used to expire stale transactions; it must be generated at the end of goods delivery because the delivery (especially for very large goods) may take longer than the payment expiration time.

Because the goods are delivered encrypted in step 4, the customer cannot use them. The key *K* needed to decrypt the goods will be delivered in the payment phase, which follows.

The Payment Phase

After the encrypted goods are delivered, the customer submits payment to the merchant in the form of a signed *electronic payment order* (EPO), in step 5. At any time before the signed EPO is submitted, a customer may abort the transaction and be in no danger of its being completed against their will. The submission of the signed EPO marks the "point of no return" for the customer.

An EPO consists of two sections, a clear part containing transaction information that is readable by the merchant and the NetBill server, and an encrypted part containing payment instructions that is readable only by the NetBill server.

After the customer presents the signed EPO to the merchant, the merchant endorses it and forwards the endorsed EPO to the NetBill server in step 6. The endorsed EPO adds the merchant's account number, the merchant's memo field, and the goods decryption key, as well as the merchant's signature.

At any time before the endorsed EPO is submitted to the NetBill server, the merchant may abort the transaction and be in no danger of its being completed against their will. The submission of the endorsed EPO marks the "point of no return" for the merchant.

Upon receipt of the signed and endorsed EPO, the NetBill server makes a decision about the transaction and returns the result to the merchant, who in turn forwards it to the customer.

The NetBill server makes its decision based on verification of the signatures, the privileges of the users involved, the customer's account balance, and the uniqueness and freshness of the EPOID. It then issues a receipt containing the result code, the identities of the parties, the price and description of the goods, the EPOID, and the key K needed to decrypt the goods. The receipt is digitally signed by the NetBill server, using the *digital signature algorithm* (DSA).

This receipt is returned to the merchant in step 7, along with an indication of the customer's new account balance (encrypted so that only they may read it). The EPOID is repeated in the customer-specific data to ensure that the merchant cannot replay data from an earlier transaction.

In step 8, the merchant responds to the request from the customer in step 5, forwarding the messages returned by the NetBill server in step 7.

9.3.3 Identities and Authentication

When a customer creates a NetBill account, they receive a unique User ID and generate the RSA public key pair associated with that User ID. This key pair is certified by NetBill, and is used for signatures and authentication within the system. In [9.1], the authors proposed that symmetric cryptography be used instead of using public-key cryptography for message authentication and encryption throughout the NetBill system because symmetric cryptography offers significant performance advantages. At the same time, the public key cryptography is used to alleviate problems with traditional symmetric-key Kerberos.

Kerberos uses a two-level ticket scheme; to authenticate oneself to a Kerberos service, one must obtain a service ticket, which establishes a shared symmetric session key between the client and server, and establishes that the Kerberos ticket granting server believes the client's identity. To obtain a service ticket, a client must first obtain a ticket-granting ticket (TGT), which proves the client's identity to the Ticket Granting Server. A client obtains a TGT via request from a *key distribution center* (KDC).

The Kerberos KDC/TGT arrangement introduces two significant problems that we may alleviate using public-key cryptography. First, because it maintains a shared symmetric cipher key with every principal in the system, it is an attractive target for attack; recovering from compromise of the KDC requires establishing new shared keys with all users of the system. Second, a KDC and TGT will be a communications or processing bottleneck if a large number of users present a heavy traffic load.

To eliminate the ticket granting server, we replace the TGT with a public key certificate, allowing each service to act as its own ticket granting server. That is, a user presents a service ticket request encrypted with a certified public key, called a *public key-based TGT* (PKTGT), and receives in response a symmetric-cipher-based service ticket. This service ticket is identical in form to a Kerberos service ticket. The key distribution center is replaced by a key repository.

This model can preserve the efficiency of symmetric ciphers for most communication and repeated authentication, and isolates the computational expense of public key cryptography to initial authentication between parties. This model is referred as public-key Kerberos, or PK Kerberos.

In the NetBill system, a customer obtains Kerberos tickets for the NetBill transaction server at the beginning of a session and obtains Kerberos tickets for merchants as he needs them. Merchant servers will continually maintain their own tickets for the NetBill transaction server.

Key Repository

Private keys are large, so users cannot be expected to remember them. Permanently storing private keys at a user's workstation poses security risks and restricts the user's electronic-commerce activities to a single workstation. NetBill uses a key repository to optionally store customers' private keys. These keys are encrypted by a symmetric key derived from a password known only to the customer.

- **Key validation and revocation certificates**

  A public-key-certificate scheme is used to bind User IDs to keys, with NetBill as the certifying authority. NetBill generates a certificate when a customer first proves his identity and begins using NetBill. However, allowing merchants, as services, to grant their own ticket based on these certificates poses a problem: NetBill is no longer involved in ticket-granting, and cannot prevent a ticket from being issued to a user with a compromised key. NetBill needs to invalidate compromised keys as quickly as possible. NetBill maintains a *certificate revocation list* (CRL) at its server. When a key is compromised, the owner creates a revocation certificate and places it in the key repository along with their key. Any party can check that a given key has not been compromised by examining the revocation list. Initially, it would seem that it is necessary for the customer and merchant to contact the server to check CRLs on each transaction. However, it is possible to eliminate this check by allowing the NetBill transaction server to do it when it processes the payment transaction. By delaying the CRL check to late in the protocol, we introduce some minor risks. Customers and merchants may disclose information, such as their preference for particular items or special prices to bogus peers, but there is no financial risk.

- **Pseudonyms**

  Some customers want to disguise their identities. NetBill provides two pseudonym methods to protect the privacy of the customer's identity: a per-transaction method that uses a unique pseudonym for each transaction, and a per-merchant method that uses a unique pseudonym for each customer merchant pair. The per-merchant pseudonym is useful for customers who wish to maintain a consistent pseudonymous identity to qualify for frequent-buyer discounts.

These pseudonym schemes are implemented by introducing a pseudonym-granting server to create pseudonymous for the customer.

9.3.4 Credentials and Authorizations

A restricted proxy is a ticket giving the bearer authority to perform certain operations named in the ticket. NetBill uses a similar construct to implement credentials to prove group membership (to allow merchants to provide discounts to special groups) and to implement access control mechanisms.

Credentials for Group Membership

An organization can provide a credential server that issues credential proxies proving membership in a group. In this case, the credential server is asserting a fact (membership in a group) about which it is authoritative. For example, an auto club may provide a credential server that issues credentials to the members of the club; merchants who offer discounts to the club's members will accept these credentials as proof of membership.

A credential issued to a customer may be unrestricted, or it may optionally be restricted for use on a specific account (for example, in order to prevent corporate employees from taking advantage of corporate discounts for personal purchases).

This is accomplished by passing the account number to the group server as part of the request. If the account number is appropriate for this group, the credential will be issued. The credential contains a cryptographic checksum of the account number and an *account verification nonce*, which is also returned to the customer along with the credential.

This nonce is a pseudorandom number ensuring that merchants can neither determine which different customers (or the same customer in repeated sessions) are using the same account nor easily verify guesses of the customer's account number. The nonce is passed along to the NetBill server in the encrypted part of the EPO so that the NetBill server can verify that checksum passed to the merchant (for his comparison to the credential) corresponds to the account number actually being used.

Credentials can also be used by cooperating merchants to restrict information access. In this way, merchants only sell to approved customers, i.e., those who can present a certain credential. This offers a solution for merchants who, for example, can restrict distribution of sensitive documents only to individuals whose credentials verify a need-to-know.

Access Control Mechanism

Access control can be implemented by using proxies, an account owner (such as a parent) may have a restriction on the account such that no purchases can be completed by a given customer (such as a child) without approval from an access-control server. This allows a different organization to provide access-control services. For example, both the PTA and a church group could offer competing access control services.

The NetBill protocols are robust against failures, and retain essential information to protect customers and merchants against fraud.

9.4 NetCheque System

The NetCheque system, under development at the Information Sciences Institute of the University of Southern California, is a distributed accounting service supporting the credit-debit model of payment. Users of NetCheque maintain accounts on accounting servers of their choice. A NetCheque account works in much the same way as a conventional checking account: account holders write electronic documents that include the name of the payer, the name of the financial institution, the payer's account identifier, the name of the payee, and the amount of the check. Like a paper check, a NetCheque bears an electronic signature, and must be endorsed by the payee using another electronic signature before the cheque will be paid.

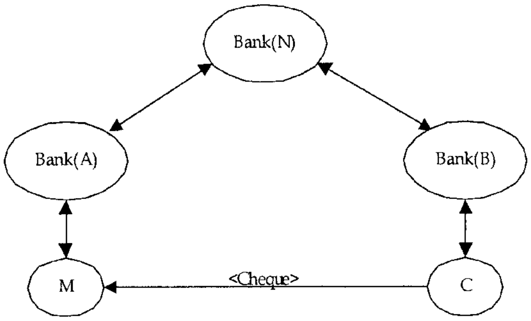


Fig. 9.3 Hierarchy of NetCheque servers

As a distributed accounting service, properly signed and endorsed cheques are exchanged between accounting servers to settle accounts through a hierarchy, as shown in Fig. 9.3. In addition to improving scalability and acceptability, clearing between servers allows organizations to set up accounts in their own in-house-accounting servers with accounts corresponding to budget lines. Authorized signers write cheques against these accounts, while the organization maintains a single account with an outside bank, integrating its own internal accounting system with the external financial system.

The NetCheque accounting system was designed originally to maintain quotas for distributed system resources, resulting in frequent transactions for small amounts. Thus, it is well suited to support small payments needed for some kinds of electronic commerce. This requirement for handling micropayments requires high performance, which is obtained through the use of conventional, instead of public-key, cryptography. This gives up some support for independent verification of payment documents at each stage in the payment pipeline.

9.4.1 Implementation Overview

The system is based on the Kerberos system [9.6]. The electronic signature used when writing or endorsing a cheque is a special kind of Kerberos ticket called a proxy. The cheque itself contains information about 1) the amount of the cheque, 2) the currency unit, 3) an expiration date, 4) the account against which the cheque was drawn, and 5) the payee or payees, all readable by the bearer of the cheque, together with 6) the signatures and endorsements accumulated during processing, verifiable by the accounting server against which the cheque was drawn. For performance, the Kerberos proxy used as a signature is based on conventional cryptography, but it may be replaced by a signature using public-key cryptography with a corresponding loss of performance.

To write a cheque, the user calls the write-cheque function, specifying an account against which the cheque is to be drawn, the payee, the amount, and the currency unit. Defaults for the account and currency unit are read from the user's chequebook file. The write-cheque function generates the cleartext portion of the cheque, obtains a Kerberos ticket that will be used to authenticate the user to the accounting server, generates an authenticator with an embedded checksum over the information from the cheque, and places the ticket and authenticator in the signature field of the cheque. The cheque is then base-64 encoded and may be sent to the payee through electronic mail, or transferred in real time as payment for services provided through an online service.

The deposit-cheque function reads the cleartext part of the cheque, obtains a Kerberos ticket to be used with the payer's accounting server, generates an authenticator endorsing the cheque in the name of the payee for deposit only into the payee's account, and appends the endorsement to the cheque. An encrypted connection is opened to the payee's accounting server and the endorsed cheque is deposited. If the payee and the payer both use the same accounting server, the response will indicate whether the cheque cleared.

If different accounting servers are used, the payee's accounting server places a hold on the funds in the payee's account and indicates to the payee that the cheque was accepted for collection. The payee has the option of requesting that the cheque be cleared in real time, though we expect there may be a charge for this service. If a cheque accepted for collection is rejected, the cheque is returned to the depositor, who can take action at that time. As a cheque is cleared through multiple accounting servers, each server attaches its own endorsement, similar to the endorsement attached by the payee.

In some cases the payee's and payer's accounting servers can settle the check directly, bypassing higher levels of the hierarchy. This is possible when the cheque is drawn on an accounting server that is trusted to properly settle accounts. Such trust might be based on certificates of insurance representing endorsement of the accounting server. In such cases, the hierarchy would still be used to settle any imbalance between credits and debits for each accounting server at the end of the day, but the cost of these transfers would be amortized over the days transactions. To determine account balances and fill out about cleared cheques, authorized users can call the statement function which opens an encrypted connection to the accounting server and retrieves the account balance for each currency unit, together with a list of cheques that have been recently deposited to, or drawn on and cleared through the account. The entire cheque is returned, allowing the user's application to extract whatever information is needed for display to the user, or for integration with other applications.

9.5 Summary

This chapter is divided into two parts, the first part describes the ways in which existing banking organizations can introduce a check-based payment system in a phased manner. The second one introduces two electronic check systems: NetBill and NetCheque.

In the NetBill system, some methods have been introduced for certified delivery, access control, user certificates, pseudonyms, and their integration. The design principle is to provide very high degrees of security and flexibility while still providing good efficiency.

The NetCheque system is a distributed payment system based on the credit-debit model. The strengths of the NetCheque system are its security, reliability, scalability, and efficiency. Signatures on cheques are authenticated using Kerberos. Reliability and scalability are provided by using multiple accounting servers. NetCheque is well suited for clearing micropayments; its use of conventional cryptography makes it more efficient than those systems based on public-key cryptography. Though NetCheque does not itself provide anonymity, it may be used to facilitate the flow of funds between other services that do provide anonymity.

9.6 References

[9.1] B. Cox, J. D. Tygar, M. Sirbu (1995) NetBill security and transaction protocol. Technical Report, Carnegie Mellon University. http://www.ini.cmu.edu/netbill

[9.2] M. Sirbu, J. D. Tygar (1995): NetBill: an electronic commerce system optimized for network delivered information and services. In: Proc. IEEE Compcon'95. IEEE Press, New York. http://ini.cmu.edu/netbill

[9.3] C. Neuman, G. Medvinsky (1995) Requirements for network payment: the NetCheque perspective. In: Proc. IEEE Compcon'95. IEEE Press, New York. http://nii.isi.edu/info/netcheque/documentation.html

[9.4] University of Southern California Chronicle (1994) The check is in the e-mail. ftp://prospero.isi.edu/pub/netcheque/information/usc-chronicle-941107/netcheque-usc-chronicle-941107.html

[9.5] D. O'Mahony, M. Peirce, H. Tewari (1997) Electronic payment systems. Artech House, Boston London.

[9.6] C. Neuman, T. Ts'o (1994) Kerberos: an authentication service for computer networks. IEEE Communications 32(9).

10 Secure Electronic Transactions: Overview, Capabilities, and Current Status

Gordon Agnew

A&F Consulting, and
University of Waterloo, Ontario, Canada

10.1 Introduction

Until recently, there were two primary forms of credit card transactions:

1) Card present and,
2) Card not present or mail order telephone (MOT).

In a typical "in store" transaction, the customer presents their credit card to perform a transaction. The merchant "swipes" the card and the customer's credit card information along with the amount of the transaction is forwarded to a payment gateway. Once the credit information is verified, the payment gateway returns an authorization to the merchant and a receipt is issued to the customer. In the event of fraud on the part of a customer, the merchant is indemnified against loss since the payment gateway authorized the transaction.

In the case of a purchase made via the telephone, the customer's credit card is not physically present for verification. Generally, the merchant simply accepts the customer's card number over the phone and completes the transaction. Since no authorization was issued by the payment gateway, liability for customer fraud rests with the merchant. For some merchants, this risk is acceptable if profit margins were large enough. On the other hand many merchants have found the risks unacceptable.

In 1996, Mastercard and Visa announced their support of a developing standard for electronic credit card transactions. This replaced the competing standards that each company was pursuing independently. In 1997Visa and Mastercard pooled their resources and formed Secure Electronic Transaction LLC (SETCo) to implement the *SET Specification* [10.4].

SETCo manages the Specification, oversees SET product-compliance testing, and promotes the use of SET as a global payment standard.

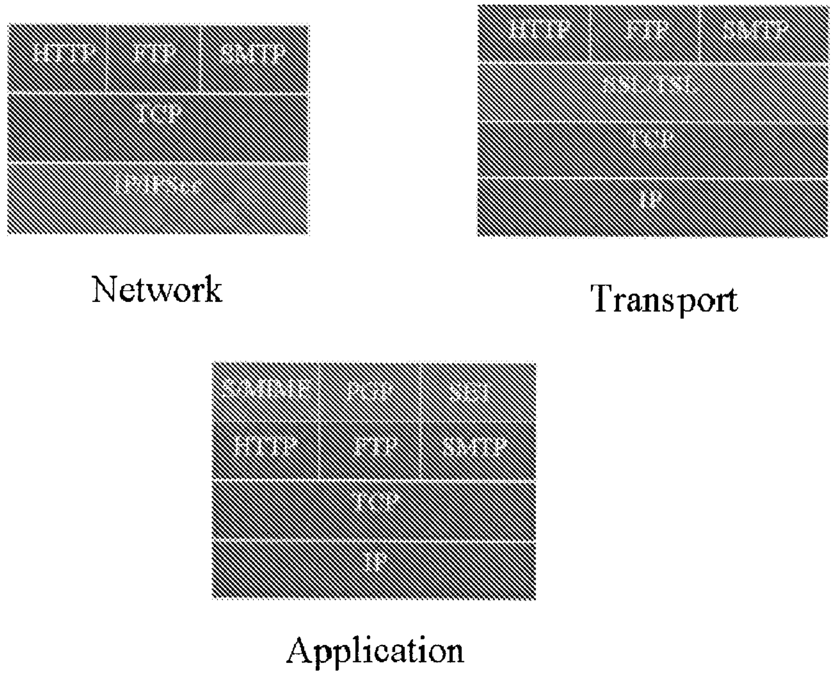
The secure electronic transaction (SET) protocol, in many ways, mirrors a card-present transaction over the Internet.

In the following sections, we will examine the details of the operation of SET as well as compare its capabilities to other protocols used in electronic commerce. In addition, we will look issues related to SET's adoption and refinements to the protocol.

10.2 Protocol Stack and Capabilities

There are many functions required to implement a large interconnected network such as the Internet. To facilitate this, network functionality is usually divided into a set of layers or the *protocol stack.* Each layer "talks" to a corresponding or "peer" layer at the other end of the communications channel. Each layer works transparently with the other layers in the network. The lowest layer in the network is the *physical* layer that involves the actual means for transporting data, for example, the actual cables or fibre optics that form the network. At the highest level is the *application* layer which are the programs that are run by the user. One major advantage of such a structure is that the user does not have to be concerned with how the lower layers are implemented. The user simply runs the application and information is passed locally down through the various layers to the Physical layer. The user's data is passed to the physical layer at the destination server then back up to the corresponding application layer at the other end of the connection. In most cases, this layer transparency is realized by a process called "encapsulation;" as data is passed from a higher layer to a lower, the lower layer takes the original data and adds header and control information then passes it down to the next layer. At the destination, the process is reversed and the header/control information is stripped off as data is passed to higher layers. This process simplifies implementation of networks but introduces some interesting security issues, as we will explain below.

There are several levels at which security can be introduced to protect Internet connections. The level in which security functions are used has a strong impact on what types of security can be provided. In Fig. 10.1, three security protocols are shown as they fit into the Internet protocol stack. The three we will consider are IPSec, SSL and SET.



Network

Transport

Application

Fig. 10.1 Security layers

10.2.1 Internet Protocol Security (IPSec)

As we see in Fig. 10.1, Internet protocol security (IPSec) is implemented in a relatively low layer. IPSec provides the facilities to encrypt and authenticate user's data (*payload*). If this done, an attacker can see the where the information is going (at least what IP addresses are involved) but not the information itself. In addition, IPSec has the option of taking a standard IP message encrypting it and placing it in a new IP message with a new "disguised" header. This is known as tunnel mode and allows users, for example, to set up private groups over the Internet (virtual private networks).

There are several advantages to providing security at this level:

- Security functions are transparent to the user – the user may not even be aware they are being used.
- The identity of participants can be protected as their IP addresses can be masked.

There are also disadvantages:

- The security functions only protect the IP layer and below – one data is passed to higher layers, it is not protected. If several users are on the same system, information for one user may be visible to other users.
- Identities of users can only be resolved to an IP address. It is common that many users may share an IP address thus, authentication of a particular user may not be possible.

10.2.2 Secure Socket Layer (SSL)

Secure Socket Layer (SSL) was developed by Netscape and is currently in version 3. It is the defacto standard for Internet security at this level and is implemented in most browsers and by most web servers. SSL is designed to provide security functions independent of the application. Since it works at a higher layer than IP-Sec, identities can be resolved to the level of an individual. By the same token, SSL by itself cannot prevent an observer from knowing who is communicating since IP addresses will be added at the lower layers.

SSL designates two types of participants: clients and servers. Clients always initiate a communications session with a server. The server is required to provide authentication information to the client (a certified public key) if requested. The client, however, is not required to provide a certified public key to the server. If this is the case, the applications using SSL may require some other means of authenticating the user (such as a user ID and password/PIN). Once the session has been negotiated, SSL provides a secure (encrypted) and authenticated (data-integrity checks) communications channel between the client and server.

10.2.3 SET

As shown in Fig. 10.1, SET provides security functions at the highest (application) layer of the protocol stack. As in our previous discussion, there are advantages and disadvantages to this. SET is an application and security its functions are not available to other applications. The integrity of SET relies on the ability to resolve identities to a particular individual, merchant or payment gateway (through the use of a full public key infrastructure as we will discuss in Section 10.3) as well as the ability to protect the information exchanged.

As with SSL, even though the information is protected, and observer can still glean information about the participants in a transaction.

10.3 SET Overview

In this section, we will examine the structure of SET and its related security functions[1].

There are two major parts to the SET protocol.

- Registration
- Transaction processing

10.3.1 SET Registration

The security and integrity of transactions are heavily reliant on the use of certified public keys or *public key certificates*. To create a certificate, the user presents unique identification information (ID) and their public key to a c*ertificate authority* (CA). Once the CA is satisfied that the user is authentic (for example, the manager of a bank may authenticate a particular customer), the CA binds the ID and public key of the user together (usually by creating a message digest[2]) then forms a *digital signature*[3] on the result. For another participant to verify the public key of a particular user, they require a trusted copy of the CA's public key in order to verify the certificate. It is assumed that a trusted version of at least on CA's public key is available to the participants.

SET recognizes three types of participants in a transaction.

- The customer (cardholder)
- The merchant
- The payment gateway.

SET then defines a hierarchical approach to creating and distributing public-key certificates for each type of participant. This is shown in Fig. 10.2. Here, the highest member of the hierarchy is the *root certificate authority* maintained by SETCo. The root authority issues public key certificates to the various payment brands. These in turn become Certificate Authorities authorized to issue certificates to their member banks.

[1] A full description of SET can be found in SET Specification Books [10.4]

[2] A message digest is a fixed length image of a longer message formed using a transformation that is "one-way" and unpredictable. That is, it is very easy to create but virtually impossible to find a second message that would create the same image. For a more in depth look at cryptographic functions, the reader is referred to [10.2]

[3] A digital signature is formed using the signer's private key. It can be verified using the signer's public key.

Further down the hierarchy are the certificate authorities associated with each type of participant in a transaction. The *payment card issuing certificate authority* issues public key certificates to customers. The *merchant bank* or *acquirer certificate authority* issues public key certificates to the merchants while payment gateways have their own certificate authority.

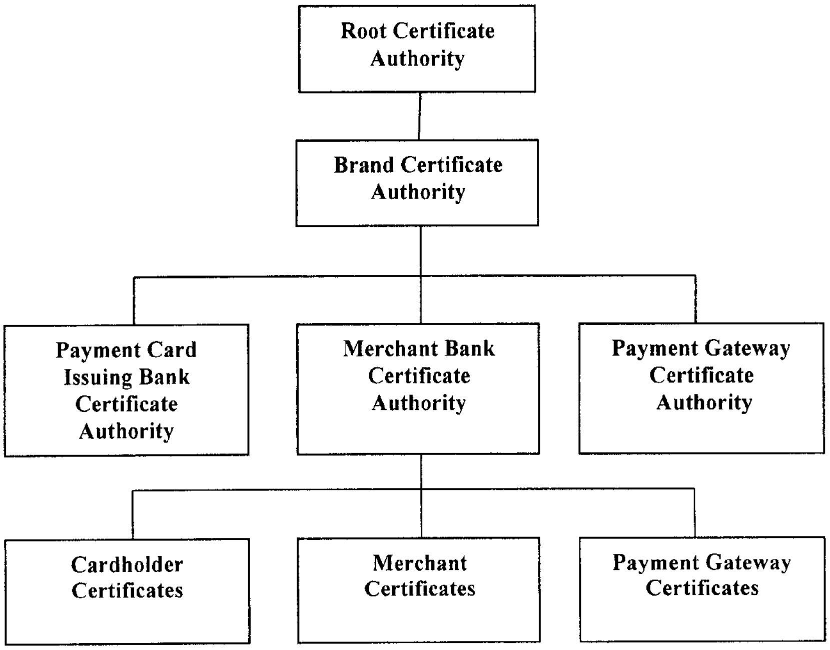

Root Certificate Authority
Brand Certificate Authority
Payment Card Issuing Bank Certificate Authority
Merchant Bank Certificate Authority
Payment Gateway Certificate Authority
Cardholder Certificates
Merchant Certificates
Payment Gateway Certificates

Fig. 10.2 SET certificate hierarchy

In such a hierarchy, a *certificate chain* can be used to verify any member of the hierarchy. For example, for a particular merchant, the certificate chain might include their own public key certificate issued by their acquirer CA, a certificate on the acquirer CA issued by the brand CA and finally the certificate of the brand CA as issued by the root CA. A trusted version of the root CA's public key would allow the chain to be verified. A graphic representation of a certificate chain is shown in Fig. 10.3.

Merchant's Certificate (from Merchant's Bank CA)	**Merchant's Bank CA Certificate (from Brand CA)**	**Brand's CA Certificate (from Root CA)**

Fig. 10.3 Example certificate chain for a merchant

10.3.2 Transaction Processing

There are three main phases in a secure electronic transaction:

- Purchase request
- Payment authorization
- Payment capture

An overview of the interaction among the participants in a transaction is shown in Fig. 10.4.

Purchase Request Phase

The details of the purchase request are shown in Fig. 10.5. Within the purchase-request phase, there are 5 basic steps, as we will describe.

Initiate Request

The process starts with the customer shopping, and selecting an item or items. The customer has a completed order form and has selected a particular payment card. The customer's (cardholder's) computer running the cardholder's software package (hereafter called just the *cardholder*) sends an *initiate request* (*P INIT REQ*) message to the merchant requesting the certified public key of the payment gateway.

Initiate Response

Once the merchant receives the initiate request, it assigns an unique transaction ID to the message and returns a signed version of the transaction ID, its own certificate and the appropriate (for the particular brand) payment gateway's certificate to the cardholder.

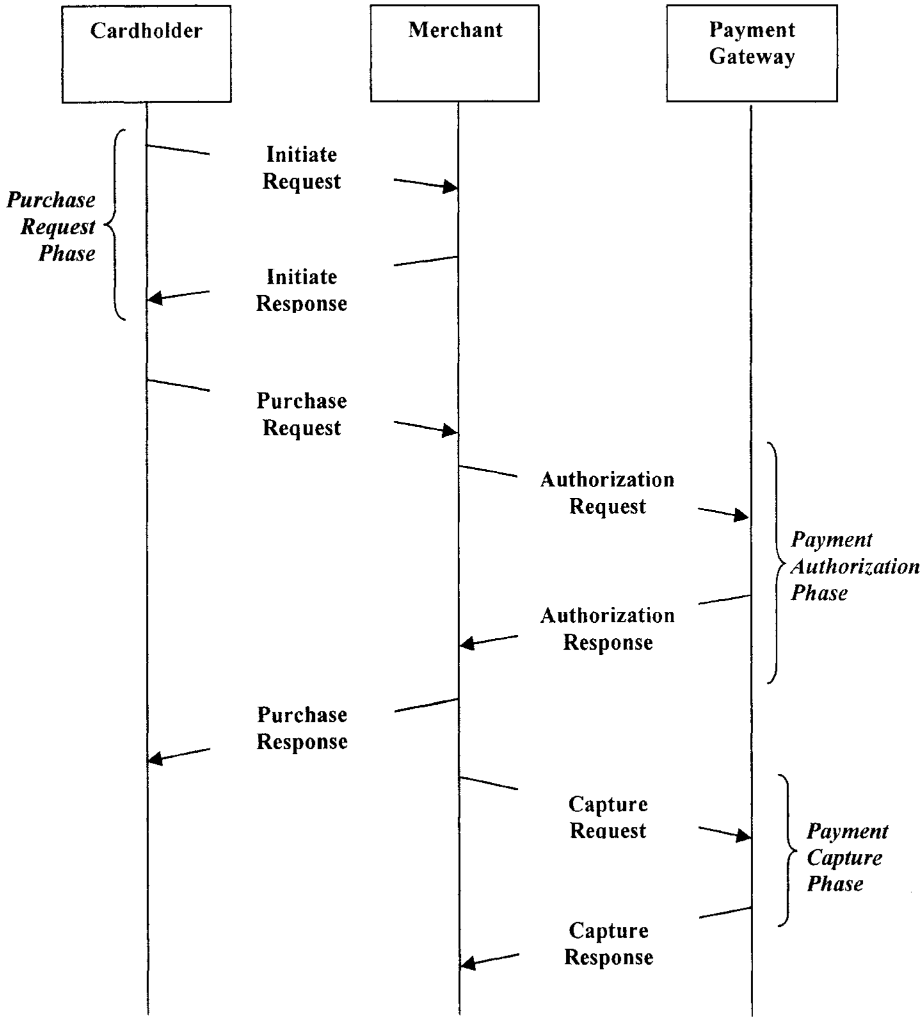


Fig. 10.4 SET overview

Cardholder Purchase Request

Once the response is received, the cardholder verifies the certificates of the merchant and gateway as well as the merchant's digital signature on the transaction information. Once this is complete, the cardholder creates two messages: an *order information* (OI) message intended for the merchant and a *payment information* (PI) message intended for the payment gateway. The PI message information such as the credit card number of the cardholder and will be concealed from the merchant. These messages both contain the unique transaction ID that the merchant assigned. This is done so that the two messages can be linked to one another.

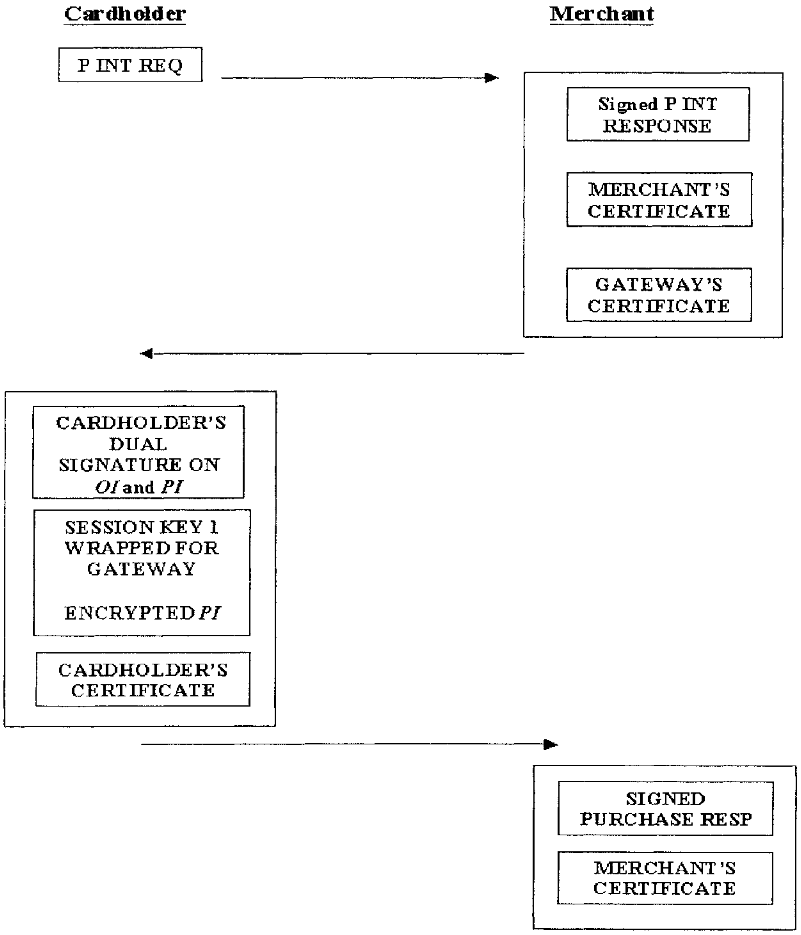


Fig. 10.5 Purchase request phase

At this point, a very elegant method is used bind the two messages together. The cardholder forms message digests of both the OI and PI. These digests are concatenated, then a third message digest is formed. This final digest is then digitally signed by the cardholder. This forms the *dual signature* on OI and PI.

The next step is used to hide the PI information from the merchant. The cardholder generates a random session key (to be used with a conventional encryption algorithm) that is used to encrypt the PI. To transport this information to the payment gateway, the cardholder combines the random session key and their account information into a message then encrypts it using the payment gateway's public key (so that only the PG can recover the account information and the session key that can decrypt the PI).

Merchant then is forwarded a message containing the PI and OI digest, the dual signature, the "wrapped" version of the PI, session key and account information and the cardholder's certificate.

The reason for the dual signature scheme is as follows: the payment gateway will only have a digest of the order information and not the order itself. The payment gateway cannot determine the purchase from that information. If a dispute arises, between the merchant and customer, the OI can be produced and the payment gateway with knowledge of the PI can regenerate the message digests and verify whose claim is correct. This is an important element in security of SET.

Merchant's Purchase Request Processing

When the purchase request is received at the merchant, it verifies the cardholder's certificate. This is then used to verify the dual signature on the OI and digest of the PI to ensure no tampering of the OI has occurred.

Once this has been verified, the merchant generates a digitally signed *purchase response* message that is returned to the cardholder.

Purchase Response

In the final step in this phase, the cardholder uses the merchant's certified public key to verify the purchase response. This is stored for future reference.

Payment Authorization Phase

This part of the protocol involves the merchant and the payment gateway. The objective is for the merchant to acquire authorization for the transaction. There are three basic steps, as shown in Fig. 10.6.

Merchant Authorization Request

The merchant starts by creating a digitally signed authorization request that includes the amount to be authorized, the transaction ID, and other details about the transaction.

The merchant generates a random session key that is used to encrypt this message. The session key is then wrapped using the payment gateway's public key.

This information is sent along with the cardholder's PI information and wrapped session key, cardholder's certificate and merchant's certificate.

Payment Gateway Processing

When the gateway receives the authorization request, it uses its private key to recover the wrapped session key. This is then used to decrypt the request. The merchant's certificate is verified then used to verify the signature on the request.

Next, the second session key and customer account information are recovered. The session key is then used to recover the PI. The cardholder's certificate is verified and the digital signature on the OI and PI is verified. As a further check, the Transaction ID's on both parts of the message are compared to ensure that they are the same.

The next operation involves the payment gateway creating a message for the issuing bank. This is done over the private financial network.

If the purchase is authorized, then a digitally signed response message is generated by the payment gateway. This message is encrypted with a new random session key that is wrapped using the merchant's public key, then forwarded to the merchant.

Merchant Response Processing

When the response is received by the merchant, the payment authorization is recovered and the signature is verified. A copy of this authorization is kept by the merchant.

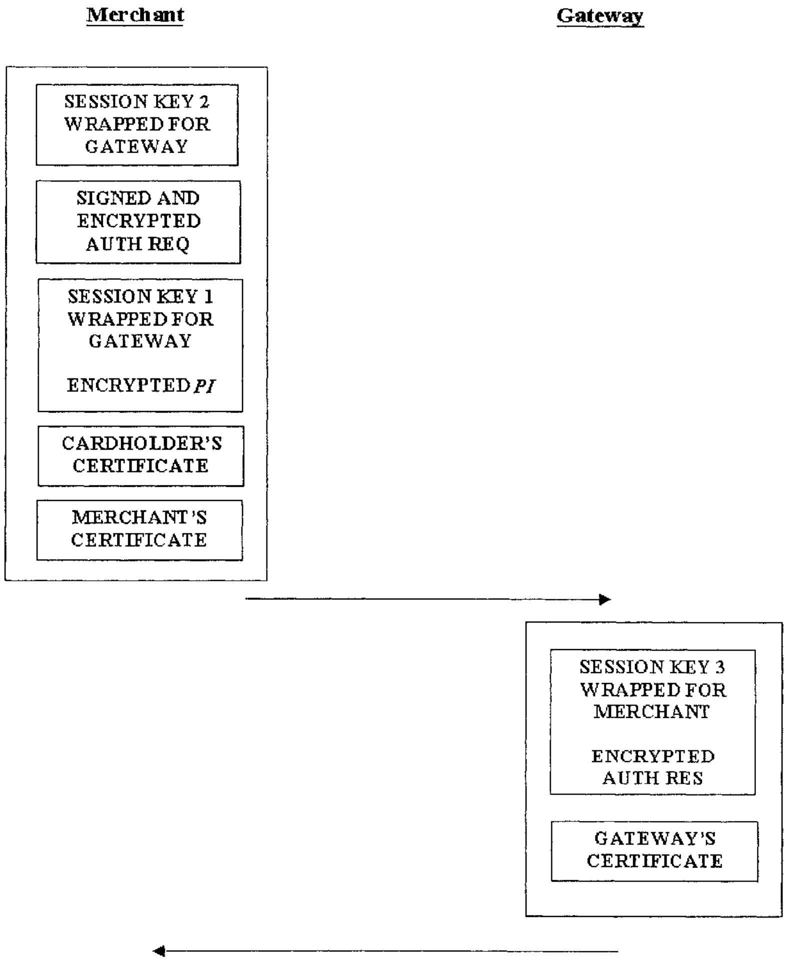

Merchant
Gateway
SESSION KEY 2 WRAPPED FOR GATEWAY
SIGNED AND ENCRYPTED AUTH REQ
SESSION KEY 1 WRAPPED FOR GATEWAY
ENCRYPTED PI
CARDHOLDER'S CERTIFICATE
MERCHANT'S CERTIFICATE
SESSION KEY 3 WRAPPED FOR MERCHANT
ENCRYPTED AUTH RES
GATEWAY'S CERTIFICATE

Fig. 10.6 Payment authorization phase

Payment Capture Phase

The final phase in the SET protocol is payment capture. In this phase, the Merchant requests payment from the payment gateway. This phase may occur sometime after the transaction has occurred and involves three basic steps, as shown in Fig. 10.7.

Merchant Payment Capture Request

The merchant creates a digitally signed payment request that includes the final transaction amount, the transaction ID, and other transaction information. This is encrypted using a new random session key that is wrapped using the payment gateway's public key. The encrypted message is sent to the payment gateway along with the merchant's certificate.

Payment Gateway Capture Processing

Upon receipt, the payment gateway recovers the session key, capture request then verifies the merchant's certificate and signature on the request

The payment gateway generates a digitally signed and encrypted response message that is forwarded to the merchant along with the gateway's certificate.

Merchant Processing of Response

This is the final step in the protocol. The merchant recovers the session key and the capture message and verifies the gateway's certificate as well as the digital signature on the message. This is stored by the gateway for reconciliation for payment from the issuer.

10.4 SET Performance

From the description of the SET protocol, it is apparent that SET provides a high level of security and privacy for the participants. This is mainly due to the extensive use of public key certificates and digitally signed and verified messages. This has several important implications. Trust in the system relies on the deployment of a full public key infrastructure. If SET is to be used on a wide-scale basis, certificates have to be issued to all users. This is an enormous and expensive task. On the other hand if the PKI is not in place, then SET will not be used by a large number of users.

In version 1.0 of SET, RSA is specified to implement the public key operations. At present a minimum of 768-bit RSA is required for security, preferably 1024-bit. Public key operations (signing/verifying, wrapping/unwrapping) are computationally intensive, and certificates are large in size and require significant bandwidth to transmit.

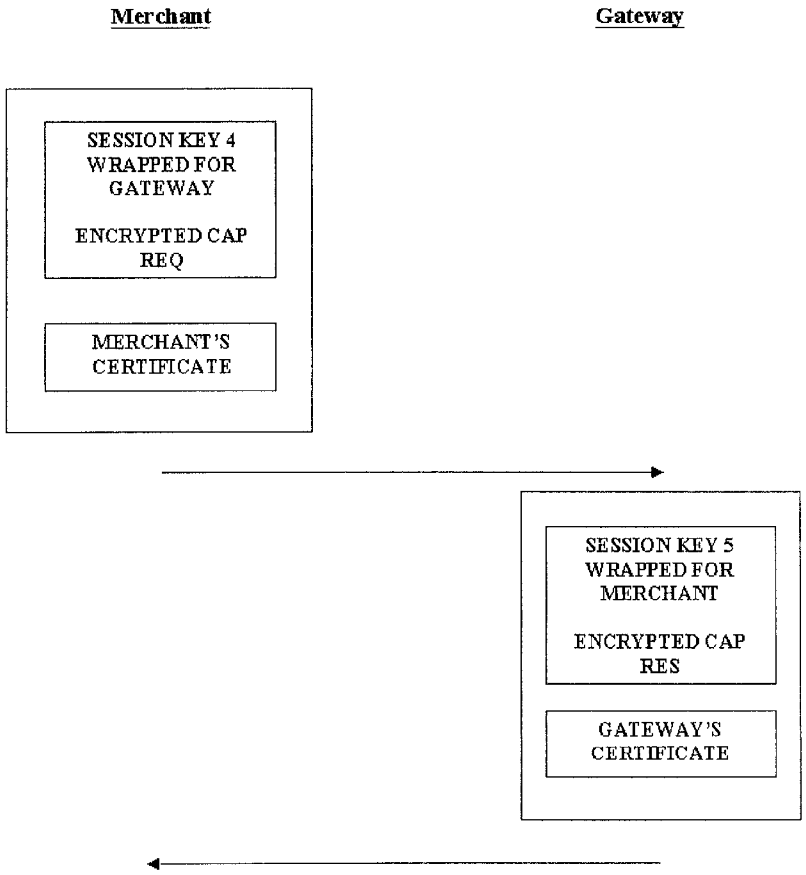


Fig. 10.7 Payment capture phase

In the case of the cardholder using a typical desktop computer, the computational load is not significant. If, on the other hand, the cardholder is not bound to a particular machine, then the cryptographic functions may be implemented in a portable token, such as a smart card. Implementing RSA on smart cards usually requires the smart card to have a cryptographic co-processor that raises the cost of the card.

There is also the issue of conducting e-commerce transactions using wireless handheld devices, such as cell phones or PDAs. In these situations bandwidth and processing power are at a premium and supporting SET may be difficult.

The GartnerConsulting Group did an extensive evaluation of the performance of SET [10.1]. In the study, it was anticipated that merchants could expect in the order of 10,000 transactions per day while a large payment gateway may approach ½ million transactions per day. In this case, software implementations of the public-key system may not be able to perform operations quickly enough; hardware accelerators may be required (adding to the cost of the infrastructure). They also examined the advantages of using other public key cryptographic systems. In their report, *elliptic curve cryptosystems*[4] (ECC) were considered and shown to have significant advantages in terms of bandwidth and processing overhead.

Sans and Agnew [10.3] present the results of an extensive study of the communications and processing overhead for SET. They show some alternative methods for processing transactions that reduce the overhead incurred using SET.

10.5 What Lies Ahead

There are a number of companies currently offering support for SET. These include IBM, Verisign, CyberTrust, Verifone, Sterling Commerce, Terisa, Netpay and GlobeSet.

SETCo lists more than 40 countries that have adopted SET in one form or another [10.4].

A proposal for SET 2.0 incorporates alternative asymmetric key cryptographic systems (specifically, elliptic curves) and SET 2.0 will also support the use of debit cards by allowing personal identification numbers (PINs) to be encrypted and included in the payment message [10.5]. In addition, a smart-card-based version known as chip-secured SET (C-SET) is being developed to allow smart cards to perform cardholder authentication and transaction security functions (encryption and signatures).

10.6 Summary

In this chapter, we have presented a detailed outline of the SET protocol. The capabilities and shortcomings of SET have been compared to other Internet security protocols.

Currently, SSL is the most widely deployed and used security protocol. It is relatively fast and provides transparent security to the user. It does not, however

[4] The reader is referred to www.certicom.com for a more complete review of ECC technology.

provide the mutual authentication and digital signature capabilities that are required for truly secure e-commerce.

SET, on the other hand, is a very robust protocol that provides a high level of security and trust. The major impediments to widespread deployment and use of SET are the current lack of a comprehensive public key infrastructure and the large overhead required to run the SET protocol. Improvements in processing power and the use of alternative public key cryptosystems such as elliptic-curve-based systems (ECC) may help to overcome some of these obstacles.

10.7 References

[10.1] GartnerConsulting: SET comparative performance analysis. www.setco.org/download/setco6.pdf

[10.2] A. J. Menezes, P. Oorschot, S. Vanstone (1997) Handbook of applied cryptography. CRC Press, New York.

[10.3] O. Sans, G. Agnew (2001) An efficient multiple merchant payment protocol for secure electronic transactions based on purchase consolidation. In: W. Kou, et al. (eds.) Electronic commerce technologies – ISEC2001, LNCS 2048. Springer, Berlin Heidelberg New York.

[10.4] www.setco.org/set_specifications.html

[10.5] www.setco.org/extensions.html

11 Credit Card-Based Secure Online Payment*

Johnny W. Wong[1], Lev Mirlas[2], Weidong Kou[3], and Xiaodong Lin[1]

[1] University of Waterloo
Waterloo, Ontario, Canada

[2] IBM Canada Ltd.
Warden Ave., Markham, Ontario, Canada

[3] University of Hong Kong
Pokfulam Road, Hong Kong

11.1 Introduction

The credit card is a popular payment method for the purchase of goods and services. Traditionally, credit cards are used by buyers to purchase merchandise from brick-and-mortar stores. Transactions are carried out face-to-face. Typically, the merchant first obtains authorization from the credit card company regarding the transaction. If the transaction is authorized, the buyer is asked to sign for the purchase, and a paper receipt stating the terms of the sale will be issued to the buyer. The merchant also verifies that the buyer's signature matches the cardholder's signature at the back of the card, and that the card has not expired.

Shopping by phone, by mail, or by fax are convenient alternatives to shopping at a brick-and-mortar store. The buyer sends the order information to the merchant by phone, mail, or fax, together with the credit card information such as the credit card number, cardholder ID, and expiry date. The order information contains the goods or services to be provided by the merchant and the agreed price. Upon receiving the order, the merchant first obtains authorization of the transaction from the credit card company. The merchant then ships the merchandise and charges the buyer through his/her credit card. For purchase by mail or by fax, the merchant has the buyer's signature on the order form. The buyer's signature is not available to the merchant when the order is by phone. To obtain a record of the order, the

* This work was supported by the IBM Centre for Advanced Studies and an IBM Faculty Partnership Award.

merchant records the phone conversation with the buyer when the order was placed.

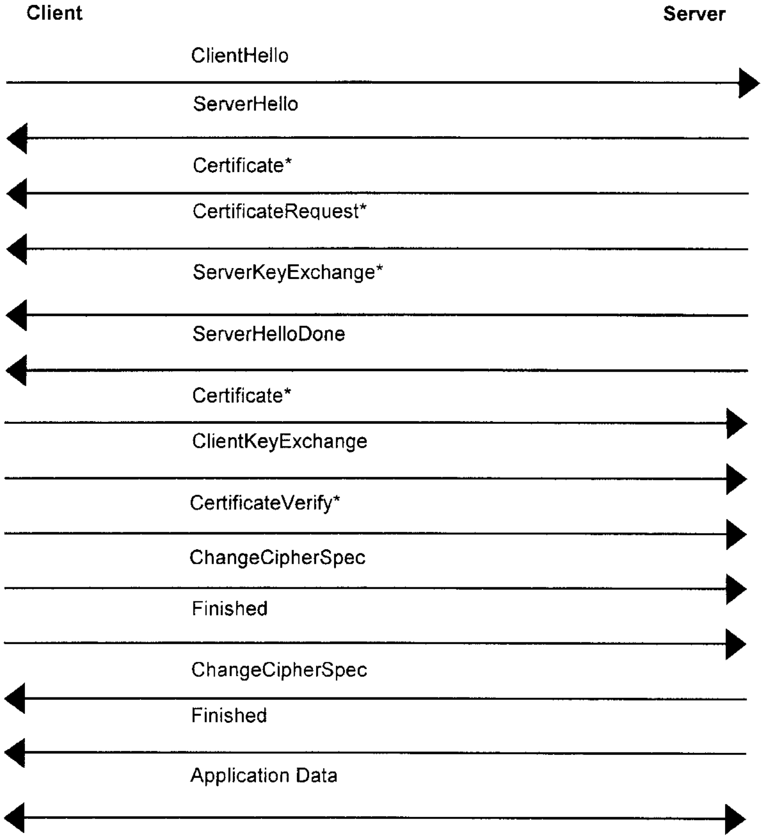
Shopping over the Internet is another alternative to shopping at a brick-and-mortar store. In this case, the order information and credit card information are transmitted over the Internet, which may not have the same level of security as phone, mail, or fax. Methods to ensure secure online payment by credit card are therefore important to the success of shopping over the Internet. This chapter is concerned with online payment methods based on the existing credit card payment infrastructure. We first provide an overview of online payment by credit card, and then discuss the trust issue related to this payment method. To overcome the trust problem, we introduce a new payment protocol using a trusted third party. This protocol can be viewed as a special case of online payment by credit card, which addresses the trust problem that the current credit-card-based online payment systems have.

11.2 Online Payment by Credit Card

When making a purchase over the Internet using a credit card, procedures for secure communication are needed to authenticate the parties involved, to ensure confidential transmission of order and payment information, and to protect the integrity of the transaction. The current approach to achieving secure communication is to use the secure sockets layer (SSL) [11.1].

As briefly discussed in Chap. 10, SSL is a protocol designed to provide secure communication. It performs server authentication, and, optionally, client authentication. With SSL, private information is protected through encryption, and a user is assured through server authentication that they are communicating with the desired website and not with some bogus website. In addition, SSL provides data integrity, i.e., protection against any attempt to modify the data transferred during a communication session. The main exchanges in an SSL session [11.5-11.6] are shown in Fig. 11.1, and the detailed descriptions for each exchange are provided in Sect. 11.6.1, Appendix A. The use of SSL has led to an improvement in the buyer's confidence when making payments by credit card over the Internet.

A credit card transaction involves five main parties: buyer, merchant, merchant bank, issuer, and acquirer. The merchant has a contract with the merchant bank to enable them to accept credit card payments over the Internet. The issuer is a financial institution such as a bank that issues a credit card to the buyer. It is responsible for the cardholder's debt payment. The acquirer, on the other hand, obtains credit card transactions from the merchant and processes them for payment. The acquirer provides authorization to the merchant that a given account is active and that the proposed purchase does not exceed the cardholder's credit limit. The acquirer also makes payments to the merchant's account, and is then reimbursed by the issuer. The merchant bank may function as the acquirer.



* Indicates optional or situation-dependent messages that are not always sent.

Fig. 11.1 The main SSL exchanges

We now describe the steps involved in handling payments by credit card over the Internet. The transaction starts with the buyer deciding to place an order online from a web page at the merchant's website. The merchant's commerce application prompts the buyer for payment information (i.e., credit card number, cardholder ID, and expiry date) along with other information such as shipping address. The buyer then enters payment information into a form secured using SSL. With the secured form, the payment information is protected as it is sent to the merchant.

Upon receiving the order, the merchant server sends the payment information to the acquirer processor for authorization, using dedicated and secure lines. The authorization is a request to hold funds for the purchase. The acquirer will verify that the credit card has sufficient funds to cover the amount of the transaction. The acquirer either authorizes a certain amount of money or declines the transaction. Each authorization reduces the available credit only, it does not put a charge on the cardholder's bill or transfer funds to the merchant.

After the transaction has been authorized, the merchant charges the authorized amount to the buyer's credit card. This is known as "capture." According to bank card association rules, the merchant is not allowed to capture a transaction until the ordered goods can be shipped, so there may be a time lag between authorization and capture. If the buyer cancels the order before capture, a "void" is generated.

Captures are accumulated into a batch and settled automatically at regular time intervals, e.g., at the end of each day. This settlement can be viewed as a transaction between the acquirer and the merchant. When a batch is submitted, the merchant's payment-enabled web server connects with the acquirer to finalize the transactions and transfer the corresponding amounts to the merchant's bank account.

At the issuer, a monthly statement is sent to the cardholder for the purchases made since the last statement.

11.3 Trust Problems in Credit Card Payments

The issue of trust in the payment process is one of the most critical aspects that determines the acceptability of a payment method. Clearly, if the parties participating in a payment transaction cannot be assured of the correctness of the transaction, then they are unlikely to conduct the transaction in the first place.

In a payment transaction, trust is engendered through the following assurances:

The buyers must be assured that:

- The transaction will result in them paying exactly the specified amount and being billed only for the item they bought at the agreed price.

- The payment information they provide will not be stolen or abused for extracting unauthorized payments from them.
- They will receive the goods that they have purchased.
- In the case of a dispute they have in their possession enough evidence to prove whether or not the payment took place.

The merchants must be assured that:

- The transaction will result in them being paid exactly the agreed amount.
- In the case of a dispute they have in their possession enough evidence to prove whether or not the payment took place.

A "good" payment protocol assures these trust points for both the buyer and the merchant. These trust points are easily met in credit card transactions at a brick-and-mortar store. Since the transaction is face-to-face, the merchant can verify the buyer's signature against that on the back of the credit card. The amount of the transaction and goods purchased can readily be confirmed. There is no danger in payment information being stolen if the buyer is dealing with a trustworthy merchant. In the case of a dispute, each party has a signed copy of the receipt which can be used as evidence that the transaction took place.

The issue of trust becomes more complicated when credit-card payments are made over the Internet. The use of SSL may not address all the trust points mentioned above. A common concern is that some of the traditional assurances of integrity of the payment transactions can become compromised. Consider the assurance mentioned above in the context of online transactions.

The issues at the buyer side are:

- The buyers would like to be assured that they are paying exactly the specified amount. In general, there is a lack of trust in online systems. As a result, there is no guarantee that the amount charged is the same as the amount requested.
- The buyers would like to be assured that the payment information provided cannot be stolen or abused for extracting unauthorized payments from them. The online security of trusted data has been compromised many times, and such events have been highlighted in the media. It seems that the task of keeping trusted data, such as credit card numbers, safe from intruders in an online site is not trivial. However, merchants who collect this data are typically not technically savvy, and do not have the resources to institute the kind of secure and trusted computing base necessary to constantly guard against possible intrusion and data theft attempts.

- The buyers would like to trust that they will receive the goods that they have purchased. This depends on the track record and reputation of the merchant.
- In the case of a dispute the buyers would like to have in their possession enough evidence to prove whether or not the payment took place. In the online world, the buyers generally do not get much protection. Sometimes a merchant will give a "transaction reference number," though this number merely indicates that a transaction has taken place, without proving the time or amount involved. Merchants have started sending e-mails confirming payment transactions, and in some cases e-mail has been accepted as evidence in court. However, there is no guarantee that the e-mail will actually be sent and, moreover, even if sent, e-mail systems have been known to lose or be unable to deliver some documents.

The best assurance that the buyer has in this area is to deal with a large and well-established company. If the transaction amount is relatively small compared to the total business done by the company, then the company will not compromise its reputation over a small dispute. However, this is not always true, and may not apply if the transaction amount is large.

The issues at the merchant side are:

- The merchants would like to be assured that they are being paid exactly the specified amount. This issue is not a concern from the perspective of the merchant.
- In the case of a dispute the merchants would like to have in their possession enough evidence to prove whether or not the payment took place. The assurance is weak because there is little evidence that the buyer has agreed to pay the specified amount. For example, the buyer may call the credit card company and claim that the transaction never took place. The merchant has little evidence to argue on. The only evidence is if the buyer has registered at the site in a way that proves their identity, and somehow signed for the purchase. In this case, one has to prove that the registration site has good password security, and even then the buyer may claim that his/her password was stolen, or given away by the merchant.

As mentioned earlier, user authentication, privacy of information, and data integrity can be assured by using SSL. However, SSL is only part of the solution to the security of online credit card transactions. It does not address all the trust issues between the buyer and the merchant. An effective approach to ensure trust is to use a *trusted third party* (TTP), who is trusted by both the buyer and the merchant.

11.4 Trusted Third Party and a Payment Protocol Using a Trusted Third Party

Long before the invention of online commerce, people resorted to the use of trusted third parties to assure transaction integrity. For example, a notary public certifies documents for correctness, which gives such documents extra weight as court evidence. Another classic example is buying a house, which is a transaction performed "in trust" by the lawyers of the buyer and the seller. In this case, the lawyers act as "trusted third parties" on behalf of their clients.

In the online world, we believe that transactions should be similarly notarized by a trusted third party, and such notarization assures both parties of the integrity of the transaction. In this section, we describe a new payment protocol that uses a trusted third party, which can be used for credit card payments and other types of payments. This protocol has several advantages over existing protocols, e.g., SET, as described in Chap. 10. These advantages are discussed below:

- There is no sharing of payment instrument between the buyer and the merchant. This results in improved protection for both parties.
- There is no need for the buyer to register with any payment service (some protocols, such as CyberCash or PayPal, require this type of registration). As a result, buyers have increased flexibility and convenience in the choice of payment service.
- The protocol is designed as a HTTP-style request-response message protocol [11.4]. This approach reduces the complexity in implementation. For example, there is no need at any point to transmit a message to two destinations and expect the receiving parties to synchronize with each other. Instead, each step in the protocol involves a single message from the sender to the receiver, and a corresponding response back from the receiver to the sender.
- The protocol is robust in the sense that recovery is possible at any point in the case of a failure; hence the protocol protects both the merchant and the buyer from insufficient or excessive charges due to communication or system failures.
- The protocol can be used with any existing payment instruments, e.g., credit cards and debit cards. This offers increased versatility in terms of supporting a variety of payment methods.
- Supporting evidence is provided to the buyer and to the merchant regarding the nature of the transaction in case dispute resolution is required. This is accomplished by sending only a digest of the details of the transaction to the TTP.
- A single TTP may be used by both the buyer and the merchant. Alternatively, the buyer and the merchant can each be represented by their own TTP.

11.4.1 Description of the New Payment Protocol

In this section, the new payment protocol for the case of one TTP is described. The protocol requires the availability of a public-key certificate authority. Any certificate authority can be used, such as Pretty Good Privacy (PGP) or commercial providers, such as VeriSign.

The basic steps of our protocol are shown in Fig. 11.2. For each step, any message transfer (if required) is secured using cryptographic technology, such as SSL.

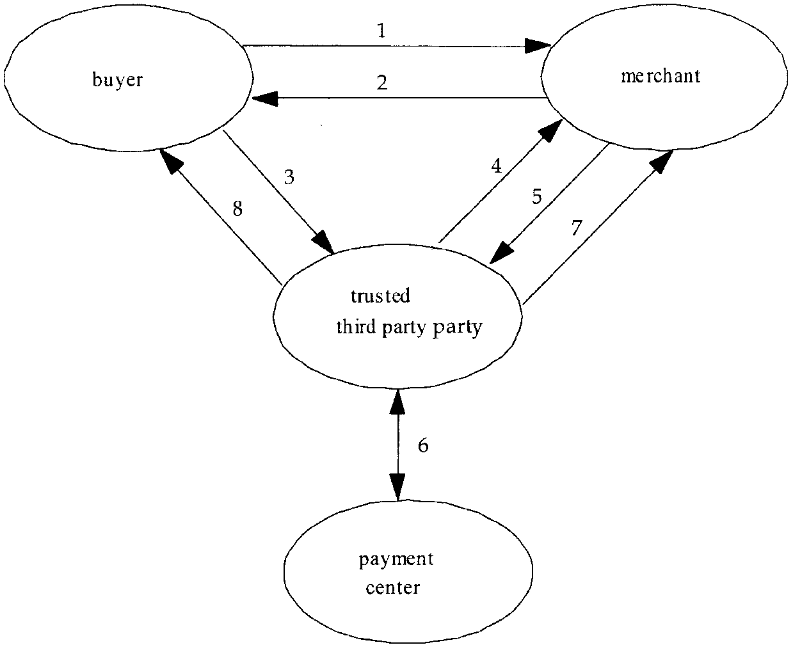


Fig. 11.2 Payment protocol with one TTP

There are eight steps in the protocol (detailed information on each step is provided in Sect. 11.6.2, Appendix B).

(1) The buyer sends an "order" message to the merchant.
(2) The merchant, upon receiving the order message, returns a "payment request" message to the buyer.

(3) The buyer, after verifying the merchant's signature, proceeds by sending a "payment" message to the TTP.
(4) The TTP, after verifying the buyer's signature, requests a confirmation from the merchant by sending a "confirmation request" message to the merchant.
(5) The merchant, upon receiving the confirmation request message, verifies the transaction ID and amount, and sends a "transaction confirmed" message to the TTP.
(6) The TTP obtains authorization from the payment center.
(7) The TTP sends a signed "merchant receipt" message to the merchant.
(8) The TTP sends a signed "buyer receipt" message to the buyer.

In these steps, all three parties, namely, the buyer, the merchant, and the trusted third party have each provided evidence of the transaction (a signed request or receipt). In the case of a dispute, the buyer has a signed payment request from the merchant and a signed receipt from the TTP. The merchant has a signed receipt from the TTP. The TTP has a signed payment from the buyer and a signed confirmation from the merchant. The signed information is sufficient for dispute-resolution purposes.

11.4.2 Extension to the Case of Two TTPs

For the case of a single TTP, the TTP could have a conflict of interest in the case of a dispute because it would be representing the interests of both the buyer and the merchant. For this reason, examples of candidates for trusted third parties are typically organizations, such as major banks or major credit card companies, which have no obvious vested interest in supporting either dispute party in a transaction.

In some cases, however, buyers and merchants may want to be represented by a trusted third party that is more active in supporting their concerns, and perhaps is targeted specifically at providing a service for their needs. In such a case, a conflict of interest can be avoided if two TTPs are involved, one for the buyer (referred to as TTP-B), and the other for the merchant (referred to as TTP-M). In the case of a dispute, TTP-B and TTP-M will be involved in dispute resolution, protecting the interests of the buyer and merchant, respectively. In addition, such a protocol with two TTPs has the potential to allow more types of organizations to assume the role of the trusted third party.

The payment protocol can readily be extended to the case of two TTPs. The basic steps are illustrated in Fig. 11.3.

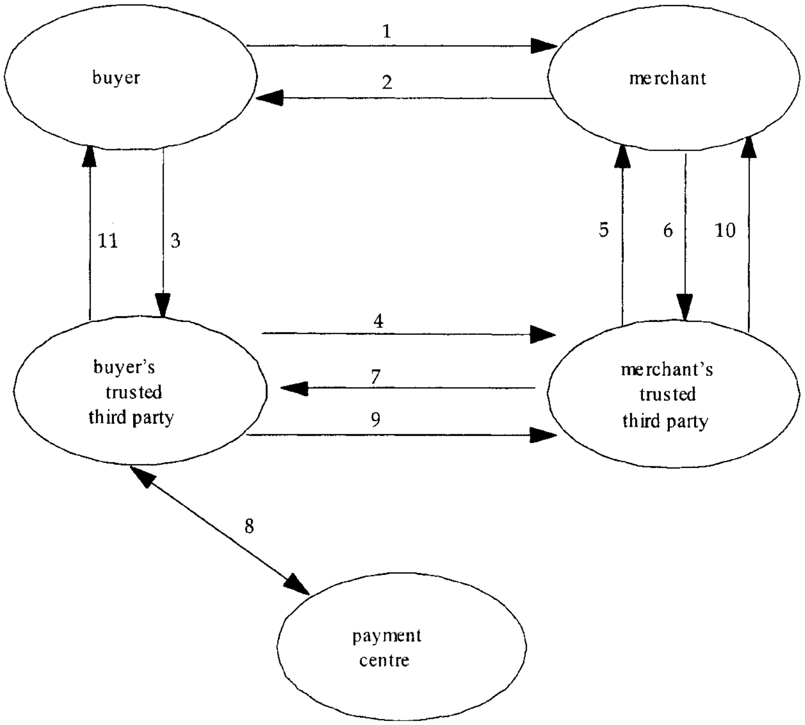


Fig. 11.3 Payment protocol with two TTPs

These steps are:

(1) The buyer sends order information to the merchant.
(2) The merchant requests payment from the buyer (message from the merchant to the buyer also contains the address of TTP-M).
(3) The buyer sends the payment information and the amount of the payment to TTP-B (this message also contains the address of TTP-M).
(4) TTP-B sends a message to TTP-M, requesting a confirmation from the merchant.
(5) TTP-M checks with the merchant to get a confirmation of the transaction and the amount of the payment.
(6) The merchant returns a confirmation to TTP-M.
(7) TTP-M forwards the confirmation to TTP-B.
(8) Message exchange between TTP-B and the payment center regarding the payment authorization.
(9) TTP-B sends a receipt to TTP-M, confirming the payment.
(10) TTP-M forwards the receipt to the merchant.

(11) TTP-B sends a receipt to the buyer.

The message contents, together with the actions taken by the buyer, the merchant, TTP-B, and TTP-M at the various steps are straightforward extensions of those for the case of one TTP (see Sect. 11.6.2, Appendix B), and will not be presented.

11.4.3 Discussion

It is important to trace the business reason for the viability of this payment protocol.

Typically, once a buyer gives the payment information to the merchant, it is the merchant's responsibility to ensure that payment is captured from the corresponding financial institution. Note that in some cases the financial institution may insure the buyer against fraudulent use of the payment information. This is the case with credit cards, where most issuers do not hold buyers responsible for payment of unauthorized transactions on their cards. The cost of this insurance is imposed on the merchants, who must pay an increased fee for online transactions.

With our protocol, some of the burden of transaction insurance is shifted to the trusted third party. The buyer's TTP provides assurance to the buyer that the correct amount of payment will be collected and passed to the merchant. At the same time, insurance against unauthorized transactions may now be split among the financial institution and the TTP, depending on the service agreement.

The merchant's TTP must provide assurance to the merchant that the buyer's payment instrument will be charged, and funds transferred to the merchant. This had been the merchant's responsibility, but is now a burden carried by the TTP.

Clearly, for the protocol to be successful there needs to be a business case to operate a buyer or merchant TTP service. The following are sample business cases:

- The TTP charges the merchant a transaction fee (similar to current credit card transaction fees), and carries the entire weight of the fraudulent transaction insurance. Typically, credit card companies charge a large premium for Internet-based transactions. By charging merchants a fixed or transaction-based service fee, the TTP would fund the insurance. In this scenario, the TTP is effectively competing with credit card companies for a portion of the payment transaction fees.

- The buyer's TTP charges the buyer a fixed service fee, or a transaction fee. The buyer would use the service because it results in more trusted transactions, with extra security provided by the TTP, as well as the TTP's enhanced insurance policy.

- The TTP charges the merchant a fixed service fee. The merchant uses the service to attract buyers who demand the use of a trusted payment protocol.

- A financial institution provides a single TTP service, to stimulate online commerce by engendering trust, and hence potentially increase its transaction revenue.

11.5 Summary

In this chapter, an overview of credit card-based online transactions is presented, including payment authorization, capture, and settlement. The issue of trust between buyers and merchants is analyzed. We have also described a new protocol for secure online payment. This protocol allows a buyer to avoid sharing key payment method information with merchants, by moving the burden of payment assurance to a trusted third party. The protocol supports privacy because the order information is not sent to the trusted third party. The buying operation is simplified because there is no need to register with a payment service.

11.6 Appendices

11.6.1 Appendix A: the Main SSL Exchanges

In this section, we provide detailed information on the main exchanges in an SSL session.

To establish an SSL session between an SSL client and server, they need to agree on the SSL protocol version to be used, select cryptographic algorithms, optionally authenticate each other, and use public-key encryption techniques to generate and exchange shared secrets. The main exchanges between the client and the server during the establishment of an SSL session are described as follows:

ClientHello and ServerHello messages: These two messages are used to establish security enhancement capabilities between the client and the server. The attributes of the two messages include:

- Protocol version
- Session ID
- List of cryptographic options
- Compression method
- Random number

Server certificate: The server sends its certificate, if it is to be authenticated. The certificate format generally follows the X.509.v3 standard [11.7].

Server key exchange message: This is optional. It is required for certain circumstances, for example, if the server has no certificate, or if its certificate is for signing only. The attributes include a key exchange algorithm and associated parameters (e.g., RSA algorithm with RSA modulus and exponent).

ServerHelloDone message: This message indicates that the hello-message phase of the handshake is complete. After sending this message, the server will be in a mode to wait for a client response. The client verifies that the server provided a valid certificate if required and checks that the ServerHello parameters are acceptable.

Client key exchange message: This message has different formats, depending on which public-key algorithm has been selected between the ClientHello and the ServerHello messages. Once again, similar to the Server key exchange message, the attributes include a key exchange algorithm and associated parameters.

ChangeCipherSpec message: This is not a handshake massage. After this message is sent, the pending CipherSpec is transferred into the current CipherSpec.

Finished message: This message is always sent immediately after a ChangeCipherSpec message to verify that the key exchange and authentication processes were successful.

11.6.2 Appendix B: Steps of the Payment Protocol for the Case of One TTP

In this section, we provide the detailed steps of the payment protocol for the case of one TTP.

The following notation is used in our description:

$CERT_j$	j's certificate (j = b for buyer, j = m for merchant, j = t for TTP)
H(x)	Cryptographic digest of x
$S_j(y)$	Signature on information set y using private key of j (j = b for buyer, j = m for merchant, j = t for TTP)
*	Optional field

Step 1. The buyer sends an "order" message to the merchant.

The "order" message contains the following information:

> order = items to be purchased, shipping information, *previously quoted price, *time stamp

The previously quoted price is an optional field. The time stamp is an optional field included to prevent a replay attack. An intruder performs a replay attack by intercepting a protected message, and replaying it at a later time. The timestamp contained in a received message can be used to determine whether this is a replay of a previously received message or not.

Step 2. The merchant, upon receiving the order message, returns a "payment request" message to the buyer.

The "payment request" message contains the following information:

> payment request = transaction ID, amount, order, validity period, $CERT_m$, *purchase agreement, S_m(transaction ID, amount, order, validity period, $CERT_m$, *purchase agreement)

The transaction ID is generated by the merchant and used by the merchant and the TTP to keep track of all the transactions. The order information is the same as that provided by the buyer. The validity period specifies the time during which the payment must be confirmed. The merchant's certificate can be used by the buyer to verify the merchant's signature. The purchase agreement is an optional field which contains information such as refund policy, product quality, warranty, etc. A digital signature is included as part of the payment request message.

Step 3. The buyer, after verifying the merchant's signature, proceeds by sending a "payment" message to the TTP.

The "payment" message contains the following information:

> payment = payment information, amount, merchant, transaction ID, $CERT_b$, *timestamp, S_b(payment information, amount, merchant, transaction ID, $CERT_b$, *timestamp)

The payment information field contains information such as the credit card number, credit card holder, and expiry date. Besides credit cards, other payment instruments such as debit cards can be used. The transaction ID is the same as that provided by the merchant. The buyer's certificate can be used by the TTP to verify the buyer's signature. Again, an optional time stamp may be included to prevent a replay attack. A digital signature is included as part of the payment message.

Step 4. The TTP, after verifying the buyer's signature, requests a confirmation from the merchant by sending a "confirmation request" message to the merchant.

The "confirmation request" message contains the following information:

> confirmation request = transaction ID, amount, status, S_t(transaction ID, amount, status)

This message contains the transaction ID, amount, and payment status. A digital signature is included as part of the confirmation request message.

Step 5. The merchant, upon receiving the confirmation request message, verifies the transaction ID and amount, and sends a "transaction confirmed" message to the TTP.

The "transaction confirmed" message contains the following information:

> transaction confirmed = transaction ID, amount, status, S_m(transaction ID, amount, status), *H(transaction ID, amount, order, validity period, *purchase agreement), S_m(H(transaction ID, amount, order, validity period, *purchase agreement))

This message contains the transaction ID, amount, and payment status. As an option, a cryptographic digest of the transaction details (namely the transaction ID, amount, order, validity period, purchase agreement), as contained in the payment request message in step 2, may be included. This digest is useful for dispute-resolution purposes. A digital signature is included as part of the transaction-confirmed message.

Step 6. Obtain authorization from the payment center.

Upon receiving the transaction-confirmed message, the TTP requests the authorized amount from the payment center. The payment center returns an approval to the TTP.

Note that any payment method can be used in this step. Note also that the requirement for payment approval is tied to the TTP's policy. It is possible that in some cases (e.g., for preferred customers) the TTP would not wait for credit approval, but would process the payment immediately. In this case, the TTP, rather than the payment center, would be taking on the responsibility for the payment.

Furthermore, different TTPs may have different policies on handling unknown or delayed credit approval requests. For example, if the approval request times out, the TTP may either refuse to process the payment, or may take the risk of processing it. Similarly, even if the payment center rejects the request, the TTP may still process it, taking on the payment responsibility as described above.

Step 7. The TTP sends a signed "merchant receipt" message to the merchant.

The "merchant receipt" message contains the following information:

> merchant receipt = payment ID, transaction ID, amount, S_t(payment ID, transaction ID, amount)

Step 8. The TTP sends a signed "buyer receipt" message to the buyer.

The "buyer receipt" message contains the following information:

> buyer receipt = payment ID, transaction ID, amount, S_t(payment ID, transaction ID, amount)

After step 8, the TTP captures the payment and transfers the funds to the merchant. This step happens offline and involves the actual payment settlement.

Note that the above steps are sequential in nature. The transaction is not complete until the last step (step 8) is performed. A timer is used at each step to protect against unusual situations where one of the parties (buyer, merchant, or TTP) is not proceeding to the next step within a predetermined time interval. For steps 1 to 6, if the timer expires the transaction is assumed to be aborted. Any subsequent messages regarding this transaction will be ignored. Therefore, up to this point, any party can abort the transaction by simply not continuing with the next step.

At step 7, if the merchant does not receive a receipt within a time-out period, the merchant attempts to obtain the receipt by sending a request message to the

TTP. If a receipt is not received after a predetermined number of attempts, the transaction is assumed to be aborted. In this case, the buyer will not receive the order, but he/she can contact the TTP to request a refund.

At step 8, if the buyer does not receive a receipt within a time-out period, the buyer may request a receipt from the TTP at a later time. This would not affect the transaction because the order will be shipped by the merchant as long as the merchant has received the receipt.

11.7 References

[11.1] A. O. Freier, P. Karlton, P. C. Kocher: SSL 3.0 protocol specification. http://home.netscape.com/eng/ssl3/index.html.

[11.2] MasterCard International Incorporated, Visa International. The SET specification. http://www.setco.org/set_specifications.html.

[11.3] D. Eastlake, B. Boesch, S. Crocker, M. Yesil (1996) CyberCash credit card protocol version 0.8. Internet RFC1898.

[11.4] R. Fielding, J. Gettys, J. Mogul, H. Frystyk, L. Masinter, P. Leach, T. Berners-Lee (1999) Hypertext Transfer Protocol – HTTP/1.1. Internet RFC2616.

[11.5] X. Lin, J. W. Wong, W. Kou (2000) Performance analysis of secure web server based on SSL. In: Proceedings of ISW2000, J. Pieprzyk, E. Okamoto, J. Seberry (eds.), LNCS 1975. Springer. Berlin Heidelberg New York, pp. 249–261.

[11.6] A. O. Freier, P. Karlton, P. C. Kocher (1996) The SSL protocol version 3.0. http://wp.netscape.com/eng/ssl3/ssl-toc.html

[11.7] W. Ford, M. Baum (1997) Secure electronic commerce: building the infrastructure for digital signatures and encryption. Prentice-Hall, Englewood Cliffs New York.

12 Micropayments

Amir Herzberg

Security Consultant
Israel

12.1 Introduction

Open data networks, such as the Internet and the wireless data networks, allow low-cost delivery of content (information) and services to a huge population (market). The production costs of content and services are often small and largely independent of the number of customers. Therefore, producers of content and services provided to many customers often want to charge very small amounts – if the payment system allows it (with reasonable overhead). Payment by credit cards, which is the common method for online consumer purchasing, involves substantial minimal fee per transaction, e.g., 20 cents, and therefore is not applicable for charging smaller amounts. This provides one definition of the micropayments, as charging amounts smaller (or close to) the minimal credit card transaction fees (of about 20 cents). There are other difficulties in using credit cards for low-value transactions, namely, substantial delay and user involvement, and the potential for disputes resulting in refunds, chargebacks, and substantial handling costs.

This creates a difficulty for many existing and potential applications and services on the Internet[1], which need a source of income to cover their costs and generate profits while the amount they can charge (for one use) is too low to justify a credit card transaction. Currently, most of the deployed services and applications are funded only by advertising or by charging substantial amount in advance for multiple purchases (e.g., subscriptions). A direct-payment mechanism could be an important alternative or complementary source of funding, especially to facilitate smaller vendors and applications where advertising cannot be used (e.g., due to lack of appropriate display, and in particular when services are consumed by automated agents without any advertising potential). This motivates the development and introduction of *micropayment* schemes and systems.

In this chapter, we focus on providing micropayment services with acceptable (low) transaction cost. This is the basic requirement from a micropayment system, namely, that it can be used to charge sufficiently low amounts, in particular, below

[1] We only mention the Internet but the discussion applies to most open networks.

credit card minimal fees (of about 20 cents). The minimal amount to be supported may be considered as a parameter of the system, or there may be a specific requirement. In particular, when considering payments that involve a manual decision element (by a person), it seems that a minimal amount of about one cent may be sufficient, as the cost of the decision process itself is probably worth about a cent, and smaller-value items should probably not require specific user decision and action (otherwise, a lower-denomination coin would have been introduced). When considering payments by a software agent of the user, e.g., to pay for the actual communication services, there may be room for payments of amounts even significantly smaller than one cent.

In Section 12.3 below we analyze the different cost factors for online payments, and in Section 12.4 we elaborate on different mechanisms used to reduce each of the significant cost factors. But first, in the next section, we provide an overview of micropayment systems.

12.2 Overview of Micropayment Systems

There are different motivations for developing new payment mechanisms. We focus on micropayments mechanisms that are operated by one or more payment service provider (PSP), allowing merchants to charge small amounts from customers. There have been many different definitions, goals, and proposals for micropayment mechanisms, including low-value offline payments (using a device rather than coins), anonymous payments (digital cash), and systems where the merchant charges a large amount once but allows the customer to use it incrementally over many small purchases (merchant acting also as a micropayment service provider).

Our focus is on the most common interpretation, namely, many payments of small amounts (micropayments) from customers to merchants, over open-data networks, such as the Internet, made by using one or more payment service providers. A PSP maintains a long-term relationship with customers and merchants, receiving payments of aggregated (large) amounts from customers and passing aggregated payments to the merchants, as illustrated in Fig. 12.1. This model assumes that consumer relationships with merchants are sporadic rather than long-term, and that a major role of the PSP is to provide facilities for efficient and secure transactions by using its relationships with the parties.

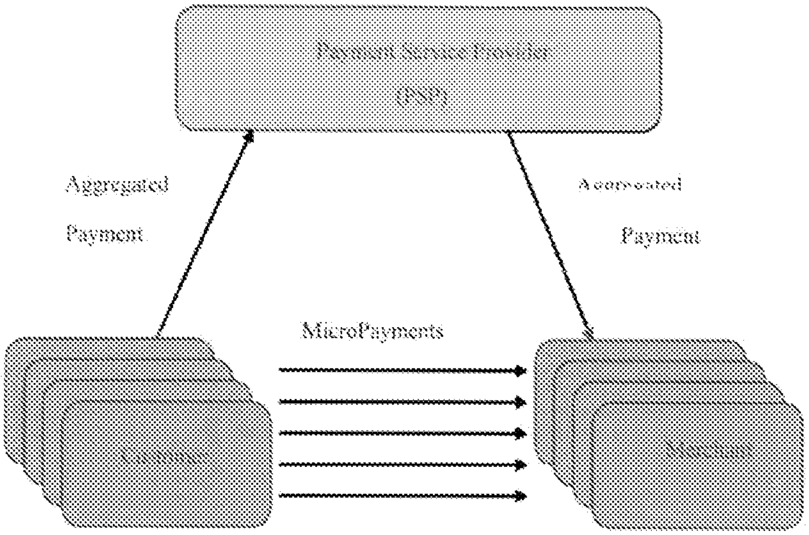


Fig. 12.1 Micropayments via a single PSP

Fig. 12.1 shows the payment relationships between the parties: sporadic micropayments from consumers to merchants, and long-term, usually periodic, payments of aggregated amounts from customers to PSP and from PSP to merchants. This does not describe the flow of messages for a micropayment transaction; we will describe different protocols, with different message flows. Payment protocols include mechanisms for *payment approval* by the customer, where the customer agrees to pay, as well as *payment authorization* by the PSP, where the PSP indicates that there are funds to cover the payment.

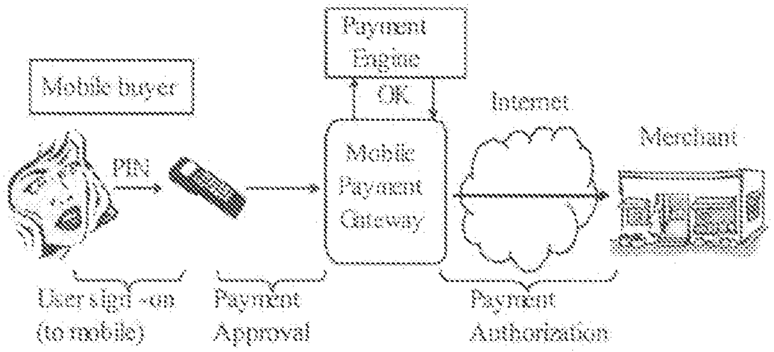


Fig. 12.2 Payments via the PSP

Payment approval and payment authorization may be integrated or separated. Separation of the payment-approval process from the payment-authorization process is appropriate, in particular, in scenarios where the PSP is (or controls) a gateway between the customer and merchant, as illustrated in Fig. 12.2. The customer approves the payment to the PSP, and then the PSP sends an *authorized payment order* (PO) to the merchant. This scenario is applicable whenever the PSP is also providing the communication services to the consumer or is in close alliance with the communication providers. In particular, this scenario is appropriate when the PSP is also the consumer's ISP or a mobile-communication provider, or when the ISP or mobile provider are cooperating with the PSP to improve the user interface for payments. In this case, the payment authorization is, naturally, always online and involves only the PSP and the merchant. Furthermore, payment authorization can be completely independent from the payment approval process between the consumer and the PSP, and certainly from the login process (if any) between the consumer and their computer or device. In particular, in this scenario we can take advantage of existing security mechanisms between the consumer and their ISP or mobile gateway to validate that the consumer approved the payment. For example, a mobile gateway usually can identify the handset, e.g. using a shared key, and the handset may identify its user, e.g., using PIN, voice recognition, or any other identification technology.

In other scenarios, the PSP is not "on the path" between the consumer and merchant, and therefore either consumer or merchant should contact it to request authorization for payment when required. This is typical, e.g., for web browsing, when the PSP is not the ISP (or in alliance with the ISP). In most micropayment systems, the consumer contacts the PSP to approve the payment and to request the PSP to authorize the PO. For technical reasons, namely, allowing the PSP to operate as an efficient server application, the PSP sends the authorized PO as a response to the consumer, who forwards it to the merchant. The merchant will later (offline) deposit the PO, often in a batch process with many other payment orders, to receive the aggregated payment.

Fig. 12.3 presents a high level illustration of the online payment process in this scenario (payment invoked by the consumer), as implemented by most currently-deployed micropayment systems, e.g., by Qpass, iPin, and TrivNet. We will later also discuss systems where the merchant is requesting the authorization from the PSP, or where there is no online payment authorization.

So far, we have discussed only a single PSP providing service to both customer and merchant. The single-PSP solution is simple and efficient. However, currently, there is no dominant single PSP for micropayments. Indeed, there are a substantial number of competing PSPs for micropayments, and we can expect more PSPs to emerge as the demand for micropayments grows and the market matures. In fact, the expected financial returns from a micropayment system may not be high enough to justify a sufficient effort by a single PSP, or even a small number of PSPs, to gain market dominance (in contrast to the 2 ~ 3 major credit card brands). We would therefore expect that there will be multiple PSPs offering micropayment services.

It is unrealistic to expect all customers and all merchants to have accounts with multiple PSPs. Instead, we expect that micropayment systems will need to support interoperability among multiple PSPs, each with its own customers and merchants, with aggregated payments and long-term relationships between the PSPs, allowing customers of one PSP to make purchases from merchants of the other PSP. A simple architecture with two PSPs is illustrated in Fig. 12.4.

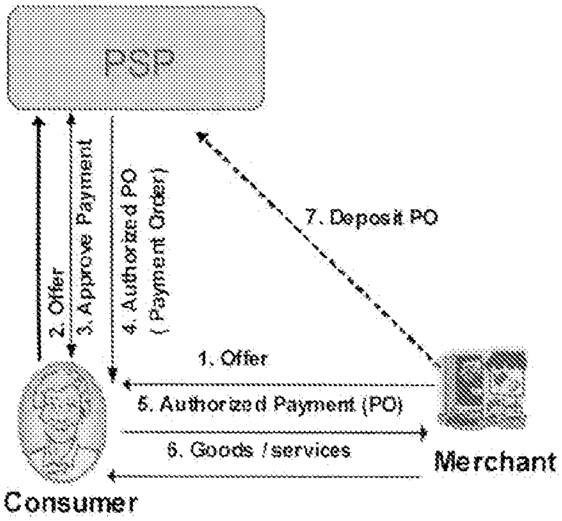


Fig. 12.3 Online payments invoked by consumer

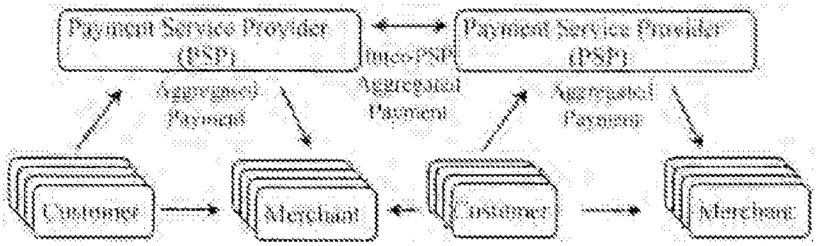


Fig. 12.4 Micropayments via two interoperating PSPs

12.3 Cost Factors for Online Payments

In order to allow economical charging of small amounts, we need to consider the different cost factors that, by affecting the payment service provider, affect the merchant directly and indirectly. Indeed, from the merchants' perspective, it may be enough to use a PSP without the substantial minimal-fee requirement of existing credit cards; but these costs are not pure profit, and the PSP needs to have a viable business case. In order to design a mechanism with substantially reduced costs, as compared to credit cards, we need to consider (and minimize) at least the most significant cost factors.

In the following sections, we consider three major categories of costs:

- **Disputes, chargebacks and their processing cost:** Many payment systems, and even certain laws, allow the customer to dispute charges, or otherwise not to pay, usually under certain circumstances. In some or all of these cases, the PSPs may *reverse* the transaction, requiring the merchant to return the funds (*chargeback*). In particular, payment orders received electronically, without a signed authorization, are often reversible. Indeed, disputes and subsequent chargebacks are substantially more common for Internet transactions than for face-to-face transactions that have been authorized with the signature of the customer. The costs here include the actual refund amount, as well as substantial processing cost (for the PSPs and the merchant) and possible penalty payments (by the merchant). The processing costs of credit card payment service providers are estimated at about $50, with penalties payments for merchants with frequent disputes of about $100 [12.34]. This is probably the most critical cost factor for micropayments, and much of the work on micropayment systems is targeted at reducing the expenses associated with it (see Section 12.4).

- **Customer acquiring and support costs:** These are the costs of encouraging customers to deploy the new service (open an account, install a wallet, etc.), and later assisting customers. These expenses may be substantial, especially compared to the small value of the transaction. The main mechanism to reduce these costs is to use simple procedures and user interface. In particular, it is highly desirable to offer an easy-to-use, "click and pay" user interface for micropayments. On the other hand, to minimize installation and support costs, the customer should be able to use standard software tools (e.g., a browser) rather than installing customer software (wallet) on the consumer's machine. Finally, PSPs should be interoperable, namely, the customer of one PSP should be able to buy from a merchant of another PSP, as shown in Fig. 12.4, so that multiple

PSPs share the customer acquiring and support costs (See more details in Section 12.5).

- **Equipment, processing, and communication costs:** These are the costs of the necessary hardware, software, and communication for processing the payments by customer, merchant, and PSP. These costs are a function of the processing and communication requirements of the payment protocol, including the dependency on online involvement of the PSP, requiring high availability. Indeed, most of the research on micropayments, and several of the deployed systems, focus on minimizing the processing and/or communication costs. In particular, many efforts have focused on reducing the processing costs by avoiding public-key operations. Another area that received a lot of attention is reduction in communication requirements, and in particular, allowing offline or semi-offline payments (in particular, "stored value" offline payments, where the merchant and browser are in direct connection but disconnected from the PSP. See more details in Section 12.6).

There are several additional cost factors, which are less significant or easier to deal with, such as:

- **Bookkeeping and auditing costs:** Many payment systems have substantial bookkeeping and auditing mechanisms and costs. It is tempting to suggest that these costs can be eliminated by simply not logging and auditing micropayments, or keeping only very partial and temporal logs. However, accurate logging and auditing is often required by law, and may also be necessary to provide non-repudiation for efficient dispute resolution and to detect fraud. Bookkeeping and auditing costs may be reduced by secure automated *record aggregation* mechanisms, e.g. [12.13], whereby the customer signs a single document, which is archived instead of multiple separate documents (similarly to the presentment of a monthly statement by utilities). Record aggregation may reduce, also help, to protect the privacy of the customer by not keeping track of individual transactions for long. Other approaches try to protect the privacy of the buyer even further, by preventing the PSP from identifying payments of a particular customer, using one of the many anonymous (digital) cash protocols, e.g. [12.3]. Some, e.g. [12.1], believe that anonymous payments would also be less expensive, by avoiding bookkeeping (almost) entirely and preventing disputes and chargebacks.

- **Point-of-sale integration costs:** These are the costs for a merchant for setting up merchandise for sale, and for publishing information and services. These costs can be reduced by simple, automated tools for the merchants (for small merchants, and for initial phases), hosting services by the PSP may also be desirable. However, these are one-time expenses and should usually be rather insignificant in the long run.

- **Credit risk:** When the customer is charged for the aggregated payments only after the purchases are made, the customer may refuse to pay their PSP. Often, PSPs eliminate this risk by requiring funds to be deposited in advance. This problem also appears when interoperating between multiple PSPs, where one PSP ends up owing the other PSP; in this case, prepayment is rarely a solution, since usually both PSPs may end up owing the other PSP. This becomes a risk-management issue, with associated costs of estimating and containing the risk.

12.4 Disputes and Chargebacks

As noted above, disputes and chargebacks, and in particular, their processing costs, are of the most significant expense factors for online credit card purchasing. In credit card purchasing, and to some extent in any remote purchasing, such as through the Internet, there are laws protecting consumer's right to dispute and reverse transactions that were not approved by the consumer, and often to some extent also transactions that were not properly fulfilled. This creates a difficult challenge to the designers of micropayment systems. In general, disputes and chargebacks for electronic payments fall into four main categories.

- **Disputes on whether payment was approved by consumer**: Consumer claims that they did not *approve* the payment.
- **Unauthorized overspending chargebacks:** PSP claims that it did not *authorize* the payment, and that there are not sufficient funds in the customer's account to cover it (in systems where the merchant should receive authorization from the PSP for each payment).
- **Disputes on delivery and/or quality**: Consumer claims that the merchant did not *deliver properly* the goods or services as ordered.
- **Chargebacks due to consumer default**: PSP claims that the consumer *defaulted,* and did not provide necessary funds.

We now discuss each of these categories for disputes and chargebacks, beginning with the last two, which we believe should simply be disallowed for micropayment systems. But first let us consider an example.

12.4.1 Example: First Virtual Payment System

A simple example is the First Virtual payment system, illustrated in Fig. 12.5, which was of the earliest payment systems proposed [12.36]. One of First Victual's goals was to offer lower fees for small charges (compared to credit cards) and avoidance of chargebacks. First Victual's system used two (non-cryptographic) security mechanisms for each purchase: a secret "first virtual account number" (FV#) provided by the consumer to the merchant and forwarded to the First Virtual Net server (on the Internet), and an email approval request, sent from the First Virtual Net server to the consumer and approved (or declined) in a message from the consumer back to the First Virtual Net server. Only after the consumer approves the payment, the merchant's account with First Virtual is credited, the merchant is informed that the payment was cleared. A credit card transaction is performed (immediately or periodically) to collect the funds from the customer's credit card account, whose details were kept in a separate First Virtual "Credit Card Server".

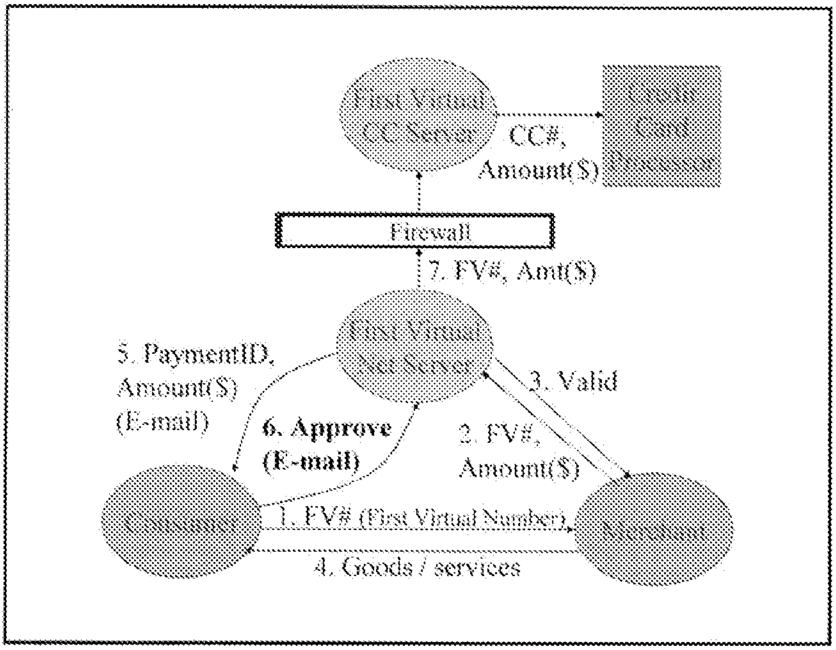


Fig. 12.5 First Virtual's centralized PSP based on email confirmation

The payment-approval process of First Virtual is subject to attacks, since standard Internet email is not secure; this makes the entire authorization process insecure. It is particularly easy to inject e-mail messages with incorrect source identification; First Virtual protects against this attack by including a random payment transaction identifier in the approval-request email message (PaymentID). However, the attacker may be able to intercept the email message in transit to the consumer, and then send the approval response with the correct PaymentID. To protect against this, First Virtual suggested that consumers may use available secure email software to digitally sign their payment approval responses.

12.4.2 Consumer Default and Disputes on Delivery/Quality

We believe that micropayment systems should not allow disputes on product delivery or quality or on chargebacks due to consumer default. The basic reason is that there may not be a long-term relationship between the buyer and the merchant. Therefore, the merchant can hardly manage such risks, e.g., the risk of the consumer defaulting on its payment obligations to its PSP.

In particular, consider disputes about whether the purchased product or service was in fact delivered, or about the quality of the product or service. Some proposed and deployed systems attempt to avoid disputes on product delivery, by using a trusted third party as an "escrow agent" that ensures delivery of product (to buyer) and payment (to merchant). Most of the deployed systems, e.g., [12.6, 12.21], focus on escrow of high-value physical products (or legal title, e.g., to real estate), and in particular involve minimal fees of several to dozens of dollars. Proposals have also been made for escrow for digital content (for efficiency, the actual escrow is often of a key to decrypt the content, see [12.4]), which may seem applicable to micropayments. However, it is impossible to completely automate the resolution of disputes regarding quality of (most) products and services. Therefore, the third party will require manual intervention for every dispute, making the process rather expensive, and inappropriate for micropayments. Therefore, micropayment service providers should prohibit this kind of dispute.

By disallowing disputes related to proper delivery/quality and chargebacks due to consumer default, micropayment systems can focus on disputes related to payment approval (by the customer) and payment authorization (by the PSP).

12.4.3 Unauthorized Overspending Chargeback

In some payment systems, a customer may overspend, i.e. approve more payments than the funds available in her account. In this case, the PSP may wish to refuse to pay (or chargeback) the merchant. Some micropayment systems allow this, claiming that merchants can accept this risk, since the amounts are small, and especially for selling information (with negligible cost for the merchant for each extra transaction).

However, since merchants do not have a long-term relationship with consumers, they often require secure *payment authorization* from the PSP, such that payments properly authorized by the PSP cannot be reversed. Micropayment schemes often focus on reducing the overhead required for payment authorization, especially on the PSP, by adopting different strategies. The overhead for payment authorization has two major elements: communication and computation.

First, consider the computational overhead, and in particular, whether the PSP must perform computationally-intensive operations such as public key signature for each payment. Public key signatures are the main technique for achieving non-repudiation. For large amounts, and when the merchant does not trust the PSP, the merchant may require non-repudiation of the payment authorization from the PSP, to make sure that the PSP is committed to transfer the funds for the payment. In particular, in multi-PSP scenarios as illustrated in Fig. 12.4, the merchant may require non-repudiation from the customer's PSP. In Section 12.6, we discuss techniques for achieving non-repudiation with reduced computational requirements, compared to digitally signing each payment authorization.

When the transaction amounts are small relative to the value of the PSP-merchant relationships, then the merchant may not demand non-repudiation for every (micropayment) transaction, and agree to receive only secure payment authorization from the PSP (not signed). When the aggregated amount of authorized (but not signed) payments exceeds some merchant-specified threshold, the merchant may require the PSP signature authorizing the total, aggregated amount (providing non-repudiation). By not signing (and validating) every micropayment authorization, the PSP (and merchant) may save some computations.

Consider now the communication required for the PSP to authorize payments. Several micropayment schemes support *offline payment authorization*, i.e., transactions where the client communicates with the merchant, without involving the customer's PSP to authorize the payment. Offline transactions can be used where communication with the PSP during the purchase process is impractical, e.g. for payment using direct communication between consumer and merchant, without requiring connectivity and communication with the PSP. Usually, the PSP will require the merchant to return funds in case of double spending; but sometimes PSPs may assume limited liability, usually when the consumer is using a tamper-resistant hardware that authorizes the payments on behalf of the PSP (and keeps track of the available funds in the consumer's account). The PSP may pre-authorize payments up to a predefined amount for a particular merchant, but this is rarely useful (since it is hard to predict purchases) and indeed rarely used.

Therefore, whenever communication with the PSP during the purchase process is feasible, micropayment schemes should use it to perform *online payment authorization.* Sometimes, as in Fig. 12.2, all communication between the consumer and the merchant flows through the PSP (or a gateway associated with it), in which case the online payment authorization does not add substantial overhead. In other cases, the customer or the merchant must contact the PSP to request payment authorization. When possible, it is preferable for the customer to request the (online) payment authorization, so that, after local validation, the merchant can respond to the purchase request from the customer immediately without having to keep the connection with the customer open while requesting authorization from the PSP.

12.4.4 Disputes on Whether Payment Was Approved by Consumer

We now focus on the most important (and common) type of disputes and chargebacks, where the customer claims they did not approve the payment transaction. To understand this problem better, consider the common mechanism of performing credit card transactions over the Internet (or phone), by sending the credit card details to the merchant. Usually, the credit card details are encrypted on transit using SSL/TLS [12.27-12.28], as illustrated in Fig. 12.6. Since payment approval is only by inclusion of the credit card number, at any tome in the transaction, an attacker with access to the card number may invoke an unapproved transaction. As a result, there are many disputes in Internet transactions, which are much larger than proportion to all credit card transactions. The dispute rate in Internet transactions was so high that it caused losses to credit card issuers, and in the first years of Internet commerce, the rate grew rapidly (for example, see practical experience in [12.34]). The rate of disputes was considerably reduced with the introduction of substantial penalties to merchants with high dispute rates[2]. Many disputes were due to unauthorized purchases by a third party that somehow got hold of the customer's details, by exposure in Internet or non-Internet transaction, or otherwise. In addition, there are disputes made by customers denying payments they actually approved, knowing that there is no way to distinguish between them and unapproved transactions, namely, that there is no non-repudiation.

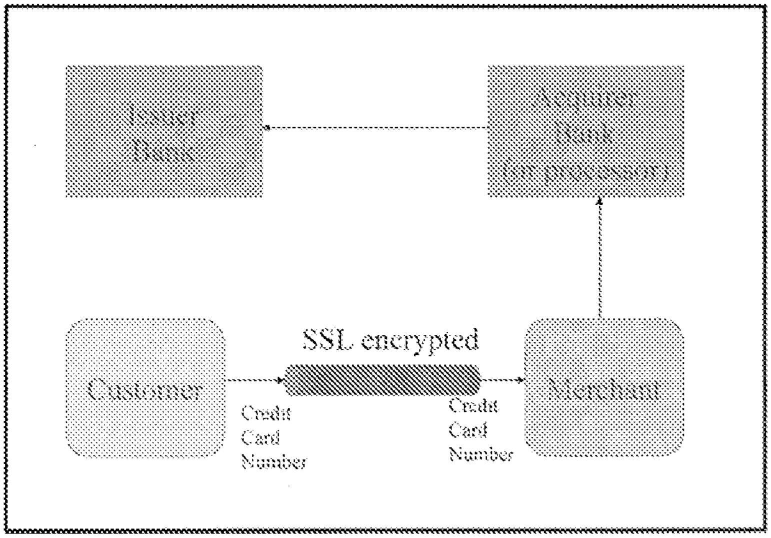


Fig. 12.6 SSL credit card payments

[2] Typical penalties for disputes: merchants with over 2.5% disputes among their transactions, pay 100$ per dispute [34].

Due to the high cost of dispute resolution and chargebacks, there have been several proposals to improve the security of the consumer approval process for credit card payments over the Internet. This has resulted in the iKP protocol [2], which evolved to the SET credit card payment standard [12.33, 12.20], and later in other proposals, such as 3D Secure and secure payment application (SPA) [20]. Most of these proposals attempt to emulate "face to face" transactions, where the customer physically signs a payment order (and presents a physical card with the same signature), often by using digital signatures. The goal is to reduce the number of disputes by preventing unauthorized use and preferably ensuring non-repudiation.

Reducing the number, and associated cost, of disputes resulting from claims of payments that were not approved by the consumer is even more critical for micropayment systems, where the average transaction amount is several orders of magnitude below the reported cost of handling a dispute in the credit card business. In the rest of this section, we discuss three major approaches to reducing the number of disputes and chargebacks due to consumer denial of approving the payments, and the associated costs.

The first approach is to perform a *secure payment approval* process, confirming to the PSP that the consumer approves the transaction. This would prevent an attacker from initiating transactions without consumer approval, which the consumer would later dispute. It also allows the PSP to detect when a consumer is disputing a properly approved transaction, although the PSP may not be able to "prove" to a third party that the transaction was properly approved, as long as non-repudiation is not provided. Possibly, the agreement between the consumer and the PSP may indicate that the consumer is willing to accept the records kept by the PSP of transactions approved by the consumer, and not to dispute them.

The second approach takes another step and provides not only secure payment approval, but also *non-repudiation* of the payment approval. Namely, whenever the PSP authorizes a transaction, it will retain a proof that allows it to convince a third party that the consumer, indeed, properly authorized the transaction. The proof will usually be in the form of a payment order digitally signed by the consumer. When complemented with an appropriate agreement with the consumer, accepting the digital signature or other proof as sufficient evidence to the consumer's approval of the transaction, this may be sufficient to prevent disputes. The agreement may also require the consumer's signature only when the aggregated amount of approved payments exceeds some predefined threshold, thereby amortizing the computational overhead over several micropayments.

The third approach takes a somewhat radical view, namely, that the only legally-acceptable way to refuse to reverse a transaction is when the PSP is technically *unable to reverse transactions*, since the necessary records do not exist. We call such payment instruments *bearer certificates*, since the payment is done by passing a message (*token, bearer certificate*) to the merchant, who deposits it at the PSP, but the message does not indicate the identity of the consumer. Designs that try to ensure irreversible transactions sometimes use *anonymous payment* ("digital cash") mechanisms, such as blinded signatures.

We discuss each of these approaches in more detail in the following sections.

Secure Payment Approval

Secure payment approval is relatively simple between two parties with a long-term relationship, such as the customer and their PSP. In this case, the payment approval is by an authenticated message from the customer to the PSP. Often, end-to-end authentication of the payment approval message from customer to PSP is possible (typically, when the customer communicates directly with their PSP, as in Fig. 12.2 or Fig. 12.3). Such end-to-end authentication of the payment approval requires only standard, pre-installed software components. The payment-approval process, when invoked by the consumer using only standard browser, as illustrated in Fig. 12.3, has the following steps:

1. The merchant presents the offer of goods or services to the consumer. In the usual browsing scenario, the offer is in a regular or special hypertext link, sometimes called a *per-fee link,* invoked by the consumer to pay, and encoding the payment details necessary to create the payment order. Inside the per-fee link, and possibly also outside it, the merchant provides the *offer description,* which is text and/or graphics describing the offer and presented to the consumer. We discuss some standardization efforts for the offer and per-fee link later on.

2. Typically, the browser displays the offer description. The customer invokes the payment process, typically by pressing the per-fee link. This results in passing the per-fee link parameters to the PSP, providing it with the payment offer details.

3. At this point, in most cases, the consumer has already been presented with the payment details in the description sent with the per-fee-link. However, a malicious merchant could have provided a description that differs from the payment details sent to the PSP. Therefore, the PSP has to validate that the consumer has actually approved the payment details. This is usually done by sending a payment approval request, e.g., as a web page. The user approves (or declines) this request. The approval process should be authenticated to ensure secure authorization.

4. Once the PSP has validated that the consumer properly approved the payment, then the PSP issues an authorized payment order (PO) sent to the consumer. The PO is often authorized by being signed by the PSP or

authenticated using a key shared between the PSP and the merchant. If the payment order as sent to the consumer might be "stolen" by an attacker and used to obtain goods and/or services for the attacker, this communication may be encrypted, e.g., using SSL.

5. The consumer's browser usually automatically processes the authorized PO by sending it as a request to the merchant. When a dedicated payment wallet is used by the consumer, it may modify the PO before sending it to the merchant, e.g., adding the consumer's signature or otherwise "validating" the payment order.

6. The payment order should be validated by the merchant, checking that it was properly authorized by the PSP. This may be done using a MAC key shared between the PSP and the merchant, or by the PSP's public key signature. Once the payment order has been validated, the merchant should provide the ordered goods or services to the consumer.

7. In an offline, "batch" process, the merchant deposits the payment orders and receives the funds.

Alternatively, and in particular, when the customer does not communicate directly with the PSP, but only via the merchant, then the customer can authenticate the payment approval by appending a message authenticator to it. The authenticator may be message authentication code (MAC), using a secret key shared between the PSP and the customer, or a digital signature, using the customer's private signature key and validated using the customer's public key. This may require dedicated payment software (*wallet*) in the customer's computer. We discuss this issue in Section 12.5.

Non-repudiation for Payment Approval

The solutions discussed so far provide different levels of security for the payment approval from the consumer. However, they do not completely prevent fraud; hackers may guess, steal, or otherwise expose passwords and keys, email may be misrouted, hackers may expose cookies in transit, and other users of the same computer may expose keys and passwords. In particular, all of the mechanisms above are vulnerable to a software virus in the consumer's computer or device (although many existing mobile devices may not be susceptible to viruses, due to limited functionality). Furthermore, the process depends completely on the trustworthiness of the PSP; a corrupted PSP (possibly hacked by an employee or third party) could claim that transactions were authorized, and in particular, compute any authenticators and authentication keys, as all such keys (if used at all) are also known to the PSP's computer. Definitely, therefore, non-repudiation is not achieved. Hence, consumers may still dispute their transactions, claiming unauthorized use, rightfully or possibly to avoid payment (without justification). In addition, of course, customer may dispute a transaction claiming dissatisfaction with the service or merchandise, as discussed earlier; but we focus on disputes claiming unauthorized use.

Some micropayment systems, e.g., First Virtual [12.36], solved the remaining disputes problem simply: they automatically refunded each dispute, and in this case did not pass the funds to the merchant. In fact, to completely protect First Virtual, they simply did not pay the merchants until the dispute period expired, thereby making additional profit from the float as well as avoiding the dependency on the merchant to actually pay them back. Considering the credit card experience showing the disputes are largely due to problematic merchants and merchant practices, there is some justification to this policy. Yet, it is clearly open to abuse by consumers, and raises a substantial business risk for merchants. Indeed, one might suspect that if a payment product adopting such a fully automated refund will become widely popular, then cheating may become commonplace. We therefore focus on systems that do not automatically accept all disputes. As noted above, we discuss only disputes claiming unauthorized transactions.

Many micropayment systems take the opposite approach, and simply forbid any disputes claiming unauthorized use (and therefore, usually, any form of dispute with the PSP). Often this is done simply by requiring the customer to agree to accept the record of transactions kept by the PSP. However, in many cases this mechanism may be unacceptable and possibly even illegal, especially when consumers are not fully protected against unauthorized charges. It appears that in order to completely disallow disputes, it is highly desirable that the PSP is able to prove to a third party that the consumer actually authorized each payment. This requires *non-repudiation.*

Non-repudiation may be achieved by using digital signatures, such as RSA to sign the payment approval from the consumer computer or device to the PSP. The customer must also agree in advance that payment orders digitally signed using their private key are to be considered as signed and authorized directly by the customer. The private signing key will be installed by the customer in their signing software, service and/or device.

There is, unfortunately, one serious pragmatic problem with digitally signing payment orders by the customer's computer or device: digital signing is *not* a standard, easy-to-use feature of widely-deployed[3] operating systems, browsers, or mobile devices. This should be compared with the solutions in the previous section, which do not offer non-repudiation, but on the other hand, do not require any new software in the consumer's computer or mobile device, or any complex operation for a consumer wishing to use them.

[3] Digital signing functionality is available in some versions of the Netscape browser, e.g. 4.04, as well as in some mobile devices.

An obvious solution is to require the consumer to install digital-signing software. Indeed, digital-signing functionality is often one of the main roles of a *wallet* utility installed on the consumer's machine. Wallets can perform other useful functions, such as payment management and logging, but their deployment is difficult and expensive. We discuss this important issue, and some possible solutions, in Section 12.5.

Another concern is that digital signing technology is relatively computationally intensive. A large amount of work on micropayment systems is focused on providing non-repudiation while minimizing or avoiding completely the use of public-key operations; see Section 12.6.

Irreversible Transactions and "Bearer Certificates"

We now discuss the third approach for preventing disputes, which takes a somewhat radical view: Design the micropayment system in such way that it will be technically infeasible to reverse payments, rather than relying on the consumer's agreement that properly authorized (or digitally signed) payments cannot be disputed. In this approach, the micropayment system operates in such a way that the PSP would not maintain track of individual payment transactions and would therefore be *unable to reverse transactions.* This kind of payment order is often referred to as *bearer certificates*, since the payment is done by the customer passing a message to the merchant, who deposits it at the PSP, but this bearer-certificate payment-order message does not indicate the identity of the consumer [12.12, 12.1]. More specifically, a bearer-certificate payment involves two separate phases:

1. Customer "buys" bearer-certificates from PSP (payment approval or withdrawal phase).
2. Customer pays merchant by providing the bearer certificate (payment authorization phase).

The PSP does not keep records of the identity of the customer who received each bearer certificate. In some proposals, the bearer certificates are provided during withdrawal, in a "blinded" manner, that does not allow the PSP to identify which bearer certificate was sent at which withdrawal. For more details on such blinded withdrawal and digital cash, see Chapter 8. In other systems, weaker anonymity is used, and the PSP simply does not maintain the records linking from the bearer certificate to a particular customer.

Bearer certificate systems should ensure that an attacker cannot "steal" the bearer certificate in transit from the PSP to the customer, from the consumer to the merchant, or from the merchant to the PSP. When the consumer is using standard, available browser software, and the bearer certificate is forwarded to the merchant exactly as sent from the PSP, the communication may be protected by using browser-provided encryption (usually SSL/TLS). In some cases, this may be avoided by requesting a bearer certificate specific to the requirements of this particular consumer. However, this may conflict with the requirement that bearer certificates cannot be linked to a particular consumer and purchase (to make it impossible to dispute transactions).

When the consumer uses dedicated wallet software, then the wallet may "activate" the bearer certificate it receives from the PSP. A bearer certificate that the wallet did not activate is not considered valid. Therefore, the attacker will not gain anything from a copy of the bearer certificate in transit from the PSP to the consumer. This activation may be as simple as attaching a random number x to the bearer certificate, where on payment approval the consumer provided to the PSP with the result of a cryptographic, a one-way hash function $h(x)$ is applied to x, and the bearer certificate is linked to $h(x)$, e.g., by including the PSP's signature on $h(x)$.

In any case, using a "bearer certificate" it should be impossible to link back from the payment order to the identity of the customer or to the withdrawal transaction in which the customer bought the bearer certificate. Therefore, the PSP simply has no way to reverse a payment done using the bearer certificate. Therefore, disputes are technically impossible. Proponents of this approach [12.12, 12.1] argue that bearer certificates are the only way to avoid disputes, since consumer-protection laws may overrule any limitations on disputes included in the agreement between the consumer and the PSP. On the other hand, using a bearer-certificate, the consumer does not receive a receipt of having paid from the payment system. This could be a substantial disadvantage for some applications.

Even if bearer certificates are used, it seems that for the goal of avoiding disputes, it may be acceptable for the PSP to know the linkage between the customer and the bearer certificate at the time of the withdrawal. This seems acceptable, as long as the PSP always erases this information later and does not provide a proof of it to the consumer. Clearly, such a solution is much simpler than the techniques used for digital cash.

12.5 Customer Acquiring and Support Costs

Businesses often spend large amounts to acquire new customers and to retain and support existing customers. Micropayments services have very small revenues per transaction and per customer. Therefore, it is especially critical to minimize the average amortized costs of customer acquiring and support efforts. In the following sections, we discuss three techniques for minimizing costs and maximizing the number of payment transactions, together ensuring low amortized costs per transaction.

1. Enable a simple-to-use, intuitive "click and pay" mechanism for payment approval. This has the dual impact of encouraging usage (more payments) and reducing support and acquiring costs (easier to use, i.e., fewer questions, easier to sell).

2. Support customers without requiring installation of a PSP-provided and supported local wallet application. This can reduce the substantial costs of providing wallet applications and supporting them (for multiple platforms), as well as the costs of convincing customers to install the local wallets.

3. Allow interoperability among the PSPs, namely, a customer of one PSP can pay a merchant of another PSP (as shown in Fig. 12.4).

12.5.1 Click and Pay Using Per-Fee Links

Micropayment systems are designed for low-value transactions. As such, it is important that the user interaction will be as natural, convenient, and quick as possible – ideally, click and pay. In fact, if the customer is not willing to pay much for a product or service, they may also not be willing to waste a lot of time and energy in the payment process. An easy, convenient and fast click-and-pay process also reduces costs for customer acquiring and support. The click-and-pay payment process becomes a natural extension of the familiar web surfing interface: To buy information or service, the user just clicks on *per-fee link*, much like clicking on a normal hyperlink.

We must ensure that the user pays only intentionally, i.e., the seller cannot trick the user into pressing a per-fee link without the user being willing to pay the price charged. For example, the IBM Micro Payments system [12.17, 12.11] provides per-fee-links which appear very similar to regular hyperlinks, by adding cues for payment. Specifically, when the cursor is over the per-fee link, the shape of the cursor changes to either a dollar sign (if the amount is over a user set threshold) or a cent sign (if the amount is under the threshold). Furthermore, the exact price is indicated in message/status area of the browser (where it normally writes the URL of the hyperlink); and the customer can specify a maximal amount for "click and pay", such that over this a pop-up box will require confirmation.

There are multiple ways for implementing per-fee links. The Web consortium developed a proposal [12.24] for per-fee-link syntax (this proposal was suspended since there were only few implementations). The proposal supported several ways for specifying and displaying per-fee-links, as <Embed>, <Applet> or <Object> elements; all of these require extensions to the standard browsers to display the per-fee-link, such as a plug-in, ActiveX control, or an applet. This requires installation of "wallet" software on the customer's machine.

It is sometimes also possible to provide per-fee link using only a standard browser to display the price simply as added text to the link. Consider Fig. 12.2, where the communication between the customer and the merchant flows through the PSP or a gateway associated with the PSP. In such scenarios, the PSP may modify the per-fee-link on its way from the merchant to the customer, and indicate the purchase details information (in particular the price) by adding the description of the payment details as textual and/or graphical hypertext link as part of the hypertext content sent to the consumer. For example, when using HTML:

<A HREF="*url*">click here to receive the song [5cents charge]</A>

The price information ([5cents charge]) was inserted or validated by the payment gateway (so that the displayed amount will be identical to the charged amount). The consumer indicates agreement by simply selecting (clicking on) this link. The transformation of the page sent from the merchant is especially natural in mobile scenarios, where the mobile gateway often creates the encoding (HTML or otherwise) appropriate to the consumer's device display capabilities.

Some caution is necessary, however, when using a regular hypertext link with the price added to the textual message as shown above, to prevent a merchant from invoking payment from a seemingly free (or less expensive) link. Namely, the PSP must accept as approval only requests that result from the consumer following the link it modified; the risk is that the merchant will insert in a page sent to the consumer a link invoking the payment procedure in the PSP but different (seemingly free) text.

To avoid this threat, the PSP may filter the hypertext to remove any fraudulent per-fee links inserted by the merchant. Another technique to prevent such fraud is to customize the per-fee links, e.g., by encoding in the hypertext link (the HREF attribute) an authenticator such as $MAC_k(price, description, clientID, time, requestID)$. The key k used in calculating the authenticator is known only to the PSP. The *price* and *description* fields ensure that the purchase conforms to what the customer approved. The *clientID*, *time* and *requestID* fields prevent the merchant from copying the authenticator field from a previous request (where *time* is used for a stateless server and *requestID* for a server who can remember states).

In addition, the gateway should validate that the page, as sent to the consumer's computer or device, does not contain script that may modify the page presented to the consumer (e.g., changing the description presented to the consumer). See [12.14] for techniques to validate that a web page is not modified by a script contained in it.

12.5.2 Local Wallets and Server Wallets

Acquiring customers, and helping them when they encounter difficulties, is difficult and expensive. It is difficult to convince customers to sign up with the system, open an account, and do any additional operations such as installation of a wallet (if necessary). Acquiring customers may therefore require substantial investment in the form of advertising and incentives, and even then, it usually takes substantial time to convince a small fraction of the potential customers to try the system.

When customers use the system, the costs of answering customer inquiries, which could be technical, financial, or administrative, can be significant. Customers are usually expecting financial service providers, including micropayment PSPs, to provide highly available and free customer support. The average cost per customer call is substantial and far exceeds the typical cost of micropayment transactions, not to mention the fees. These costs may exceed the thin profit margin per transaction, especially of a micropayment service provider.

Customer acquiring and support costs depend significantly on whether the customer has to install and use dedicated *local wallet* software to authorize micropayments, or whether the customer can use pre-installed, general-purpose mechanisms, such as the browser and/or e-mail utilities, with the payment functionality provided by a *wallet server* in the network. Dedicated, local-wallet software, running on the customer's computer or device, can provide better user interface and functionality, e.g., provide transaction log. Most importantly, dedicated wallet software can perform public-key digital signatures for payment orders, providing non-repudiation of the fact that that customer authorized the payment, and possibly avoiding dispute resolution and chargebacks.

However, it is important to realize that a malicious local user, or program, can often bypass local wallet software. In particular, when running insecure operating systems such as Windows, any malicious program (e.g., a virus) or any other user of the computer will be able to perform unauthorized signatures. Therefore, even a local, dedicated wallet cannot completely prevent unapproved payments.

Furthermore, it is substantially harder to motivate customers to download and install dedicated wallet software, compared to opening an account using purely pre-installed, general purpose mechanisms (e.g., a browser). Furthermore, dedicated, locally installed wallet software requires substantially more support costs, e.g., to resolve installation issues, and to support different operating systems. As a result, wallet software results in high customer acquiring and support costs, and a lower adoption rate. Another problem for local wallets is that consumers may use multiple devices and computers, e.g., at home, office, and while mobile, resulting in coordination problems between multiple wallet installations of the same consumer. Indeed, as shown in [12.19], there have been only failures so far in efforts to introduce wallet software, for micropayment as well as for credit card and other payments, and essentially all existing deployments support a wallet server.

However, a server wallet operated by the PSP has some significant drawbacks as well, in particular, in forcing the consumer to completely trust the PSP, which may necessitate dispute resolution with its associated high costs. In order to support micropayments without allowing disputes, the PSP may find it necessary to offer the customer control over the payment authorization, possibly as an option.

The Future: Multiple and Third-Party Wallets

We expect that in the future, PSPs will offer their customers to either use the PSP's server wallet, in which case they agree to trust it completely, or to use local-wallet or third-party operated server wallet that provides digitally-signed payment orders to the PSP, in which case non-repudiation is achieved by the public key signature. The PSP would offer local wallet software that signs the payment orders. Alternatively, the PSP may allow the customer to use third-party provided wallets to sign the payment orders, where the third-party wallets may be local or server-based. The third parties will offer the wallets as a productivity aid and often as part of a larger money-management or personal information management software or service, e.g., as part of the services offered by a portal. If some of the operations are especially sensitive, then the PSP may even require approval by more than one of the wallets of the same customer, providing additional protection against unauthorized use (but this additional complexity is probably not necessary for micropayments). By offering customers the option of using local wallets and third-party operated wallets, the PSP should be able to avoid dispute resolution process entirely, and therefore maintain low operational costs.

In Fig. 12.7, we show such an architecture, where the PSP allows the customer to use the PSP's wallet server, a local wallet provided by the PSP, or a third-party-provided wallet server or local wallet. The customer may authorize payments from their own computer or another computer, or from any of a variety of devices (different wallet servers may support different devices). This architecture requires protocols for managing an account and authorizing transactions from multiple wallets (software agents of the consumer), ensuring that the entire log of transactions is available to one or more wallets as needed. For such protocols also supporting record aggregation, see [12.13].

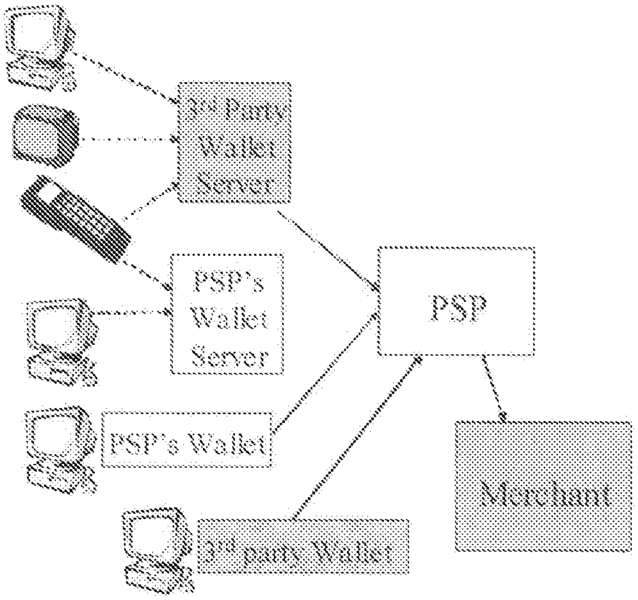


Fig. 12.7 Multiple wallets, devices, and computers for the same account

To invoke a wallet server, the merchant usually includes a link to it in the web page, with text describing the offer. It is possible for a single per-fee link to invoke one of multiple wallet servers as needed; this is achieved by the merchant including a link to a special "PSP directory" site. The PSP directory can try to detect automatically the identity of the customer's PSP, e.g., using a cookie stored in the browser and sent automatically to the PSP directory (see Fig. 12.8).

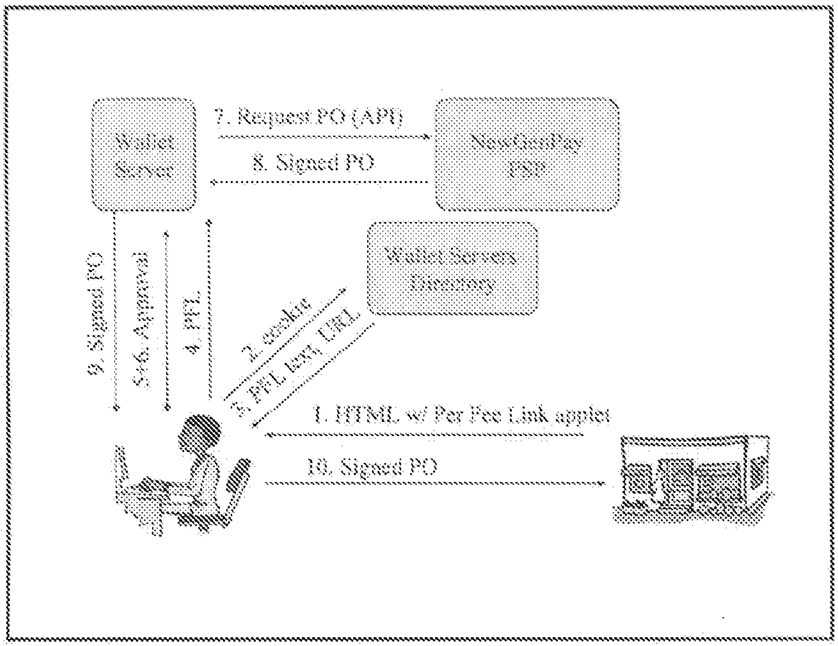


Fig. 12.8 Automated wallet server selection from directory

12.5.3 Building Critical Mass and Acceptability by Interoperability Among PSPs

For a micropayment system to be profitable, it is critical to have a large number of transactions, customers, and merchants. This is required both to reduce the amortized cost of fixed expenses (such as software development), and to make it easier to acquire customers and merchants (since customers and merchants prefer systems where they can transact with most merchants and customers, respectively).

Many micropayment service providers approach this problem by simply planning (or hoping) to become the dominant micropayment PSP, providing services to most customers and merchants. However, in an open, competitive market, it will probably require a very large investment in order to become the dominant PSP, and there is a substantial risk of failure to meet this goal.

An alternative approach is for multiple PSPs to cooperate, allowing customers of each PSP to pay merchants of each of the PSPs. This approach of cooperation between competitors is well established in many other areas of the modern, global economy. Indeed, there are several large networks connecting competitors and allowing them to interoperate, particularly, in payments, e.g., the credit card networks, clearing networks between banks, ATM networks, and others. It is therefore reasonable to expect a similar global network will handle micropayments.

It is difficult and expensive to establish trust among competitors, and even more so among many competitors. However, trust is required for allowing an interoperable payment network, where a merchant can receive a payment authorized by different PSPs. Existing, deployed payment networks handle this problem in one of two ways:

- **Centralized solution:** The payment network is "owned" by a single entity, usually called a *brand*. The brand sets up rules of operation for all the participating PSPs, possibly including technical means, such as communication protocols, and audits their operation. The recipient of payment trusts only the brand (and possibly their PSP). The major credit card systems, e.g., Visa and MasterCard, operate in this way. This solution could be appropriate for micropayments as well, if a very small number (say under five) of dominant micropayment brands would become dominant. However, substantial investment is necessary to establish a dominant brand. Such investment may not be economical for establishing a brand for micropayments where the expected fees are low.

- **Offline clearing solution:** The other approach minimizes any assumptions about global trust or organization. This is the approach normally used to deposit checks from remote, unknown banks. Namely, funds are available to the depositor (e.g., the merchant) only after the depositor's bank has received the funds from the payer's bank. This solution has two problems that seem to make it unsuitable for electronic payments (and in particular micropayments):

    - The merchant must wait very long to know if the payment is valid.

    - A corrupted intermediary bank (PSP) may deposit the check, receive the funds, but keep the funds to itself while claiming to the depositor that the payment was rejected. With physical checks, the merchant usually receives back the check in case of failure to prevent this attack (and for other remedies).

We conclude that, to provide interoperability among PSPs, it is desirable to ensure security without requiring global trust in all PSPs, as such, global trust will require central management and ownership (which may not exist).

Open, Decentralized Payment Network

We now describe the design of open, decentralized payment network, as presented in [12.17, 12.11, 12.16]. This design is based on two principles:

- **Minimal trust requirements – only between PSP and its account owners:** All account-based payment systems require trust between the party maintaining the account (e.g., bank or PSP) and the account owner (customer, merchant, or another PSP). This trust allows the account owner to deposit money and received payment orders in the account, trusting that the PSP will properly credit the account and that the funds will be available to the account owner and used only according to the account owner's instructions. Similarly, the PSP may trust the account holder to provide funds to cover any debt due to credit payments from the account. Such direct trust between the PSP and the owners of the accounts kept by the PSP is unavoidable, on the other hand, it is relatively easy to manage, as it involves only two parties with long-term relationship. However, no other trust relationships should be required. By avoiding any `global` trust requirements we make it easier and cheaper to manage risks and avoid fraud, thereby reducing costs.
- **Automated dispute resolution between PSP and account owners:** By specifying the exact terms for any transaction related to the account, we can completely automate the resolution of any disagreement between them.

The main mechanisms for achieving these goals are two digitally signed messages: the *payment order* and the *payment routing table* (PRT). Consider the simple scenario with two PSPs, A and B, illustrated in Fig. 12.9. In this scenario, the customer C accepts an offer (flow number 3) from the merchant M, and pays for it by sending to the merchant a payment order signed by PSP_A (flow number 6). The merchant can immediately validate the payment order. The merchant can then deposit the payment order, as in flows 7 and 8. Deposit can be immediate or delayed; the merchant (and intermediate PSP_B) may wait some time for possible batching with other deposits for efficiency. The immediate, local validation, without online communication by the merchant, is possible using information that the merchant received from PSP_B in the payment routing table signed by PSP_B, in an offline process before the purchase, in flow 2.

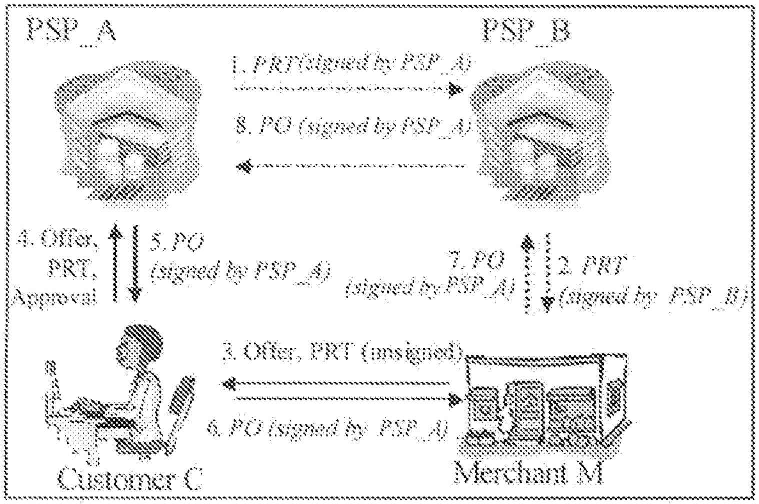


Fig. 12.9 Interoperability between two PSPs

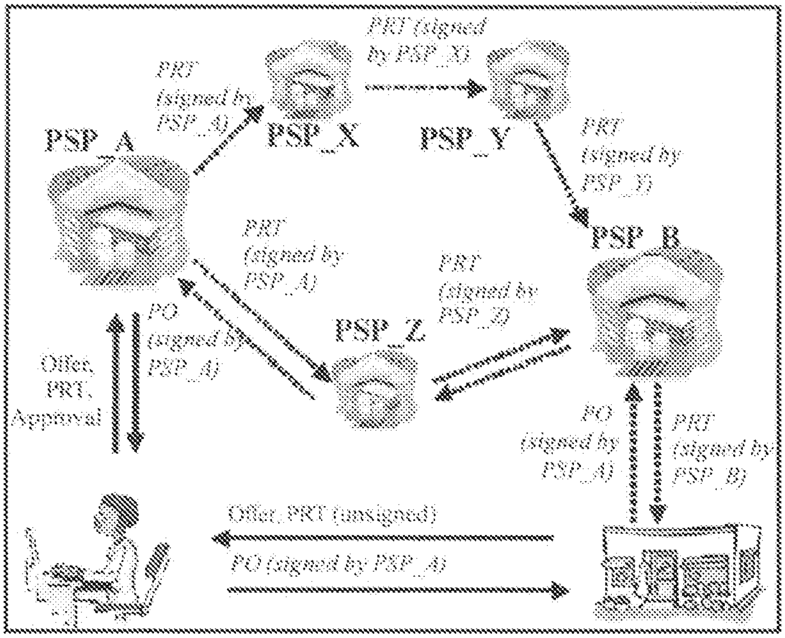


Fig. 12.10 Interoperable PSP network (example)

The payment routing table is sent in advance, e.g., daily, from each PSP to all of the entities keeping an account with this PSP and which may receive payment orders for deposit. In Fig. 12.9, PSP_A sends (e.g.. daily) PRT to PSP_B, and PSP_B sends PRT to the merchant. For a more complex scenario, consider Fig. 12.10, where we show five PSPs. Normally, each PSP will send a PRT message periodically to all of its merchants and to all of the PSPs keeping an account with it (namely, to every entity that may deposit a payment in this PSP). For efficiency, a single PRT message may contain payment routing entries for multiple issuing PSPs (i.e., for payment orders issued by different PSPs).

The PRT contains all the information necessary to allow the merchant to process a payment order signed by PSP_A. The PSP sending the PRT will always sign it with its private key (e.g., PSP_B). Payment service providers construct their outgoing PRT messages, based on the incoming messages, and on local policies (e.g., for limiting the damage if a particular PSP does not keep its commitments, or for maximizing revenues via adjustment of fees). The PRT, signed by PSP_B, will include the following details for every PSP from which PSP_B agrees to receive payment orders:

1. The identity of the account holder (merchant M).

2. The public key of the PSP issuing the payment order (e.g., PSP_A).

3. The maximal total amount of payments from this PSP allowed (until the next PRT).

4. The fees applied to deposits of payment orders from this PSP (e.g. maximum between 5 cents and 2%). The merchant receives the amount in the payment order minus these fees (no additional, "hidden" fees).

5. Minimal deposit time, in which deposits will be honored only if made before this time.

6. Acceptable proof of transmission identifies what is a sufficient proof of transmission of a message and of the time of transmission. This proof allows automated resolution of disputes on whether payments were deposited in time or expired. Typically, the proof will be either the transmission log of the PSP, or a signed receipt from one or more trusted third parties store-and-forward servers. The store-and-forward servers allow the depositor (merchant or PSP) and the PSP to resolve disputes due to failure of communication between them; as long as the total pending payments amounts are not very large, the PSP log files may be acceptable.

7. The path of PSPs through which the payment orders will be deposited. In the scenario in Fig. 12.9, this will only include the identity of PSP_B. In the scenario in Fig. 12.10, this may include the path {PSP_B, PSP_Y, PSP_X} or the path {PSP_B, PSP_Z}.

8. The validity period during which the PSP issuing this PRT (e.g., PSP_B) is committed to the terms specified in it. Later deposits may not be honored, unless covered by a new PRT.

9. A unique identifier for this PRT (e.g., counter).

10. Few other technical details may be added to ensure that there are no disagreements, see [12.16].

The Payment Order (PO) contains the precise details of the payment, including:

1. The amount to be paid (and currency).

2. The path of PSP's through which this PO is to be deposited.

3. The expiration time (the issuer will not pay for the PO after this time – it will become void).

4. The identifier of the PRT that this payment is applied to.

5. The issuing time of this PSP and a serial number of PO issued at this time (for detecting replays).

6. Possibly, additional conditions on the validity of this PO, such as a hash *h(x)* sent by the customer, such that the payment order is valid only if the customer attaches to it the pre-image *x*.

The merchant also sends a (simplified) version of the PRT to the customer, and the customer may send it to her PSP (PSP_A). This allows the customer to choose the best route (and possibly currency) to use in the payment order.

12.6 Equipment, Processing, and Communication Costs

In this section, we discuss schemes to reduce the cost of the equipment necessary for the micropayment system, and in particular the cost due to the processing and communication requirements. Computational and communication requirements are reasonably well defined, and minimizing them is an obvious goal, easily measured, and similar to problems of minimizing communication and processing complexities in many other algorithms. Therefore, it is not surprising that much (or most) of the academic research on micropayments has focused on minimization of processing and communication costs. However, while these techniques are of algorithmic interest, we caution that the actual impact on cost and performance may be insignificant in many cases, as processing and communication costs are often a negligible compared to the other expenses (and delays) in the system.

Most of the efforts in reducing complexities and requirements focused on one of the following areas:

1. *Avoiding or reducing the use of public key cryptography* (digital signatures), since it is relatively computationally intensive.

2. *Allowing offline (or semi-offline) payments*, where the PSP is not involved during the process of payment (or most payments, respectively). An important form of offline payments are *stored-value payments.*

We now discuss offline (and semi-offline) payments and then techniques to avoid or reduce the use of public key cryptographic operations.

12.6.1 Reducing Communication Costs – Offline (and Semi-Offline) Payments

As discussed in Section 12.4, most micropayment systems require secure authorization of every transaction by the PSP, to assure the merchant that the payment is final and that the PSP will not refuse to pay, claiming that the customer spent more than the maximal amount allowed (overspent). Usually, this requires the PSP to authorize each payment online. In many cases, the online authorization is easy and inexpensive, as the PSP controls a gateway on the communication path between the consumer and the merchant, as in Fig. 12.2. However, in other cases, the PSP is not on the communication path; in order to involve it, the merchant must contact it (as in Fig. 12.5) or the customer must contact it (as in Fig. 12.3). This requires additional appropriate communication capabilities and capacities by the PSP and merchant or customer, in particular:

- The PSP must have sufficient capacity to handle the maximal number of concurrent payment authorization requests expected at peak time.

- Either merchant or buyer (or both) must have communication capabilities to the PSP online (during payment process). This is usually the case when paying using communication devices, such as mobile phones or computers connected to the Internet. In fact, in many applications, it is feasible and cheap to add communication capabilities to either the payer or payee (e.g., add a GSM module to a vending machine). However, there are scenarios where both payer and payee are disconnected and where it is not reasonable to require one of them to add communication capabilities, e.g., payments between two handheld devices.

In this section, we look at mechanisms for avoiding or reducing the online authorization requirements from the PSP, thereby eliminating or reducing the costs of the communication (messages) and of the necessary communication capabilities.

Micropayment schemes are called *offline* if the PSP is *never* involved during the payment process and *semi-offline* if the PSP is involved only in few payment transactions. Both offline and semi-offline payment schemes may reduce the load on the PSP, and in particular avoid bottlenecks at peak hours. Offline payments may further allow applications where payments are between a (low-cost, mobile) device carried by the customer and a (low-cost, fixed or mobile) point-of-sale device, without any network connectivity to the PSP.

The main approaches to providing offline and semi-offline payments are:

- **No authorization:** A trivial way to avoid online authorization is not to require any authorization by the PSP at all. The risk of overspending is borne by the PSP or the merchant, depending on the agreement between them. However, this requires the PSP or merchant to cover the cost of overspending by customers, which may be an unacceptable risk and expense.

- **Random or threshold authorization:** To reduce the number of authorization requests from the PSP, while limiting the risk to the merchant of a payment being cancelled due to overspending, the merchant can decide on whether online authorization by the PSP is required for each transaction. When using randomized or threshold authorization, either the merchant or the consumer wallet can request the authorization; requesting by the merchant is more direct as the merchant decides on the need for online authorization, but requesting indirectly by the consumer's wallet allows a simple, efficient client/server design for the merchant. In early versions of IBM Micro Payments [12.11, 12.17] we used threshold authorization, where the merchant requests payment authorization from the PSP when reaching a threshold amount (of that particular transaction or of all transactions pending authorization by that buyer). However, we later changed to online authorization, since we found that the its overhead is negligible, that merchant servers may spend considerable resources to keep track of total purchasing per customer, and that merchants are alarmed and confused by the possibility of unapproved payments being cancelled later. We also considered using random authorization, as proposed (independently) in Agora [12.9], where the merchant requests authorization randomly and the PSP identifies and blacklists any overspending customers. However, maintaining and distributing the blacklists may become a bottleneck and open the system to denial-of-service attacks, while on the other hand the overspending is still possible (until detected and until blacklist is updated). This, combined with the simplicity of opening multiple micropayments accounts, makes this solution impractical.

- **Pre-authorization:** The PSP may authorize the customer payments up to a pre-defined limit to a *specific* merchant. Overspending is then limited to multiple payments to that specific merchant, which the merchant can easily detect (e.g., using sequence number). When the PSP pre-authorizes payments (up to some limit, for a given merchant), it actually delegates it authority to the customer, who provides the final payment authorization directly to the merchant or point-of-sale (e.g., vending machine). The customer may digitally sign each payment authorization, or use one of the techniques described below to authorize the payments with reduced computational requirements. Pre-authorization is a semi-offline technique, as the customer needs to request pre-authorization for each specific merchant.

- **Stored Value Payments:** Taking one-step further than pre-authorization, the PSP can avoid online authorization completely by delegating its authority to authorize payments to a *tamper-resistant module trusted by the PSP,* e.g., a smartcard provided by the PSP. The module keeps track of the spending by the consumer, and authorizes payments only as long as the customer does not overspend. In a sense, the funds (value) which the customer can legitimately spend are stored in the device; hence, the name *stored value.* Stored value solutions depend on the temper-resistance of the module, to prevent duplication of money. Tamper resistant modules seem to require hardware (there are some efforts to create tamper-proof software, by *obfuscation*, but recent negative results seem to indicate that this is difficult or impossible). This introduces significant installation costs. Even tamper-resistant hardware is often subject to attacks. Therefore, stored-value protocols should limit the damage due to exposure of the keys of a limited number of modules. To limit the damage, most stored-value systems use a different key for each module and blacklist over-spending modules (but this requires identification of over-spending modules and informing all merchants, a non-trivial undertaking). Some authors claim that stored-value payment devices have the advantage of a total limit to the value lost if the card/device is stolen and abused. However, such a limit is easy to achieve with an account-based solution, simply by setting a limit to the amounts that the customer can spend using a given key.

12.6.2 Reducing Computational Complexity due to Public Key Operations

As discussed in Section 12.4, non-repudiation of payment approval is highly desirable, as it can help to reduce disputes and detect consumer fraud. The main cryptographic tool for achieving non-repudiation is public key digital signature, typically using the RSA or DSA algorithms [12.30, 12.5]. However, most public-key cryptographic mechanisms, and in particular digital signatures, are computationally intensive operations, compared to hash functions and shared-key cryptographic mechanisms. The ratios in the processing times depend, of course, on specific functions and implementations, but ratios of 100 and even substantially more are quite common. Much of the research on micropayment systems focused on reducing this computational burden by designing micropayment protocols and systems that avoid the use of public key cryptography, use only a very small number of public-key operations, or use more special, efficient public-key cryptographic mechanisms. In this section, we review some of these techniques.

We comment, however, that while the goal of avoiding or minimizing the use of (computationally intensive) public-key operations is natural and interesting, the actual cost of their processing time may be negligible compared to other costs and overheads in a practical micropayment system. The computation of an RSA digital signature [12.30], on typical desktop machines, takes only very few milliseconds; validation usually takes even less (when a small public exponent is used). The DSA algorithm [12.5] is about as efficient (but with computation faster than validation). Hardware accelerators can further substantially reduce the overhead. Therefore, it seems that for most realistic micropayment applications, computation, and validation of a digital signature on the payment order is not a significant cost factor. We therefore believe that techniques to avoid or reduce the use of public key operations and, in particular, of digital signatures and their validation, should only be used in special circumstances, and only when they do not result in a more substantial increase in other expenses. For example, a situation where it may be important to avoid digital signatures by the buyer is when the buyer is using inexpensive mobile devices for payments, such as a key-chain gadget or smart card.

The main techniques for avoiding or minimizing the computational burden due to public-key cryptographic operations are:

- Use authentication mechanisms that do not provide non-repudiation, such as (shared-key) message authentication code (MAC). This is appropriate between two parties with a long-term relationship, such as customer and their PSP or merchant and their PSP, and as long as the total amounts are not too high. However, this is not recommended between two parties with a sporadic, ad-hoc relationship, such as customer and merchant, or when the total amounts become larger than the value of the relationship. The possible savings in computation time may become smaller than the added risks and operational costs due to dispute resolution and customer support. Proposals for micropayment systems using MAC instead of digital signatures include NetBill [12.4] and MilliCent [12.23, 12.8].

- Use public-key signature algorithm that is substantially more efficient than [12.30] or [12.5]. There were several proposals of significantly more efficient public-key signature schemes, e.g., [12.31]. However, none of these schemes has yet gained sufficient adoption, and the amount of cryptanalysis effort to break them are, so far, limited, therefore their use is not recommended for sensitive and high-value signatures or where it may lead to disputes.

- Use an online/offline signature scheme as proposed in [12.7]. With these schemes, the payment is signed (online) using a one-time (or limited-use) public key digital signature scheme, which is substantially more efficient than regular, unlimited use public key signature schemes such as [12.30] and [12.5]. The public-key of the one-time scheme is signed in advanced (offline), using a regular digital signature scheme. This reduces the number of computations required online (during the payment process). By using an appropriate limited-use scheme, it may also be possible to reduce the average computational load. These schemes are easily adopted for semi-offline payments where the PSP pre-authorizes the one-time signature scheme for a particular merchant and a specific maximal amount, and the consumer applies the one-time signature to authorize a specific amount (up to the maximal).

- Use one-way hash functions, which are much more efficient than public-key signatures. These techniques fall into the following two categories:

    - *Hash chains* and *Hash trees:* this technique uses digital signatures for non-repudiation, but only once for multiple purchases between the same customer and merchant. Often, all purchases must be of the same amount. The customer or PSP digitally signs one (pre-) authorized payment order for the merchant, which the customer uses to authorize multiple purchases. We can therefore use this technique for semi-offline pre-authorized payments, where the PSP pre-authorizes the payment and the customer provides the final authorization. The (pre-)authorized, signed payment order includes a value y, which is the result of repeatedly applying l times a one-way hash function h to randomly chosen seed x. Namely, $y=h^{(l)}(x)$. The signature also includes a monetary value per pre-image, say c. To pay ic (e.g., i cents, when c is declared to equal one cent), the customer sends the authorized payment order together with a value x_i such that $y=h^{(i)}(x_i)$, namely, y is the result of applying i times the one-way hash function h to the value x_i. As long as $i \leq l$ it is very easy for the customer to compute this since he knows x. Therefore, repeated payments of the same amounts to the same merchant require only few computations from consumer and seller. This scheme is useful when the consumer buys repeatedly from the same merchant (and usually for the same amounts). Proposals based on hash chains include PayWord [12.29], micro-iKP [12.15] and others. Some variants use the natural extension of the hash-chain idea into a *hash-tree*, e.g., [12.18], for improved performance and flexibility.

- *MicroMint* [12.29]: This scheme is unique in requiring the PSP to perform a "hard" cryptographic operation, i.e., an operation that is assumed to required huge computational resources, but is easy to verify. This is justified by performing many such operations together, which is substantially more efficient *per operation* than performing only a single "hard" operation or relatively few operations, as can be expected of an attacker. Rivest and Shamir [12.29] suggest as "hard" operation to find a *k-way collision* for a collision-resistant hash functions, namely, values $(x_1, x_2, .. x_k)$ such that $h(x_1)=h(x_2)= ... =h(x_k)=y$. It is easy to see that, indeed, if searching for collisions by exhaustive search, the overhead *per each k-way collision* is much smaller if a large number of collisions are collected together. The scheme has several variants, including identifying the buyer and possibly even the seller (e.g., by producing only strings where specific bits are the buyer/seller identity), non-repudiation (e.g., by signing a common prefix to all produced coins), and others.

- Probabilistic payments: This is another hybrid technique, where the merchant receives one signed payment order for a substantial maximal amount, but with additional messages defining the actual amount paid, thereby allowing the same pre-authorized payment to be used for many micropayments. However, in this case, the micropayments are not done by gradually increasing the value (as with hash trees), but by gradually raising the *probability* of payment of the maximal (total) amount. Therefore, each micropayment is done by increasing the expected value that the merchant will receive – but the amount that the merchant actually receives is always either zero or the maximal amount. See such techniques in [12.26 , 12.22].

12.7 Summary

When the buyer and seller are in physical proximity, it is easy to pay small amounts (micropayments) using cash. There is a substantial number of such transactions, however their total value and importance is quite limited. However, remote micropayments, over a network, are an important challenge. On the one hand, we cannot physically transfer cash (or other object); and alternative existing payment mechanisms involve substantial fees, impractical for small amounts. On the other hand, there are critical needs in e-commerce for micropayments, in particular to pay for information, evaluations and services.

There have been many efforts to enable micropayments, however none has succeeded yet. To understand what makes micropayments so difficult, we reviewed the cost factors involved in payment transactions. Most of the research on micropayments focused on reducing processing costs, and in particular, avoiding the use of (computationally intensive) public key operations; we discussed some of the techniques, but also noted that in reality, the processing costs of (reasonably-efficient) implementations are not one of the most significant cost factors, even when using public key signatures.

The two most important cost factors, in practice, are (1) disputes, charge-backs and their processing, and (2) customer acquiring and support. The best way to deal with disputed micropayments is by providing secure payment authorization, and refusing to reverse the (properly-authorized) micropayments. We discussed some of the challenges of this approach, including legal and marketing issues which are beyond the scope of this work.

Finally, we discussed ways to minimize customer acquiring and support costs. We argue that these costs could be minimized in one of two opposing approaches: (1) having one or two dominant providers of micropayment services, reducing competition and associated expenses while allowing relatively high profit margins, or (2) coopetition in a network of many interoperable micropayment service providers, using appropriate secure protocols. We presented the principles of an appropriate protocol for interoperability between competing micropayment providers.

12.8 References

[12.1] A. Back (1999) Bearer = anonymous = freedom to contract. Cryptography mailing list, Feb. 1999. http://www.privacy.nb.ca/cryptography/archives/cryptography/html/1999-02/0108.html.

[12.2] M. Bellare, et al. (2000) Design, implementation and deployment of the iKP secure electronic payment system. J Selected Areas in Commun 18 (4): 611–627.

[12.3] D. Chaum (1992) Achieving electronic privacy. Scientific American, August Issue, pp. 96–101.

[12.4] B. Cox, J. D. Tygar, M. Sirbu (1995) NetBill security and transaction protocol. In: Proceedings of the First USENIX Workshop on Electronic Commerce.

[12.5] National Institute of Standards and Technology (1994) Digital signature standard (DSS). FIPS PUB 186, US Department of Commerce.

[12.6] Escrow.com (2002) Escrow payments: process overview. http://www.escrow.com/solutions/escrow/process.asp.

[12.7] S. Even, O. Goldreich, S. Micali (1996) On-line/off-line digital signature. J Cryptol 9: 35–67.

[12.8] S. Glassman, M. Manasse, M. Abadi, P. Gauthier, P. Sobalvarro (1995) The millicent protocol for inexpensive electronic commerce. In: 4th WWW Conference Proceedings, O'Reilly, New York, pp. 603–618.

[12.9] E. Gabber, A. Silberschatz (1996) The Agora electronic commerce protocol. In: Proceedings of 2nd Usenix Conference on Electronic Commerce.

[12.10] P. M. Hallam-Baker (1995) Micro payment transfer protocol (MPTP). W3C Working Draft WD-mptp-951122.

[12.11] A. Herzberg (1998) Safeguarding digital library contents – charging for online content. Digital Library Magazine, January Issue.

[12.12] R. A. Hettinga (1998) A market model for digital bearer instrument underwriting (manuscript). http://www.philodox.com/modelpaper.html.

[12.13] A. Herzberg, I. Mantin (2002) Secure transactions with multiple agents (manuscript).

[12.14] A. Herzberg, D. Naor (1998) Surf'N'Sign: client signatures on Web documents. IBM Sys J 37(1): 61–71.

[12.15] R. Hauser, M. Steiner, M. Waidner (1996) Micro-payments based on iKP. IBM Research Report 2791.

[12.16] A. Herzberg, E. Shai, I. Zisser (2000) Decentralized electronic certified payment order (US patent application).

[12.17] A. Herzberg, H. Yochai (1997) Mini-pay: charging per click on the Web. In: Proceedings of the 6th WWW conference.

[12.18] C. Jutla, M. Yung (1996) Paytree: amortized signature for flexible micropayments. In: Proc. 2nd USENIX Workshop on Electronic Commerce, pp. 213–221.

[12.19] K. Böhle (2001) Access is king: about the bright future of server-based e-payment systems. ePSO Newsletter, No. 6. http://epso.jrc.es/newsletter.

[12.20] L. Loeb (1998) Secure electronic transactions: introduction and technical reference. Artech House, Boston London.

[12.21] O. T. Lee (2001) Trust and confidence with escrow payment service to DRIVE Internet/eCommerce transactions. CommerceNet Singapore (CNSG) eSecurity and ePayment Seminar. http://www.cnsg.com.sg/archive/eSecurity%20Stratech%20011017.pdf.

[12.22] R. J. Lipton, R. Ostrovsky (1998) Micro-payments via efficient coin-flipping. In: Proceedings of Second Financial Cryptography Conference, LNCS 1465. Springer, Berlin Heidelberg New York, pp. 72–82.

[12.23] M. S. Manasse (1995) The Millicent protocols for electronic commerce. In: First Usenix Workshop on Electronic Commerce.

[12.24] T. Michel (ed.) (2000) Common markup for micropayment per-fee links. W3C Working Draft. http://www.w3.org/TR/Micropayment-Markup/.

[12.25] M. Peirce, D. O'Mahony (1995) Scaleable, secure cash payment for WWW resources with the PayMe protocol set. In: Proceedings of the Fourth WWW Conference.

[12.26] R. L. Rivest (1998) Electronic lottery tickets as micropayments. In: R. Hirschfeld (ed.) Financial Cryptography: FC'97, LNCS 1318. Springer, Berlin Heidelberg New York, pp. 307–314. http://citeseer.nj.nec.com/rivest98electronic.html.

[12.27] E. Rescorla (2000) SSL and TLS: designing and building secure systems. Addison-Wesley, New York.

[12.28] T. Dierks, C. Allen: The TLS protocol: version 1.0. Network Working Group, Internet Engineering Task Force (IETF). http://www.ietf.org/rfc/rfc2246.txt.

[12.29] R. L. Rivest and A. Shamir (1996) PayWord and MicroMint--two simple micropayment schemes. In: 1996 RSA Security Conference Proceedings.

[12.30] RSA Laboratories. PKCS#1 – RSA cryptography standard. http://www.rsasecurity.com/rsalabs/pkcs/pkcs-1/.

[12.31] A. Shamir (1993) Efficient signature schemes based on birational permutations. In: D. R. Stinson (ed.) Proceedings of CRYPTO'93, LNCS 773. Springer, Berlin Heidelberg New York, pp. 1–12. http://citeseer.nj.nec.com/shamir93efficient.html.

[12.32] O. Steeley (2001) Guaranteed transactions: the quest for the "Holy Grail." ePSO Newsletter, No. 10. http://epso.jrc.es/newsletter

[12.33] MasterCard and VISA: Secure Electronic Transactions. http://www.setco.org/set.html.

[12.34] T. Schurer (2001) Largest German credit card issuer on massive reduction of charge backs. ePSO Newsletter, No. 10. http://epso.jrc.es/newsletter.

[12.35] L. Tang, S. Low (1996) Chrg-http: A tool for micropayments on the World Wide Web. In: Proceedings of the Sixth Usenix Security Symposium, pp. 123–129.

[12.36] D. New (1995) Internet information commerce: First Virtual approach. In: Proceedings of the First Usenix Workshop on Electronic Commerce.

13 Industrial E-Payment Systems and Solutions*

Zheng Huang [1], Dong Zheng [1], Zichen Li [2], and Weidong Kou [3]

[1] Shanghai Jiao Tong University
Shanghai, China

[2] Tsinghua University
Beijing, China

[3] University of Hong Kong
Pokfulam Road, Hong Kong

13.1 Introduction

As e-commerce over the Internet is taking off, online payment (or e-payment) has become an essential piece of the e-commerce puzzle. To support e-commerce, a variety of industrial e-payment systems and solutions have been developed and deployed in many countries. These e-payment systems and solutions enable transactions for people to trade goods or services for money. It is not our desire to cover the entire e-payment industry in a single chapter. Rather, we prefer to select a few e-payment solutions and introduce them to the readers as real-life e-payment examples, or, to some extent, as e-payment case studies. In this chapter, we select three e-payment solutions for discussion, including Visa Cash, iPIN, and PayPal. For each of them, we describe design goals, features, functions, and security mechanisms. In addition, in the appendices of the chapter, based on the available information, we selectively present the architecture of these payment systems.

13.2 Visa Cash

Visa Cash is the first e-payment system and solution that we have chosen for discussion [13.1]. The reason for selecting Visa Cash is that it is a global brand.

* Supported by the National Natural Science Foundation of China (Grant Nos. 90104032, 90104005, and 60173032).

Visa Cash is an electronic purse. It is a card that works like cash. A microchip is embedded in each card to store a specific amount of money and do some cryptography calculation. The Visa Cash system is a secure application module (SAM)-based system. With Visa Cash, one can pay for everyday necessities without having to carry around a pocket full of change. It is a fast, easy, convenient method of payment, which can be used for small purchases such as pay phones, cinema tickets, parking machines, or public transportation. Visa Cash can be used either in the real world or in cyberspace over the Internet.

Since its launch in 1995, Visa Cash has been widely used worldwide, including Argentina, Australia, Brazil, Canada, Columbia, Hong Kong, Ireland, Israel, Japan, Mexico, Norway, Puerto Rico, Russia, Spain, Taiwan, the UK, and the US.

13.2.1 Design Goals of Visa Cash

The design goals of Visa Cash are:

- Provide a payment product that is more convenient than cash, particularly in small-value transactions.
- Support a multi-currency capability and currency exchange.
- Supporting multi-applications on a single Visa Cash card.
- Offer similar convenience to cardholders wherever they are, traveling around the world or staying at home.
- Offer merchants a means to serve both domestic and traveling cardholders for the payments.
- Enable card issuers and acquirers to offer internationally acceptable electronic-purse services to cardholders and merchants.
- Work consistently within the existing financial environment.

13.2.2 Features of Visa Cash

Visa Cash has the following features:

- Easy to use:
  To use a Visa Cash card, one can simply insert it into the merchant's Visa Cash card reader. Visa Cash card's current balance will be displayed to let the cardholder know in advance how much Visa Cash they have to spend. The cashier will enter the amount of the transaction which is displayed to the cardholder. When the cardholder presses the "accept" button on the card reader, the amount of the purchase is automatically deducted from the Visa Cash card balance. The Visa Cash card's new card balance is then displayed.

- Anonymous:
  Compared with a Visa credit card, Visa Cash provides anonymity by not requiring the user's ID at the time of payment.

- Flexible card types:
  Visa Cash provides two types of cards, namely, disposable cards and reloadable cards. Disposable cards are loaded with a predefined value. They come in denominations of local currency, and make use of low-cost memory cards to store Visa Cash money. When the value stored in a card is used up, one can discard the card and purchase a new card. A reloadable card comes without a predefined value. This type of card can be loaded with value through specially configured devices, such as ATMs and other load devices. When the value is used up, the cardholder can reload the card. In terms of implementation, these two types of cards can be implemented based on either a proprietary or an open platform.

- Wireless supports:
  Visa Cash cards can be loaded through GSM networks.

13.2.3 Functions of Visa Cash

Visa Cash cards provide convenient and fast means for the cardholders and enable them to always have exact change. The cardholders can perform the following functions:

- Load: transfer money from a bank account to the Visa Cash card,
- Unload: the inversion of load,
- Currency exchange,
- Purchase,
- Purchase reversal,
- Incremental purchase,
- Cancel last purchase,
- Personalization: storing some personal information,
- On-line updates to the card application data.

Visa Cash can be used in many places, such as quick-serve restaurants, convenience stores, vending machines, gas stations, transportation, sundries stores, cinemas, newsstands, parking garages, grocery stores, department stores, taxis, parking meters, cafeterias, and video stores.

The most successful application of Visa Cash may be electronic tickets for the public transportation systems of Madrid and Barcelona, Spain. In these cities, Visa Cash is not only used in the public transportation system, but also used for car parking, public telephones, etc. Up to now, there have been at least 50 million Visa Cash cards issued in Spain.

13.2.4 Security of Visa Cash

The security of Visa Cash is based on the tamper-resistant property of smart cards. The security requirement is mainly authentication between each party in a Visa Cash system. In offline transactions, the smart card and merchant's terminal must perform mutual authentication using asymmetric cryptography. The authentication is a type of dynamic authentication (challenge/response).

In order to authenticate a smart card, it must be loaded with the following keys:

- Symmetric MAC (message authentication code) key,
- Card secret key,
- Card public key certified by the issuer with their private key,
- Issuer public key certified by a certification authority with their private key,
- Certification authority public key.

In order to authenticate the merchant's terminal, the terminal must be loaded with the following keys:

- PSAM (purchase secure application module) secret key,
- PSAM message authentication codes (MAC keys),
- PSAM public key certified by the acquirer with its private key,
- Acquirer public key certified by a certification authority (the same certification authority as that of the card) with its private key,
- Certification authority public key.

These keys stored in smart cards and merchant's terminals are protected by a hardware-security module. There are multiple certification authorities (CA) in the Visa Cash system and they are organized into different levels (international level, country, or region level).

13.2.5 Architecture and Workflow of Visa Cash

Participants in a Visa Cash system include: (1) the cardholder who holds a Visa Cash card; (2) the merchant whose terminal contains a card reader to read the user's card and a SAM smart card, to execute a transaction and to receive the transferred cash value; (3) the acquirer who collects and possibly aggregates transactions from several purchase devices for delivery to one or more system operators; (4) the card issuer who is responsible for the provision and distribution of integrated circuit cards, and who also authenticates load requests and transaction records, and provides cardholders with customer services. In addition, there may be a load acquirer through whom a load transaction and currency exchange transaction is initiated.

The typical workflows of a Visa Cash system include generic load transaction, processing a transaction between a cardholder and a merchant, and clearance. The descriptions for each of these three workflows follow.

Workflow for Generic Load Transaction

In this transaction, the cardholder loads money to the Visa Cash card from a linked account. The linked account is the cardholder's account at the card insurer.

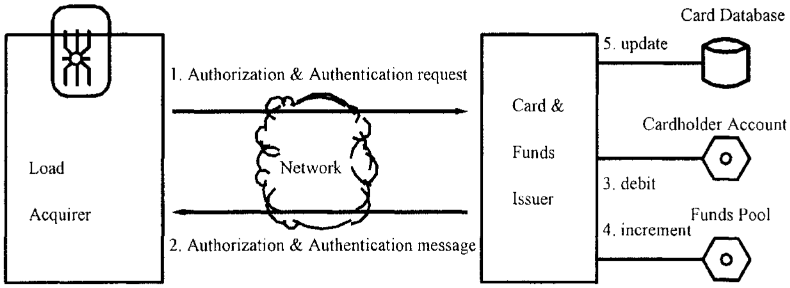


Fig. 13.1 Load workflow

(1) The load acquirer sends a combined authorization and authentication request to the card-and-funds issuer via the network.
(2) The card-and-funds issuer sends the positive combined authorization and authentication message to the load acquirer via the network. The electronic purse card is loaded.
(3) The card-and-funds issuer debits the cardholder account.
(4) Increments the funds pool of the currency loaded.
(5) The card-and-funds issuer updates the card database.

Transaction Workflow between a Cardholder and a Merchant

The cardholder places their Visa Cash card into the merchant's card reader. Then the Visa Card and the merchant's purchase secure application modules (PSAM) interact as follows (see Fig. 13.2):

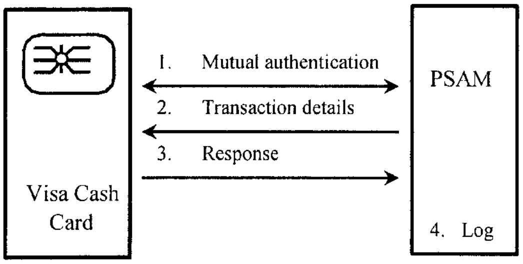


Fig. 13.2 Offline transaction workflow

(1) Mutual authentication to ensure that the Visa Card is valid and the PSAM is not a fraud.
(2) PSAM constructs the correct cryptographic details about the transaction and sends the details to the customer's card to activate the Visa Cash card to do the transaction.
(3) Visa Cash card sends a response to the PSAM and the PSAM verifies the response from the Visa Cash card to ensure that funds were deducted.
(4) PSAM produces the transaction log entry.

Workflow for Clearance and Settlement

The offline transaction between the cardholder and merchant has been done. The merchant wants to get back the money that the cardholder paid (see Fig. 13.3).

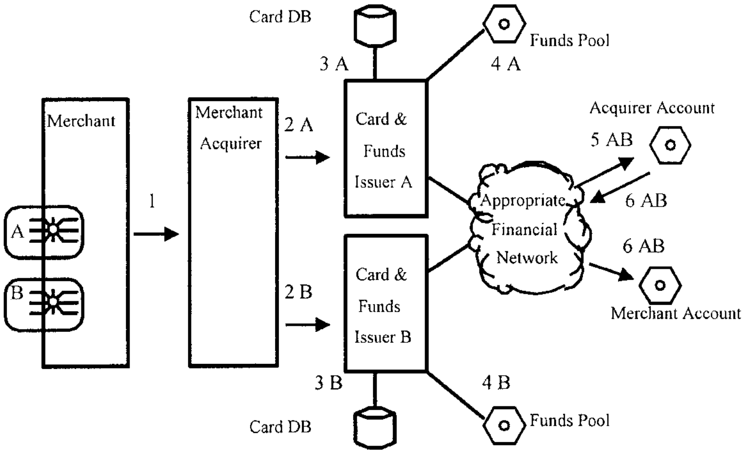


Fig. 13.3 Clearance and settlement workflow

(1) The merchant sends the transactions to the merchant acquirer.

(2) The merchant acquirer sends the transactions to the card issuer via the respective networks.
(3) The card issuer updates their card databases.
(4) The card issuer's funds pool of the transaction currency is decremented and the merchant acquirer's account is credited via the respective networks.
(5) The merchant acquirer's account is debited through the appropriate financial network.
(6) The merchant's account is credited through the appropriate financial network.

13.3 iPIN E-Payment

iPIN is an e-payments company headquartered in Northern California's "Silicon Valley" [13.2]. It also has offices in North America, Europe, and Asia. iPIN provides e-payment solutions for Web and wireless purchase transactions. It also offers e-payment solutions to telecommunications operators, financial institutions, automotive OEMs, and ISPs who want to provide their consumers with choices in paying for digital and hard goods across access devices, alternative payment options, and multiple networks and standards. We chose iPIN as the second example for e-payment systems and solutions because it is a premier provider of e-payment software worldwide, and because it has experience in the banking, credit card, telecommunications, software, and Internet industries.

13.3.1 Design Goals of iPIN

The iPIN e-payment solution is a platform to support payment across vertical markets in both the virtual and physical world. With iPIN, banks, telecommunications companies, and other partners can offer their customers the next generation of online and wireless payment products. Consumers can make payments for their purchases from any Internet-enabled device by selecting the payment method. For example, the consumers can make their payment by direct debit, credit cards, or prepaid cards.

By authorizing companies to take advantage of iPIN e-payment solutions, customers can now build their own payment networks, interconnect with industry leaders and merchants across the globe, or simply make their existing e-commerce and m-commerce initiatives more productive.

13.3.2 iPIN E-Payment Features

The features of the iPIN e-payment solution include:

- The ability to support multiple access devices: handsets (wireless devices), personal digital assistants, and Internet appliances.
- Multiple payment options enabled by iPIN's multiple payment instrument (MPI) module. An electronic MPI is similar to a physical wallet in that the consumers can select from several options in their billfold, i.e., debit cards, credit cards, stored value cards, etc., to make purchases.
- The iPIN e-payment solution is able to operate interchangeably across global networks and standards.
- The iPIN e-payments solution can be adapted to fit customers' needs by plugging into one of the add-on iPIN e-payment modules. In addition, business rules and security measures can be customized to meet specific requirements. The modular applications allow iPIN to more closely match customers' needs.

13.3.3 iPIN E-Payment Functions

iPIN e-payment provides the following functions:

- Account management: users are able to view their transaction history and change their security settings.
- Person-to-person payment (P2P).
- Payment with direct debit account.
- Payment with pre-paid/stored value.
- Bill aggregation.
- Exceptions processing.
- Sales and remittance reporting.
- Payment with any Internet-enabled PC, PDA, or mobile phone.
- Roaming freedom: layered authentication allows users to cross devices anytime, anywhere.

13.3.4 Security of iPIN

The security of iPIN e-Payment includes the following components:

- Data segregation:
  When a consumer initiates a purchase, they are redirected to the iPIN e-payment solution platform residing behind the partner's firewall for

authentication, account selection, and authorization. Once authorized, the partner sends an authorization message to the merchant.

- Message integrity:
  All messaging between consumers, merchants, and partners is digitally signed using private keys, and encrypted using SSL.
- Authorization timeouts.
- Anti-robot and anti-spoof mechanisms.
- Transaction and spending limits.
- Non-payment account suspension.

The iPIN security mechanism is scalable. The customers using the iPIN e-payment solution are able to select the level of the security that is required as well as the method that will be used in the purchase. For example, in the purchase authentication, the iPIN payment platform may simply read the terminal ID for a US$ 0.50 purchase of information to be placed against a monthly bill, and require no other form of authentication. At the other end of the spectrum, a $150 purchase could be authorized against a private credit line or a checking account that may require the customer to authenticate via a digital certificate, a biometric parameter, password, and/or secret questions and answers.

Using the standard implementation of the iPIN security system, the sensitive account information is never electronically transmitted to the merchant when making a purchase, so iPIN customers are assured of the end-consumers' security.

13.3.5 Architecture and Workflow of iPIN

The iPIN e-payment architecture shown in Fig. 13.4 is comprised of the following components:

- Transaction acquisition,
- Transaction processing and real-time accounting,
- Clearing and settlement,
- Customer and merchant care.

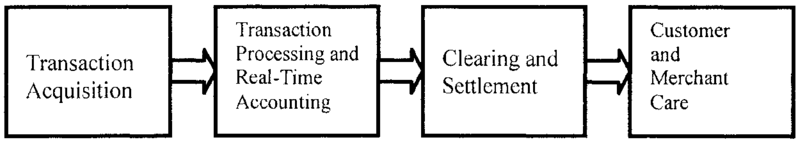

Transaction Acquisition
Transaction Processing and Real-Time Accounting
Clearing and Settlement
Customer and Merchant Care

Fig. 13.4 iPIN e-payment function and architecture

Transaction Acquisition

The transaction acquisition allows the secure capture and authorization of a user transaction across multiple electronic channels. The functions include (see Fig. 13.5):

- Transaction authorization and confirmation,
- Multi-level user authentication,
- Transaction connectivity and routing.

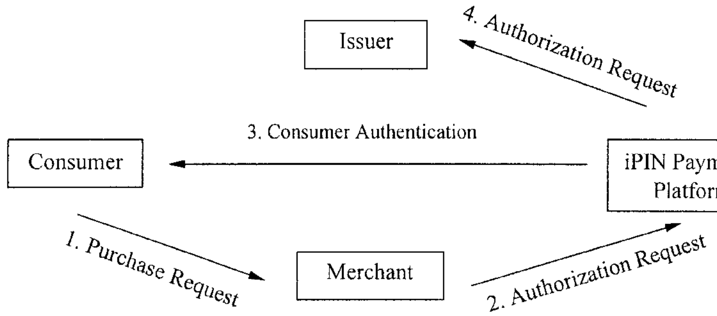


Fig. 13.5 Transaction acquisition

Transaction Processing and Real-Time Accounting

The functions of this component include transaction accounting, consolidation, revenue sharing, and transaction fee calculation and reporting (see Fig. 13.6).

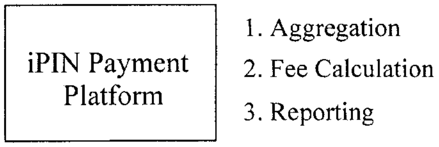


Fig. 13.6 Transaction processing and real-time accounting

Clearing and Settlement

This component facilitates the management of accounts receivable, accounts payable, and general ledger interfaces to financial systems. It includes the following four interfaces (see Fig. 13.7):

- Interface for accounts receivable and accounts payable,
- Interface to treasury management,
- Interface to financial management,
- Interface to settlement network.

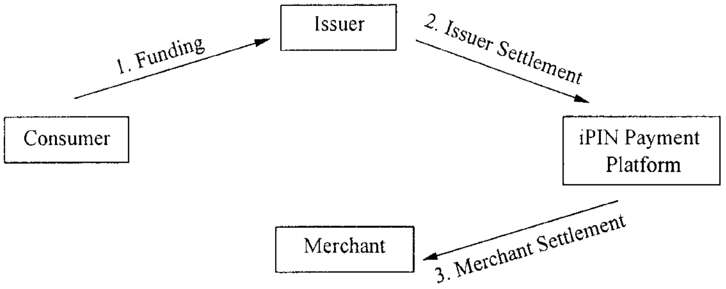


Fig. 13.7 Clearance and settlement

Customer and Merchant Care

This component provides the following tools about which the customer and merchant care (see Fig. 13.8):

- iPIN customer tools:
  Administrative tools, activity reporting, customer support.
- End-consumer tools:
  Payment panel, account management, and self-care.
- Merchant tools:
  Activity reporting, exception, and dispute handing.

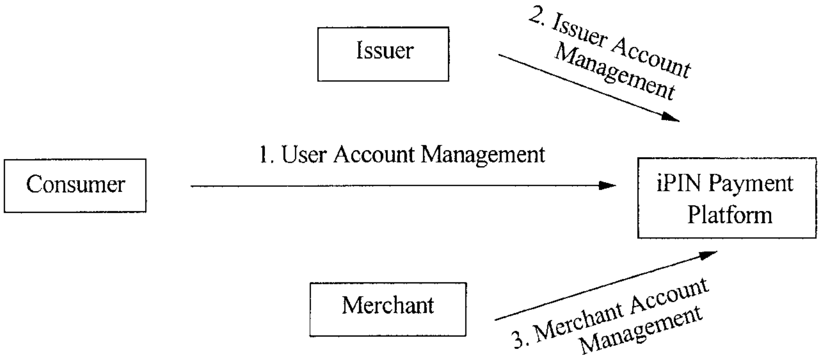

Issuer
2. Issuer Account Management
1. User Account Management
Consumer
iPIN Payment Platform
Merchant
3. Merchant Account Management

Fig. 13.8 Customer and merchant care

13.4 PayPal

PayPal is designed for handling payments and money transfers for small businesses, online merchants, individuals, and others currently poorly served by traditional payment mechanisms. The PayPal network extends the existing financial infrastructure of bank accounts and credit cards and creates a global payment solution. PayPal enables any business or consumer with an email address to securely, conveniently, and cost-effectively send and receive these payments. Any business or consumer with an email address can send and receive payments online instantly.

Paypal is now offering services to users in 38 countries including the US. It has over 16 million registered users, including more than 3 million business accounts.

13.4.1 Design Goals of PayPal

E-Commerce Services for Businesses and Individuals

PayPal [13.3] enables quick and easy payment processing for websites, classified ads, auction sites, and email (i.e., anywhere a person wants to collect payments online). PayPal supports the following functionality:

- Accept credit card and/or bank account payments for single or multiple item purchases.
- Sell products to PayPal users in 38 countries outside the US.
- Collect subscription or recurring payments.
- Gather donations or “tips.”
- Get instant notification when you receive payments.

Online Auction Services

PayPal also supports online auction services. One can

- Insert PayPal logos into any number of listings automatically.
- Notify winning bidders instantly.

Person-to-Person Payment Services

PayPal makes person-to-person payment easy without the headache of the traditional person-to-person payment process involving checks, stamps, and envelopes. With PayPal one can:

- send money online from a credit card or bank account,
- request money from an individual or group,
- use a virtual debit card for safe and easy online shopping.

13.4.2 Features of PayPal

Paypal has the following features:

- Three types of accounts:
  PayPal offers three types of accounts according to the different needs. These are personal, premier, and business accounts.
- No service fee or low service fee:
  For a personal account, sending money and receiving money is free. For premier and business accounts, sending money is free of charge, while for receiving money the fee is very low.
- The PayPal VISA credit card account will be governed by the Providian, which protects customer privacy and handles customer information in a secure and confidential manner.
- It allows customers to pay anyone who has an email address.
- Over 33,000 websites accepted PayPal at the time of writing.

13.4.3 Functions of PayPal

There are two main functions in the PayPal system, namely, "send money" and "request money."

Send Money

"Send Money" allows the customer to pay anyone who has an email address. One can make the payment by just entering the recipient's email address and the amount that one wishes to send. The payment can be made either using a credit card or through a checking account.

Request Money

"Request Money" offers the customer an organized method to request and track funds. To send an auction invoice or a personal bill, one just needs to enter the recipient's email address and the amount that one wishes to request.

13.4.4 Security of PayPal

PayPal stores credit card and bank account information only in encrypted form on computers that are not connected to the Internet. PayPal restricts access to the customer's personally identifiable information to employees who need to know that information in order to provide products or services to the customer.

Secure Web Sites

When customers log into their PayPal accounts, customers will always be on a secure web site. Whenever entering sensitive personal information (such as checking account or credit card numbers) onto the secure web site, the web site encrypts the information that the customer sends to and receives from the site.

Data Security and Encryption

PayPal automatically encrypts the confidential information in transit from the customer computer to PayPal using the secure sockets layer protocol (SSL) with an encryption key length of 128 bits. Before a customer even registers or logs on the PayPal site, the server checks that the customer is using an approved browser, that is, one that uses SSL 3.0 or higher. Once the customer information reaches PayPal, it resides on a server that is secured both physically and electronically. The servers sit behind a secure firewall and are not directly connected to the Internet, so that the customer's private information is accessible only to the authorized computers.

PayPal's Identity Verification System

Verification provides the customer with some more information about the people with whom the customer transacts through PayPal, so that the customer may make more informed decisions.

There are several ways to take advantage of PayPal's verification process and decrease customer-fraud risks:

- When the customer receives a payment:
  After logging into the customer's account, the customer can go to the "History" sub tab of the "My Account" tab, find the payment in question and choose the status link (e.g., "Pending") in the Status column. This will take the customer to a payment details page. Next to the sender's name, the customer will find their verification status (verified, unverified, or international).
- When the customer sends a payment:

> As the customer is sending a payment, on the "Send Money: check the details of your payment" page, a reputation link is provided where the customer may view the recipient's status (verified, unverified, or international).

Additional Verification

If PayPal cannot verify the information that the customer provides, or if the customer requests a withdrawal by check to an address other than the customer-verified credit card billing address, PayPal asks the customer to send additional information to PayPal by fax (such as the customer's driver's license, credit card statement, and/or a recent utility bill or other information linking the customer to the applicable address), or to answer additional questions online to help verify the customer's information.

13.4.5 Architecture and Workflow of PayPal

Workflow of PayPal Email Payments

The PayPal email payments system allows the user to send money instantly and securely to anyone with an email address. The workflow of PayPal email payments is shown in Fig. 13.9. The description of the workflow is as follows:

(1) The payer signs up and enters bank account information, the payee's address, and the dollar amount to PayPal.
(2) The payment is transferred from the payer account to the payee account.
(3) The payee gets an email notification with a link.
(4) The payee follows the link to sign up.
(5) The payee withdraws money or mails it to others.

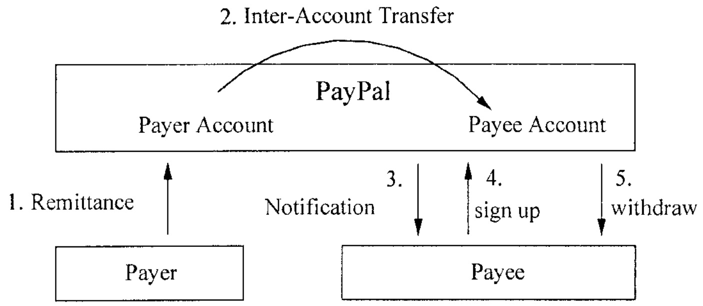


Fig. 13.9 Workflow of PayPal email payments

Workflow of PayPal Mobile Home Banking

Paypal mobile home banking allows peer-to-peer payments via wireless PDAs or web phones and allows money to be transferred from a credit card account to the recipient's PayPal account. The workflow of PayPal mobile home banking is shown in Fig. 13.10.

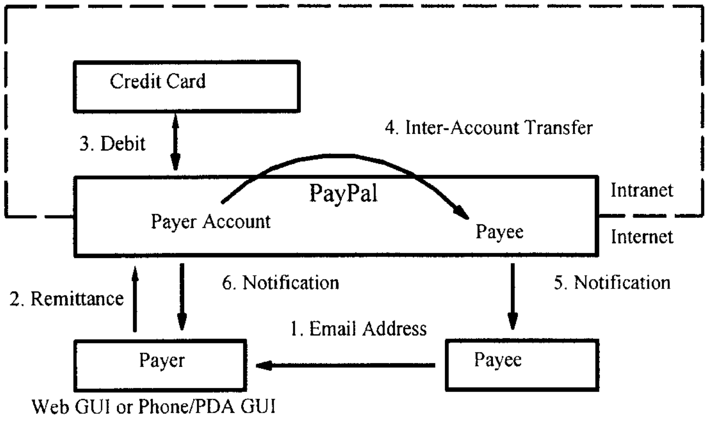


Fig. 13.10 Workflow of PayPal mobile home banking

The description of the workflow in Fig. 13.10 is as follows:

(1) The payee sends the payee's email address to the payer.
(2) The payer signs up and enters the credit card (or bank account) information, the payee's address, and the dollar amount to PayPal.
(3) Using the credit card, the payment is deducted from the payer's credit card.
(4) The payment is deducted from the payer's PayPal account and the payment is credited to the payee's PayPal account.
(5) The payee receives an email notification.
(6) The payer receives an email notification.

13.5 Summary

In this chapter, we have discussed three e-payment solutions: Visa Cash, iPIN, and PayPal. Visa Cash is an electronic purse, and it belongs in the digital cash payment category. iPIN provides e-payment solutions for Web and wireless purchase transactions. It supports various payment methods. For example, with iPIN e-Payment solutions, the consumers can make their payments by direct debit, credit card, or prepaid card. PayPal conducts payments through email. With PayPal, any business or consumer with an email address can send and receive payments online instantly.

13.6 References

[13.1] Visa Cash. http://international.visa.com/ps/products/vcash/ and http://international.visa.com/fb/paytech/vcash.jsp.

[13.2] iPIN. http://www.ipin.com/.

[13.3] PayPal. http://www.paypal.com/.

[13.4] M. H. Sherif (2000) Protocols for secure electronic commerce. CRC Press, Boca Raton London New York Washington DC.

[13.5] P. Wayner (1997) Digital cash (2nd ed.). AP Professional, Boston New York London.

14 Challenges and Opportunities in E-Payment

Weidong Kou

University of Hong Kong
Pokfulam Road, Hong Kong

14.1 E-Commerce Challenges: E-Payment Security and Privacy

The rapid growth of online business transactions indicates that e-commerce over the Internet is an irreversible trend. Based on various reports from leading international consulting firms such as Forrester Research and International Data Corporation, it is predicted that B2B e-commerce will be worth as much as 7 trillion of US dollars in a few years, and B2C will also be worth over hundreds of billions of US dollars in the United States alone.

Although e-commerce has huge potential, there are challenges for people in adopting e-commerce in their daily life. For example, in Hong Kong it was found that only 4% of Hong Kong Internet users bought goods or services or traded securities online, according to a recent survey conducted by the Census and Statistics Department of Hong Kong. In the United States, according to E-Stats released by the Department of Commerce, retail e-sales were $10 billion US dollars in the fourth quarter of 2001, about 1.2% of total retail sales in that quarter. In Europe, according to Datamonitor, the size of the European online market in 2001 is about $3.23 billion US dollars.

What are the barriers that prevent e-commerce from reaching the mass market? A recent survey report shows that payment security is a major concern for online shopping. Consumers are not willing to expose their credit card numbers online if they are not certain whether the numbers are securely transferred and saved.

Another important issue is privacy. Currently, online merchants usually require consumers to fill in detailed private information, including address and credit card information. Consumers do not like to have their shopping activities easily traceable. Consumers want easy access to premium content without the hassle of disclosing personal credit card information to unknown sites, or going through a tedious registration and authorization process.

A recent survey on merchants shows that they are concerned that B2C e-commerce may not be cost-effective if there are not enough Internet buyers and that the cost of setting up and operating a payment-enabled Web storefront is high. These concerns contribute to the slow growth of online merchants.

In fact, both consumers and merchants face one common problem from different perspectives, that is, the lack of a secure, reliable, cost-effective, and easy-to-use online payment solution. There are some electronic payment systems for B2C e-commerce commercially available. These systems are not flexible enough to handle different payment methods. Some systems have captured the US market share, and some have captured small segments of the European and Asian markets. However, until now none of these payment systems has been able to achieve the critical mass required for B2C e-commerce to take off across the entire globe.

14.2 E-Payment Systems Supporting Multiple Payment Methods

By analyzing the current business problems in B2C e-commerce, it is found that these problems can be looked at from the three participants' perspectives:

- Consumers' concern about privacy, payment security, and convenience.
- Merchants' concern about the number of online buyers, payment options, and high cost of setting up payment-enabled Web storefronts.
- E-payment service providers face the problem of not having enough consumers and merchants.

To address these problems and meet the challenges of e-commerce, there is a need to develop an e-payment system that supports multiple payment methods including credit/debit cards, prepaid cards, and a variety of smart cards. It also supports payment through an account with a telephone company or an ISP.

Through this e-payment system, the above problems are addressed from three perspectives:

- From the consumers' perspective, to enhance payment security and provide multiple payment options, such an e-payment system should include a strong privacy protection mechanism. This will increase consumers' confidence in online shopping and payment.
- On the merchants' side, through the multiple payment options, merchants will be exposed to more Internet users thus enlarging their customer base. Consumers will also benefit from a large merchant network.

- With multiple payment options, strong security, and proper privacy protection, payment service providers will benefit from increased numbers of online consumers and merchants.

One of the key requirements of the e-payment system is to provide a privacy protection mechanism. To ensure the protection of consumers' privacy, a secure scheme should be included in the system to protect consumers' privacy and anonymity in e-commerce. The system can be designed in such a way that no single party knows the details of the entire shopping transaction. The transaction can be traced only if three parties (the merchant, the payment gateway, and the payment service provider) are working together under a court order. The idea is that the merchant does not have to know the customer identity; the merchant does not need to access any account information that is private to the customer; the bank does not need to know the order information to authorize the payment from the customer's account as long as the customer has enough money to cover the transaction.

The privacy protection scheme is based on blind signature techniques. A blind signature is a regular digital signature with the following features:

- The signatory does not know the content of the message that they are signing, as the message has been blinded before reaching them.
- From the signed blinded message, the signature on the message can be recovered by the party who blinded the message in the first place.
- After the message and the signature are revealed to the public, the signature can be verified, but the signatory cannot trace who blinded the message in the first place.

Blind digital signature has been studied for sometimes [14.1]. The objective is to design and implement the e-payment system with privacy protection and secure payment, based on a blind digital signature.

To protect consumers' privacy, the following design requirements have to be met: First, it is not acceptable for any single party in the system to know every detail of an online transaction, for example, the consumer's identification, the product that the consumer is buying, the quantity, and the price. Second, the merchant needs to have the payment authority's confirmation so that the merchant is guaranteed to be paid. Third, the payment authority must be comfortable with providing such a confirmation without knowing the transaction details. Here, the payment authority can be either a payment service provider or a billing company such as an ISP. The existing blind-signature protocols do not support the above design requirements.

To protect the consumer, the merchant will provide the consumer with a receipt of payment for the products purchased. The receipt contains a digital signature of the merchant and the purchase date and time. The exchange or refund can be arranged with this receipt according to the purchase agreement between the merchant and the consumer. Such an agreement can be included in the receipt. For example, the "final sale" products cannot be refunded or exchanged; or some products can be exchanged within 7 days after the purchase; or some products can be refunded provided that they are not opened or have a defect.

14.3 Smart Cards and Digital Cash

Smart cards provide a means of storing and processing value for digital cash. In particular, reloadable smart cards have become very popular nowadays. For example, a number of cities around the world are using or plan to use smart cards in their public transportation systems. Such cities include Washington DC, San Francisco, San Diego, Montreal, London, Singapore, Hong Kong, and many others. The Washington metropolitan transit pilot using smart cards was launched in 1999. It has 210,000 smart cards in circulation. As much as $200 US can be loaded onto the cards. Among these examples, the best one is the Octopus card in Hong Kong, as it has become a necessity for people's daily travel needs. With an estimated 10 million passenger journeys each day on Hong Kong's wide variety of public transportation services, the Octopus card provides evidence for the potential success of digital cash payment.

It was reported recently that the Octopus card is expanding its business into a variety of applications apart from public transportation, including 7-Eleven convenience stores, ticketing for the Broadway cinema chain, fast-food shops, such as Café de Coral, Maxim's, Starbucks, and even the giant hamburger chain McDonald's.

To examine the success of the Octopus card, in addition to what was discussed in Chapter 5, the following factors are crucial:

- Anonymity: no customer information is carried on the Octopus card and when using the Octopus card to make a payment, no privacy information is involved in the transaction.
- Risk is small as most people typically only load less than $200 Hong Kong dollars (or US$ 25.64) onto the card. If the card is stolen or lost, it is not a big concern to people.
- It is easy to reload the card with cash. There are many places in Hong Kong where people can reload their Octopus cards with cash, for example, at MTR (subway) stations or 7-Eleven convenience stores.

The Octopus card is an excellent candidate payment method for e-commerce applications over the Internet. However, there is a major issue that has to be resolved before the Octopus card can become popular as the preferred e-payment method over the Internet, that is, how to read from and write to an Octopus card using PCs, PDAs, pocket PCs, cellular phones, or other pervasive devices, as currently most PCs or pervasive devices are not equipped with an Octopus card reader. It is not realistic to ask every owner of a PC or pervasive device to purchase such a reader. In addition, there are issues of security and reliability related to the Octopus card readers.

14.4 Micropayment Issues and Solutions

Micropayment is particularly applicable to e-commerce over the Internet. Micropayment deals with a very small payment, typically in the range from one cent to a few dollars. Sometimes, the payment can be even a fraction of one cent. The applications for micropayment include "pay per click" for an image, a piece of music (or video), online gaming, an online report, or a piece of online information. The business justification for micropayment is the huge online customer base even if each transaction value may be tiny. The popular credit card-based online payment method may not be appropriate for small-value transactions because there is a minimum credit card charge, usually about a quarter (US$ 0.25) or so. The amount in a micropayment transaction may be too low to justify the payment using a credit card.

The major issues for micropayment are that the payment-processing cost is relatively high compared to the amount in a micropayment transaction, and the cost of implementation of existing micropayment schemes is also high. When implementing a micropayment system, one needs to consider the cost of the infrastructure support such as the communication, computation, hardware equipment, and associated software. In addition, there are other costs involved, such as the cost of customer acquisition, the cost of handling disputes and chargebacks, and the cost of customer support.

To reduce the payment-processing cost, one obvious solution is to aggregate many small micropayments into a few regular payments. The question is how can the aggregation be done. There are several ways to do the aggregation. For example, it can be performed per user session, or it can be done in different user sessions but limited to a specific online merchant. Of course, it can also be done across different merchants through a payment-service provider. Different aggregation schemes require different accounting and user transaction management approaches. Again, we encounter the cost issue for implementing the aggregation schemes for micropayment, which could be substantially high.

To handle the dispute, the customer needs to directly interact with the online merchant. In addition, the proper security measures, such as digital signatures, must be in place. In case there is a dispute, at least digital signatures can provide non-repudiation of a transaction.

To save on communication costs, it is desirable to handle the payment offline and to make the payment protocol non-interactive, for example, through an email, to reduce the number of round-trip communications between the merchant's server and the customer's browser. The offline payment can be achieved through the aggregation of authorizations and deposits. The micropayment service provider only needs to examine the value flow when necessary.

14.5 Summary

In this chapter, we have discussed the challenges and opportunities of e-payment. The major challenges for e-payment are as follows:

- **Freedom to choose an e-payment method:** giving online customers freedom to choose which e-payment method they prefer (that means, the e-payment systems/solutions need to support multiple e-payment methods).
- **Security:** how to make e-payment more secure to ensure the safety of the customers' online transactions.
- **Privacy:** how to protect the online customers' private information
- **Anonymity:** how to make the e-payment anonymous.
- **Risk:** how to reduce the online customers' risk involved in e-payment.
- **Convenience:** how to provide the online customers with convenience.
- **Cost:** how to reduce the implementation and processing costs of e-payment systems/solutions.

The opportunities are to respond to the online customers' needs and meet the challenges identified above by developing new e-payment systems/solutions.

14.6 References

[14.1] D. Chaum (1983) Blind signature for untraceable payments. In: Advances in cryptology. Plenum Press, New York, pp. 199–203.

[14.2] E. Mohammed, et al. (2000) A blind signature based on the discrete ElGamal signature. In: Proceedings of the 17th National Radio Science Conference, pp. C25:1–6.

[14.3] J. L. Camenisch, J. M. Piveteau, M. A. Stadler (1994) Blind signature based on the discrete logarithm problem. In: Advances in cryptology – Eurocrypt'94, pp. 428–432.

[14.4] R. Anderson, C. Manifavas, C. Sutherland (1996) Netcard – a practical electronic cash scheme. In: Cambridge Workshop on Security Protocols.

[14.5] D. Chaum, S. Brands (1997) Minting electronic cash. IEEE Spectrum, February Issue.

[14.6] D. Chaum (1992) Achieving electronic privacy. Scientific American, August Issue.

[14.7] A. Herzberg, H. Yochai (1997) IBM-MP: charging per click on the web. In: Proceedings of the 6th WWW Conference.

[14.8] W. Song (2000) An investigation on micropayment and its implementation. Project Report, ETI, The University of Hong Kong.

[14.9] R. Weber (2000) Market analysis of digital payment systems. Technical Report (TUM-I9818). Institute for Informatics, Technology University of Munich.

[14.10] Z. Li, W. Kou (2002) A batch verification scheme for RSA digital signatures. Submitted for publication.

[14.11] Z. Huang, K. Chen, W. Kou (2002) A blind digital signature scheme. Submitted for publication.

[14.12] D. Zheng, W. Kou, K. Chen, Z. Gan (2002) Subliminal-free digital signature schemes in the presence of an active warden. Submitted for publication.

Glossary

Application programming interface (API)

A set of the specific methods, services, or instructions prescribed by a computer program by which a programmer writing an application program can make requests of the computer program.

Authentication

Providing assurance that the entity (user, host, and so forth) requesting access is the entity that it claims to be.

Behavioral biometric

A biometric that is characterized by a behavioral trait that is learned and acquired over time, rather than a physical or physiological characteristic (contrast with physical biometric).

Biometric system

An automated system capable of: capturing a biometric sample from an end-user; extracting biometric data from that sample; comparing the biometric data with that contained in one or more reference templates; deciding how well they match; and indicating whether or not an identification or verification of identity has been achieved.

Certificate

A digital credential in a public-key cryptography system, which contains the certificate holder's name and public key, a serial number, the expiration date of the certificate, and the digital signature of the certificate authority that issued the certificate.

Certificate authority (CA)

A trusted entity that is part of a public key infrastructure (PKI) and that creates, issues, and manages certificates for PKI users.

Certificate revocation list (CRL)

A list of certificates issued by a certification authority (CA) that are no longer valid. The CRL is maintained and published by the CA.

Chargeback

A process where the PSP requires the merchant to return funds for a disputed or cancelled payment.

Cookie

A file sent by a web server to a browser and stored by the browser. The cookie includes a destination address as a URL, possibly with wildcards. When the browser sends any request to a web server corresponding to the destination address, the browser attaches the cookie to the request. Cookies are used to identify the consumer, especially for repeat access to the same site.

Credit risk

The risk that the consumer will fail to pay the payment service provider (PSP) for aggregated payments (when the PSP charges the consumer after payments were made).

Digital cash

An electronic form of cash in a cash-like e-payment system with which a person can make online payment for goods or services purchased over the Internet.

Digital check

An electronic form of a check in a check-like e-payment system where the check can be conveyed across computer networks.

Digital signature

A digital string produced by applying a cryptographic algorithm with the private-key information on a message/document to authenticate the message/document.

Dispute resolution

A process invoked by the consumer to cancel a transaction (payment) that the consumer believes was not authorized or should be cancelled for other valid reasons.

Enrollment

The process of collecting biometric samples from a person and the subsequent preparation and storage of biometric reference templates representing that person's identity.

Enrollment time

The time a person must spend to have their biometric reference template successfully created.

Enrollment station

A workstation at which an individual's biometrics (fingerprint, voiceprint, etc.) and personal information (name, address, etc.) can be entered into a bioidentification system.

Escrow agent

A party that receives payments from the consumer and goods from the merchant, and, only when both were received properly, delivers the goods to the consumer and the payment to the merchant.

Extensible Markup Language (XML)

Universal format for structured documents and data on the Web, supporting the customized tags, enabling the definition, transmission, validation, and interpretation of data between applications and between organizations.

Extraction

The process of converting a captured biometric sample into biometric data so that it can be compared to a reference template.

False acceptance rate (FAR)

The probability that a biometric system will incorrectly identify an individual or will fail to reject an impostor. Also known as the Type II error rate.

False rejection rate (FRR)

The probability that a biometric system will fail to identify an enrollee, or verify the legitimately claimed identity of an enrollee. Also known as the Type I error rate.

HyperText Transfer Protocol (HTTP)

Standard transfer protocol used in the Internet, which defines how messages are formatted and transmitted, and what actions web servers and browsers should take in response to various commands. For example, when entering a URL in a browser, one actually sends a HTTP command to the Web server and instructs it to fetch and transmit the requested Web document.

Internet protocol security (IPSec)

A set of security functions and options available at the IP level.

Internet service provider (ISP)

A company that provides users with access to the Internet. For a monthly fee, the ISP provides users with a software package, user ID, password, and access phone number. Some ISPs also provide users with a modem to enable users to access the Internet.

Irreversible transactions

Payments that are done in such a way that the PSP cannot technically reverse them with a chargeback to the merchant, since there is no identification of the merchant.

Kerberos

In Greek mythology, the three-headed dog that guards the entrance to the underworld. In network security, Kerberos is a cryptographic authentication system that makes use of a third-party server to authenticate clients and servers. The system was developed in the Athena Project at the Massachusetts Institute of Technology (MIT).

Key

(a) A small piece of data used in conjunction with an algorithm to encrypt or decrypt messages/data of arbitrary size (see also PKI), or (b) an attribute whose value serves to identify a unique record in a database/table (e.g., employee ID number may be the primary key used to locate and identify a specific employee's personnel data, such as name, address, telephone number, salary).

Mail-order telephone (MOT)

Classification of credit card transaction performed when the credit card is not physically present for verification

Message authentication code (MAC)

A fixed-size binary code obtained by applying a shared-key cryptographic algorithm to an arbitrary amount of data to serve as an authenticator of the data.

Micropayment

A payment of small amounts, close to or below the credit card minimal fees (of about 20 US cents).

Micropayment system

A system allowing merchants to charge many payments of small amounts (micropayments) from customers over open data networks such as the Internet by using one or more payment service providers (PSPs).

Mobile agents

A computer program that represents a user and can migrate autonomously from node to node in a computer network, to perform some computation on behalf of the user.

Mobile agent host

A computer program running in a networked computer that provides an execution environment where mobile agents can execute their code and can communicate with one another.

NetBill

A payment system where the digital check is used to sell and deliver low-priced information goods.

NetCheque system

A distributed accounting service supporting the credit-debit model of payment.

Nonrepudiation

A proof that the consumer approved a particular action, typically a payment.

Octopus card

A smart card system used in Hong Kong for local transportation fare collection.

Offline payments

Payments between the consumer and the merchant which do not require communication with other parties such as the PSP.

Order information (OI)

Information included in a SET transaction to describe the transaction.

Original equipment manufacturer (OEM)

A biometric organization (manufacturer) that assembles a complete biometric system from parts, or a biometric module for integration into a complete biometric system.

Payment approval

A process where the customer agrees to a particular payment.

Payment authorization

A process where the PSP takes responsibility for a payment, in particular by indicating that there are funds to cover the payment.

Payment gateway (PG)

Entity in a SET transaction that handles credit card verification and authorization of transactions.

Payment information (PI)

Information included in a SET transaction to describe a payment (such as the credit card holder and number).

Payment order (PO)

A message indicating payment to the merchant.

Payment routing table (PRT)

A message sent by a PSP to a merchant or another PSP, indicating the terms under which the PSP sending the PRT is willing to receive payment orders issued by other PSPs.

Payment service provider (PSP)

An entity that maintains a long-term relationship with customers and merchants, receiving payments of aggregated (large) amounts from customers, and passing aggregated payments to the merchants.

Penalty payment

A payment by a merchant who has had too many disputes and/or chargebacks.

Personal identification number (PIN)

A security method whereby a (usually) four-digit number is entered by an individual to gain access to a particular system or area.

Physical/physiological biometric

A biometric that is characterized by a physical characteristic rather than a behavioral trait (contrast with behavioral biometric).

Prepayment

Requiring funds to be deposited in advance.

Private key

In public-key cryptography, this key is the secret key. It is primarily used for decryption but is also used for encryption with digital signatures.

Public key

In public-key cryptography, this key is made public to all. It is primarily used for encryption but can be used for verifying signatures.

Public-key cryptography

Cryptography based on methods involving a public key and a private key.

Public-key infrastructure (PKI)

Structure used to issue, manage, and allow verification of public-key certificates. PKI is a security framework for messages and data, based on the notion of a pair of cryptographic keys (i.e., one public and one private) and used to facilitate security, integrity, and privacy.

Radio frequency identification (RFID)

The use of radio waves to facilitate wireless (contactless) communication with a chip or device.

Record aggregation

Replacing multiple separate documents, e.g., payment orders, with a single aggregated document, e.g., a payment order.

Rejection/false rejection

When a biometric system fails to identify an enrollee or fails to verify the legitimately claimed identity of an enrollee. Also known as a Type I error.

Response time/processing time

The time period required by a biometric system to return a decision on the identification or verification of a biometric sample.

Secure electronic commerce

A form of commerce conducted via electronic means, but designed with security in mind to enable identification, authentication, authorization, or payment processing.

Secure electronic transaction (SET)

A protocol for secure payment processing over the Internet in which credit card information (e.g., Visa, MasterCard) is not read or stored by a merchant. The protocol links many parties, including the customer, merchant, acquirer, and certification authorities. The protocol is designed to emulate card-present transactions.

Secure sockets layer (SSL)

A protocol originally introduced by Netscape to secure communication between web servers and web clients, supported by most web browsers and servers; superceded by TLS.

Semi-offline payments

Payment protocol where most transactions are offline (involve only communication between the consumer and merchant, not with the PSP), but sometimes communication with the PSP is necessary.

Smart card

A plastic card with an embedded chip to enable payment processing or digital identification. A typical smart card chip includes a microprocessor or CPU, ROM (for storing operating instructions), RAM (for storing data during processing), and EPROM (or EEPROM) memory for nonvolatile storage of information.

Software agent

A computer program that acts autonomously on behalf of a person or organization to accomplish a predefined task or a series of tasks.

Stored-value card

A smart card that comes preloaded with a certain amount of value (e.g., money, phone calls, transit trips), but which cannot be reloaded.

Stored-value payments

Offline payments where the consumers have complete control over the payments, in particular they can pay any merchant without contacting the PSP.

Subscriber identification module (SIM)

SIM is for GSM digital telephony. SIM smart cards are used to provide user authentication, voice/data integrity, and confidentiality.

Symmetric cryptography

A way of keeping data secret in which the sender and receiver use the same key.

T=0/T=1 Protocols

ISO 7816 asynchronous byte (T=0) and block (T=1) transmission protocols at the data-link layer, used for communication between a smart card and a reader.

Threshold

The acceptance or rejection of biometric data is dependent on the match score falling above or below the threshold. The threshold is adjustable so that the biometric system can be more or less strict, depending on the requirements of any given biometric application.

Transmission control protocol (TCP)

Internet protocol which manages message exchanges at the transport level.

Transport-layer security (TLS)

An IETF (Internet Engineering Task Force) standard protocol to secure communication between web servers and web clients, supported by most web browsers and servers; the previous version was called SSL.

Trusted third party

An organization or entity that is impartial to both the customer and the merchant (or buyer and seller), is trusted by both, and whose testimony is accepted as valid evidence in a court of law.

URL

Uniform Resource Locator specifying the unique address of a Web document.

Validation

The process of demonstrating that the system under consideration meets in all respects the specification of that system.

Wireless application environment (WAE)

The application framework for WAP applications. WAE consists of a set of standards that collectively define a group of formats for wireless applications and downloadable content.

Wireless application protocol (WAP)

A specification that allows users to access information instantly via handheld wireless devices such as cellular phones, pagers, and personal digital assistants (PDAs) through wireless communication networks and the Internet.

Wireless datagram protocol (WDP)

A datagram protocol for non-IP wireless packet data networks. WDP specifies how different existing bearer services should be used to provide a consistent service to the upper layers of the WAP architecture framework.

Wireless markup language (WML)

An XML-based markup language for wireless handheld devices, including cellular phones, pagers, and PDAs.

Wireless session protocol (WSP)

A protocol family derived from the HTTP version 1.1 standard with extensions for wireless data applications. WSP provides WAP applications with a consistent interface for session services.

Wireless telephony applications (WTA)

A framework for integrating wireless data applications with voice networks. WTA is a collection of telephony-specific extensions for call and feature control mechanisms that make advanced mobile network services available to the mobile users.

Wireless transaction protocol (WTP)

A protocol operating on top of a secure or insecure datagram service. WTP is an extremely lightweight request-response-acknowledge transaction protocol.

Wireless transport-layer security (WTLS)

A security protocol based on SSL and adapted to wireless networks and datagram transports.

About the Editor

Weidong Kou is Associate Director of the E-Business Technology Institute (ETI) and Adjunct Professor of the Department of Computer Science and Information Systems at the University of Hong Kong.

Prof. Kou also serves as Adjunct Professor of the Department of Computer Science and Electrical Engineering at the University of Maryland in US, Shanghai Jiao Tong University, South China University of Technology, and Lan Zhou University in China, and Guest Professor of Sun Yat-Sen University, South East University, and Beijing University of Posts and Telecommunications in China. In addition, he is a member of the Advisory Committee on Computer Science and Electrical Engineering at the University of Maryland in Baltimore, Co-chair of the Technical Advisory Board of the e-Generation Technology Center at Shanghai Jiao Tong University, Deputy Director of the Academic Committee of the National Key Laboratory of the Ministry of Education of China on Computer Networking and Information Security at Xidian University, and Technology Advisor for the IBM Great China Group's University Relationship Program.

Prof. Kou was a Research Professor at Rutgers University. He served as the Industrial Co-leader of a major project of the CITR (Canadian Institute of Telecommunications Research, a Canadian National Center of Excellence), *Enabling Technology for Electronic Commerce*, for more than three years. He served as a member of American national standard committees, ANSI X9B9 (Financial Image Interchange) and ANSI X3L3 (JPEG and MPEG), for more than four years. He has also served as a Guest Editor of special issues on e-commerce for the *International Journal on Digital Libraries* and the *ACM Computing Survey*. Prof. Kou was the Founding Chair of the International Symposium on Electronic Commerce (ISEC), and from 1998 to 2001 he was the General Chair and Program Chair for the ISECs and International Workshops on Technological Challenges of Electronic Commerce.

Since joining ETI at the University of Hong Kong in August 2000, Prof. Kou has been leading the e-commerce and wireless research and development efforts. Notably, Prof. Kou and his team were awarded the Innovation and Technology Fund (ITF). The ITF exercises, being highly competitive and placing great emphasis on local relevance, select only projects with great potential for Hong Kong. Out of a total of 19 proposals submitted in January 2001 by all sectors in Hong Kong, only three projects were awarded, and two of these came from the teams led by Prof. Kou. The total funding for the two winning projects was over 17 million Hong Kong dollars for a period of two years. One of these projects focuses on payment technologies for electronic commerce.

Prof. Kou has over 12 years of industrial experience in the software development and management in North America. Prior to joining ETI, Prof. Kou was Principal Investigator at the IBM Center of Advanced Studies in Toronto, Canada, where he led R&D projects on e-commerce. From 1995 to 1997, he was an Architect of a major IBM B2B e-commerce project for a national government at the IBM Industrial Solution Development Center in Canada. Prior to joining IBM in 1995, he was the Chairman of the Imaging Committee at the AT&T Imaging Systems Division, where he led a number of financial imaging projects. Prior to joining AT&T in 1991, he was Senior Software Engineer at Siemens in Toronto, Canada, where he invented compression algorithms and implemented them in Siemens' imaging products. He received various invention achievement and technical excellence awards from IBM, AT&T, and Siemens.

Prof. Kou has authored/edited five books in the areas of e-commerce, security, and multimedia technologies, and published over 50 papers on journals and conferences, including papers in prestigious journals such as *IEEE Transactions on Communications, IEEE Transactions on Signal Processing, IEEE Transactions on Acoustics, Speech and Signal Processing*, and *International Journal of Computer and Information Science*. He has also authored nine US and Canadian issued and pending patents.

One of Prof. Kou's books, *Digital Image Compression: Algorithms and Standards*, published by Kluwer Academic Publishers in 1995, has been widely used in a variety of universities around the globe as a recommended reference book, for example in Southern Queensland University in Australia, Catalunya University in Spain, Saarland University in Germany, Glasgow University and the University of London in the UK, Chalmers University in Sweden, Bandung Technology Institute in Indonesia, Stanford University, George Mason University, Ohio State University, and Albany New York State University in the US, and Calgary University in Canada.

Prof. Kou received his Ph.D. degree in Electrical Engineering in 1985 from Xidian University, and M.S. degree in applied mathematics in 1982 from Beijing University of Posts and Telecommunications, respectively. He was a Postdoctoral Fellow at the University of Waterloo, Canada, from April 1987 to February 1989.

Prof. Kou is a Senior Member of IEEE, and a member of the Advisory Committee of W3C. He was elected as a member of the New York Academy of Sciences in 1992.

Contributors

Gordon B. Agnew received his B.Sc. and Ph.D. in Electrical Engineering from the University of Waterloo in 1978 and 1982, respectively. He joined the Department of Electrical and Computer Engineering at the University of Waterloo in 1982. In 1984 he was a visiting professor at the Swiss Federal Institute of Technology in Zurich where he started his work on cryptography. Dr. Agnew's areas of expertise include cryptography, data security, protocols and protocol analysis, electronic commerce systems, high-speed networks, wireless systems, and computer architecture. He has taught many university courses and industry-sponsored short courses in these areas, and authored many articles. In 1985, he joined the Data Encryption Group at the University of Waterloo. The work of this group led to significant advances in the area of public-key cryptographic systems including the development of a practical implementation of elliptic-curve-based cryptosystems. Dr. Agnew is a member of the Institute of Electrical and Electronics Engineers, a member of the International Association for Cryptologic Research, a Foundation Fellow of the Institute for Combinatorics and Its Applications, and a Registered Professional Engineer in the Province of Ontario. Dr. Agnew has provided consulting services to the banking, communications, and government sectors. He is also a co-founder of Certicom Corp., a world leader in public-key cryptosystem technologies.

Amitabha Das received the B.Tech. degree in Electronic and Electrical Communication Engineering from the Indian Institute of Technology, Kharagpur in 1985, and the M.S. and Ph.D. degrees in Electrical and Computer Engineering from the University of California, Santa Barbara in 1989 and 1991, respectively. He is currently an Associate Professor in the School of Computer Engineering in Nanyang Technological University, Singapore. His current research interests include mobile agents, e-commerce, mobile databases, and data mining. Dr. Das is a member of IEEE. He can be reached by email at asadas@ntu.edu.sg.

Amir Herzberg is an independent security consultant. He graduated from the Technion, Israel, in 1982, and since then has worked as an engineer and researcher, mostly in security and communication areas. After completing his D.Sc. (Computer Science) at the Technion in 1991, Dr. Herzberg joined IBM Research, filling research and management positions in New York and Israel. During 2001 he was CTO of NewGenPay, a spin-off of the IBM Micro Payments project. Since January 2002, he has been a security consultant and teaches in Tel Aviv and Bar Ilan universities. Dr. Herzberg headed the W3C MicroPayments working group and contributed to several standards, including IP-Sec and SET. He is interested, and published, in the areas of security, applied cryptography, and fault-tolerant protocols. He is writing a book on "Secure Communication and Commerce Using Cryptography" (see http://amir.beesites.co.il).

Zheng Huang received his B.S. degree and M.S. degree from Tong Ji University in 1997 and 2000, respectively. He is currently a Ph.D. student at the Department of Computer Science of Shanghai Jiao Tong University. His advisor is Prof. Kefei Chen; information security is his major. He can be reached by e-mail at huangzheng@cs.sjtu.edu.cn.

Ed Knorr is a tenure-track instructor at the University of British Columbia (UBC). He received his Ph.D. in Computer Science from UBC. His previous degrees include an M.Sc. degree from UBC, and a B.Math. degree from the University of Waterloo. Dr. Knorr's research interests include data mining, outliers, database systems, and electronic commerce (e.g., security, privacy, usability, smart cards, and digital money).

Hui Li graduated from Fudan University in 1990 and received his Ph.D. degree from Xidian University in 1998. He is currently an Associate Professor at the School of Communication Engineering, Xidian University and Deputy Director of the Academic Department of the Key Lab for Computer Networking and Information Security. His research interest is in the area of information security.

Zichen Li received his Ph.D. degree in Signal Design and Information Processing from Beijing University of Posts and Telecommunications in 1999. From 1999 to 2002, he was a postdoctoral fellow at Tsinghua University. Dr. Li has been an Associate Professor and the Chairman of the Department of Computer Science and Technology at Jiaozuo Institute of Technology (JIT), Henan Province, China. Dr. Li is currently on leave from JIT and is working at the E-Business Technology Institute of the University of Hong Kong as a Project Manager. His research interests include information security, cryptography, and e-commerce.

Xiaodong Lin obtained his Ph.D. degree from Beijing University of Posts and Telecommunications in 1998. He subsequently spent two years at the University of Waterloo as a postdoctoral fellow. He is currently a senior security architect at Intellitactics, Inc., Canada. Dr. Lin has published more than 20 papers in journals and conferences. His research interests include network security (particularly enterprise security management, intrusion detection, performance analysis, vulnerability and exploit analysis, and penetration testing), applied cryptography, data mining, and distributed systems.

Lev Mirlas graduated in Engineering Science from the Faculty of Applied Science and Engineering at the University of Toronto in 1989, and obtained his Master's degree in Computer Engineering from the same university in 1995. He is a senior engineer at the IBM Canada Toronto Laboratory, where he has worked in the areas of trusted distributed computing and electronic commerce, including electronic procurement, insurance industry information exchange, and B2B commerce. He is a Registered Professional Engineer in the Province of Ontario.

Yi Mu received his Ph.D. from the Australian National University in 1994. Upon completion of his Ph.D., he took up a research associate position in the Centre for Computer Security Research, University of Wollongong, Australia. In 1995, Dr. Mu joined the Distributed Systems Security Research Unit in the School of Computing and IT at the University of Western Sydney (UWS), Australia, as a Postdoctoral Research Fellow. He became an Associate Lecturer in 1996 and then a Lecturer in the School of Computing and IT at UWS. He joined the Department of Computing, Macquarie University, Australia, as a Senior Lecturer, in 2001. His current research interests include electronic commerce, mobile security, access control, mobile agents, and cryptography. Dr. Mu has over 50 research publications in international journals and refereed conference proceedings. He is a regular reviewer for some major international journals and conferences. He has been a member of the Program Committee for many international conferences.

Khanh Quoc Nguyen received his Ph.D. in Secure Electronic Commerce from the University of Western Sydney, Australia, in 2000. He worked as a security engineer at the Motorola research laboratory in Australia for two years before moving to the security lab of Gemplus in Singapore in 2001. His research interests are mainly in the fields of electronic commerce security, smart-card security, and public-key cryptography. He has published a number of research papers in electronic commerce security.

Simpson Poon is Professor, Chair of Information Systems at Charles Sturt University, Australia. He has been a visiting lecturer at the University of Hong Kong. Dr. Poon earned his Ph.D. in Information Systems from Monash University, Australia. He was the Founding Director of the Centre of E-Commerce and Internet Studies at Murdoch University, Australia. Dr. Poon has been an e-business consultant and has worked with both government and business organizations in Australia and Asia. He has published widely in the area of e-business in both academic and professional journals. Dr. Poon can be reached at spoon@csu.edu.au.

Vijay Varadharajan is currently the Microsoft Chair Professor at Macquarie University, Australia. He did a Ph.D. in Computer and Communication Security in the UK and has been working on various aspects of security technology over the last

19 years. He has done research in formal security models, security in distributed systems and networks, security policies, design and analysis of security protocols, design of security architectures, cryptography, secure electronic payment systems, and mobile networks security. His research work has contributed to the development of several secure systems in the commercial arena in the areas of secure distributed authentication, DCE security, distributed authorization and authorization servers, secure mobile systems, secure portable information appliances, auditing management tools for networked systems, LAN and SMDS secure network systems, secure distributed applications, and smart card systems. Prof. Varadharajan has published over 160 papers for international journals and conferences on various aspects of security technology and the applications mentioned above. He has also co-authored a book on network security and has co-edited three books on information security and one on distributed systems.

Yumin Wang is Professor at the School of Communications Engineering, Xidian University, Xi'an, P.R. China. Since the 1960s, he has conducted research in the areas of information theory, information security, and cryptology. He is a Fellow of the Chinese Institute of Electronics and the Chinese Institute of Communications. He has published several books and over 100 papers on information theory, information security, and e-business.

Johnny W. Wong received his Ph.D. degree in Computer Science from the University of California at Los Angeles in 1975. Since then, he has been with the University of Waterloo where he is currently a Professor of Computer Science. From 1989 to 1994 he was Associate Provost, Computing and Information Systems. Dr. Wong has published over 100 technical papers in the areas of information delivery systems, network resource management, performance evaluation, and distributed systems. Among his many professional roles are Editor of Wide Area Networks of IEEE Transactions on Communications (1989 to 1992), member of the Editorial Board of Performance Evaluation (1986 to 1993), member of the Editorial Board of IEEE/ACM Transactions on Networking (1997 to 2000), Technical Program Chair of IEEE INFOCOM'84, and General Chair of the 1999 International Conference on Network Protocols.

Bo Yang received his B.S. degree from Beijing University in 1986, and M.S. and Ph.D. degrees from Xidian University in 1993 and 1999, respectively. Dr. Yang is currently Associate Professor at the School of Communication Engineering, Deputy Director at the Key Lab of the Ministry of Education of China for Networking and Information Security, and Associate Dean of the School of Information Engineering in Xidian University, Shaanxi Province, P.R. China. He is a Senior Member of the Chinese Institute of Electronics (CIE), and a member of the specialist group on computer network and information security in Shaanxi Province. His research interests include information theory and e-commerce. He can be reached by email at yangbo@mail.xidian.edu.cn.

Li Yu graduated from Heilongjiang University in 1998. She received her M.S. degree in optics from Harbin Institute of Technology, Department of Physics in 2000. She is currently pursuing her Ph.D. degree in Harbin Institute of Technology, Department of Computer Science and Technology. Her research interests include image processing, biometrics, and pattern recognition. She can be reached at lyu@mbox.hit.edu.cn.

David Zhang graduated in Computer Science from Beijing University in 1974 and received his M.Sc. and Ph.D. degrees in Computer Science and Engineering from Harbin Institute of Technology (HIT) in 1983 and 1985, respectively. From 1986 to 1988, he was a postdoctoral fellow at Tsinghua University and became an Associate Professor at Academia Sinica, Beijing, China. He received his second Ph.D. in Electrical and Computer Engineering at the University of Waterloo, Ontario, Canada, in 1994. Currently, he is a Professor in the Polytechnic University of Hong Kong. He is a Founder and Director of both Biometrics Research Center in the Polytechnic University of Hong Kong and the Harbin Institute of Technology, supported by UGC/CRC, the Hong Kong Government, and the National Nature Scientific Foundation (NSFC) of China, respectively. In addition, he is a Founder and Editor-in-Chief of the International Journal of Image and Graphics, and an Associate Editor of the IEEE Trans. on Systems, Man and Cybernetics, Pattern Recognition, the International Journal of Pattern Recognition and Artificial Intelligence, Information: International Journal, the International Journal of Robotics, and Automation and Neural, Parallel and Scientific Computations. So far, he has published over 180 articles and seven books on his research areas. He can be reached at csdzhang@comp.polyu.edu.hk.

Fangguo Zhang received his B.Sc. degree from the Mathematics Department of Yantai Normal University, Shandong, China, in 1996, his M.S. degree from the Applied Mathematics Department, Tong Ji University, Shanghai, China, in 1999, and his Ph.D. degree in Cryptography from Xidian University, Shaanxi, China, in 2002. He is presently a Postdoctoral Fellow at the Cryptology and Information Security Lab, in the Information and Communications University (ICU), Taejon, Korea. His research interests are elliptic curve cryptography, hyperelliptic curve cryptography, and secure electronic commerce. He can be reached at zhfg@icu.ac.kr or fgzh@hotmail.com.

Dong Zheng received his M.S. degree in Mathematics from Shaanxi Normal University in 1985 and Ph.D. degree in Cryptography from Xidian University in 1999. From 1999 to 2001, he was a Postdoctoral Researcher at the Department of Computer Science and Engineering of Shanghai Jiao Tong University, where he is currently an Associate Professor of Computer Science. Dr. Zheng has published over 40 technical papers in the areas of mathematics, cryptography, and information security. His current research interests include cryptography, network security, and e-commerce. He can be reached at zheng-dong@cs.sjtu.edu.cn.

Index

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

NOTICE OF ALLOWANCE AND FEE(S) DUE

24341 7590 09/16/2022
Morgan, Lewis & Bockius LLP (PA)
1400 Page Mill Road
Palo Alto, CA 94304-1124

EXAMINER	
POINVIL, FRANTZY	
ART UNIT	PAPER NUMBER
3698	

DATE MAILED: 09/16/2022

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
17/654,732	03/14/2022	Paresh K. Patel	104402-5065-US	5715

TITLE OF INVENTION: METHOD AND SYSTEM FOR PRESENTING REPRESENTATIONS OF PAYMENT ACCEPTING UNIT EVENTS

APPLN. TYPE	ENTITY STATUS	ISSUE FEE DUE	PUBLICATION FEE DUE	PREV. PAID ISSUE FEE	TOTAL FEE(S) DUE	DATE DUE
nonprovisional	SMALL	$600	$0.00	$0.00	$600	12/16/2022

THE APPLICATION IDENTIFIED ABOVE HAS BEEN EXAMINED AND IS ALLOWED FOR ISSUANCE AS A PATENT. PROSECUTION ON THE MERITS IS CLOSED. THIS NOTICE OF ALLOWANCE IS NOT A GRANT OF PATENT RIGHTS. THIS APPLICATION IS SUBJECT TO WITHDRAWAL FROM ISSUE AT THE INITIATIVE OF THE OFFICE OR UPON PETITION BY THE APPLICANT. SEE 37 CFR 1.313 AND MPEP 1308.

THE ISSUE FEE AND PUBLICATION FEE (IF REQUIRED) MUST BE PAID WITHIN THREE MONTHS FROM THE MAILING DATE OF THIS NOTICE OR THIS APPLICATION SHALL BE REGARDED AS ABANDONED. THIS STATUTORY PERIOD CANNOT BE EXTENDED. SEE 35 U.S.C. 151. THE ISSUE FEE DUE INDICATED ABOVE DOES NOT REFLECT A CREDIT FOR ANY PREVIOUSLY PAID ISSUE FEE IN THIS APPLICATION. IF AN ISSUE FEE HAS PREVIOUSLY BEEN PAID IN THIS APPLICATION (AS SHOWN ABOVE), THE RETURN OF PART B OF THIS FORM WILL BE CONSIDERED A REQUEST TO REAPPLY THE PREVIOUSLY PAID ISSUE FEE TOWARD THE ISSUE FEE NOW DUE.

HOW TO REPLY TO THIS NOTICE:

I. Review the ENTITY STATUS shown above. If the ENTITY STATUS is shown as SMALL or MICRO, verify whether entitlement to that entity status still applies.

If the ENTITY STATUS is the same as shown above, pay the TOTAL FEE(S) DUE shown above.

If the ENTITY STATUS is changed from that shown above, on PART B - FEE(S) TRANSMITTAL, complete section number 5 titled "Change in Entity Status (from status indicated above)".

For purposes of this notice, small entity fees are 1/2 the amount of undiscounted fees, and micro entity fees are 1/2 the amount of small entity fees.

II. PART B - FEE(S) TRANSMITTAL, or its equivalent, must be completed and returned to the United States Patent and Trademark Office (USPTO) with your ISSUE FEE and PUBLICATION FEE (if required). If you are charging the fee(s) to your deposit account, section "4b" of Part B - Fee(s) Transmittal should be completed. If an equivalent of Part B is filed, a request to reapply a previously paid issue fee must be clearly made, and delays in processing may occur due to the difficulty in recognizing the paper as an equivalent of Part B.

III. All communications regarding this application must give the application number. Please direct all communications prior to issuance to Mail Stop ISSUE FEE unless advised to the contrary.

IMPORTANT REMINDER: Maintenance fees are due in utility patents issuing on applications filed on or after Dec. 12, 1980. It is patentee's responsibility to ensure timely payment of maintenance fees when due. More information is available at www.uspto.gov/PatentMaintenanceFees.

PART B - FEE(S) TRANSMITTAL

Complete and send this form, together with applicable fee(s), by mail or fax, or via EFS-Web.

By mail, send to:
Mail Stop ISSUE FEE
Commissioner for Patents
P.O. Box 1450
Alexandria, Virginia 22313-1450

By fax, send to: (571)-273-2885

INSTRUCTIONS: This form should be used for transmitting the ISSUE FEE and PUBLICATION FEE (if required). Blocks 1 through 5 should be completed where appropriate. All further correspondence including the Patent, advance orders and notification of maintenance fees will be mailed to the current correspondence address as indicated unless corrected below or directed otherwise in Block 1, by (a) specifying a new correspondence address; and/or (b) indicating a separate "FEE ADDRESS" for maintenance fee notifications.

CURRENT CORRESPONDENCE ADDRESS (Note: Use Block 1 for any change of address)

24341 7590 09/16/2022

Morgan, Lewis & Bockius LLP (PA)
1400 Page Mill Road
Palo Alto, CA 94304-1124

Note: A certificate of mailing can only be used for domestic mailings of the Fee(s) Transmittal. This certificate cannot be used for any other accompanying papers. Each additional paper, such as an assignment or formal drawing, must have its own certificate of mailing or transmission.

Certificate of Mailing or Transmission

I hereby certify that this Fee(s) Transmittal is being deposited with the United States Postal Service with sufficient postage for first class mail in an envelope addressed to the Mail Stop ISSUE FEE address above, or being transmitted to the USPTO via EFS-Web or by facsimile to (571) 273-2885, on the date below.

(Typed or printed name)

(Signature)

(Date)

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
17/654,732	03/14/2022	Paresh K. Patel	104402-5065-US	5715

TITLE OF INVENTION: METHOD AND SYSTEM FOR PRESENTING REPRESENTATIONS OF PAYMENT ACCEPTING UNIT EVENTS

APPLN. TYPE	ENTITY STATUS	ISSUE FEE DUE	PUBLICATION FEE DUE	PREV. PAID ISSUE FEE	TOTAL FEE(S) DUE	DATE DUE
nonprovisional	SMALL	$600	$0.00	$0.00	$600	12/16/2022

EXAMINER	ART UNIT	CLASS-SUBCLASS
POINVIL, FRANTZY	3698	705-044000

1. Change of correspondence address or indication of "Fee Address" (37 CFR 1.363).

□ Change of correspondence address (or Change of Correspondence Address form PTO/AIA/122 or PTO/SB/122) attached.

□ "Fee Address" indication (or "Fee Address" Indication form PTO/AIA/47 or PTO/SB/47; Rev 03-02 or more recent) attached. **Use of a Customer Number is required.**

2. For printing on the patent front page, list
(1) The names of up to 3 registered patent attorneys or agents OR, alternatively,
(2) The name of a single firm (having as a member a registered attorney or agent) and the names of up to 2 registered patent attorneys or agents. If no name is listed, no name will be printed.

1____________________________

2____________________________

3____________________________

3. ASSIGNEE NAME AND RESIDENCE DATA TO BE PRINTED ON THE PATENT (print or type)

PLEASE NOTE: Unless an assignee is identified below, no assignee data will appear on the patent. If an assignee is identified below, the document must have been previously recorded, or filed for recordation, as set forth in 37 CFR 3.11 and 37 CFR 3.81(a). Completion of this form is NOT a substitute for filing an assignment.

(A) NAME OF ASSIGNEE

(B) RESIDENCE: (CITY and STATE OR COUNTRY)

Please check the appropriate assignee category or categories (will not be printed on the patent) : □ Individual □ Corporation or other private group entity □ Government

4a. Fees submitted: □Issue Fee □Publication Fee (if required) □Advance Order - # of Copies ______________________

4b. Method of Payment: *(Please first reapply any previously paid fee shown above)*

□ Electronic Payment via EFS-Web □ Enclosed check □ Non-electronic payment by credit card (Attach form PTO-2038)

□ The Director is hereby authorized to charge the required fee(s), any deficiency, or credit any overpayment to Deposit Account No. __________

5. **Change in Entity Status** (from status indicated above)

□ Applicant certifying micro entity status. See 37 CFR 1.29

□ Applicant asserting small entity status. See 37 CFR 1.27

□ Applicant changing to regular undiscounted fee status.

NOTE: Absent a valid certification of Micro Entity Status (see forms PTO/SB/15A and 15B), issue fee payment in the micro entity amount will not be accepted at the risk of application abandonment.
NOTE: If the application was previously under micro entity status, checking this box will be taken to be a notification of loss of entitlement to micro entity status.
NOTE: Checking this box will be taken to be a notification of loss of entitlement to small or micro entity status, as applicable.

NOTE: This form must be signed in accordance with 37 CFR 1.31 and 1.33. See 37 CFR 1.4 for signature requirements and certifications.

Authorized Signature ____ Date ____________________________________

Typed or printed name ____ Registration No. _____________________________

UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
17/654,732	03/14/2022	Paresh K. Patel	104402-5065-US	5715

24341 7590 09/16/2022

Morgan, Lewis & Bockius LLP (PA)
1400 Page Mill Road
Palo Alto, CA 94304-1124

EXAMINER	
POINVIL, FRANTZY	
ART UNIT	**PAPER NUMBER**
3698	

DATE MAILED: 09/16/2022

Determination of Patent Term Adjustment under 35 U.S.C. 154 (b)

(Applications filed on or after May 29, 2000)

The Office has discontinued providing a Patent Term Adjustment (PTA) calculation with the Notice of Allowance.

Section 1(h)(2) of the AIA Technical Corrections Act amended 35 U.S.C. 154(b)(3)(B)(i) to eliminate the requirement that the Office provide a patent term adjustment determination with the notice of allowance. See Revisions to Patent Term Adjustment, 78 Fed. Reg. 19416, 19417 (Apr. 1, 2013). Therefore, the Office is no longer providing an initial patent term adjustment determination with the notice of allowance. The Office will continue to provide a patent term adjustment determination with the Issue Notification Letter that is mailed to applicant approximately three weeks prior to the issue date of the patent, and will include the patent term adjustment on the patent. Any request for reconsideration of the patent term adjustment determination (or reinstatement of patent term adjustment) should follow the process outlined in 37 CFR 1.705.

Any questions regarding the Patent Term Extension or Adjustment determination should be directed to the Office of Patent Legal Administration at (571)-272-7702. Questions relating to issue and publication fee payments should be directed to the Customer Service Center of the Office of Patent Publication at 1-(888)-786-0101 or (571)-272-4200.

OMB Clearance and PRA Burden Statement for PTOL-85 Part B

The Paperwork Reduction Act (PRA) of 1995 requires Federal agencies to obtain Office of Management and Budget approval before requesting most types of information from the public. When OMB approves an agency request to collect information from the public, OMB (i) provides a valid OMB Control Number and expiration date for the agency to display on the instrument that will be used to collect the information and (ii) requires the agency to inform the public about the OMB Control Number's legal significance in accordance with 5 CFR 1320.5(b).

The information collected by PTOL-85 Part B is required by 37 CFR 1.311. The information is required to obtain or retain a benefit by the public which is to file (and by the USPTO to process) an application. Confidentiality is governed by 35 U.S.C. 122 and 37 CFR 1.14. This collection is estimated to take 30 minutes to complete, including gathering, preparing, and submitting the completed application form to the USPTO. Time will vary depending upon the individual case. Any comments on the amount of time you require to complete this form and/or suggestions for reducing this burden, should be sent to the Chief Information Officer, U.S. Patent and Trademark Office, U.S. Department of Commerce, P.O. Box 1450, Alexandria, Virginia 22313-1450. DO NOT SEND FEES OR COMPLETED FORMS TO THIS ADDRESS. SEND TO: Commissioner for Patents, P.O. Box 1450, Alexandria, Virginia 22313-1450. Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it displays a valid OMB control number.

Privacy Act Statement

The Privacy Act of 1974 (P.L. 93-579) requires that you be given certain information in connection with your submission of the attached form related to a patent application or patent. Accordingly, pursuant to the requirements of the Act, please be advised that: (1) the general authority for the collection of this information is 35 U.S.C. 2(b)(2); (2) furnishing of the information solicited is voluntary; and (3) the principal purpose for which the information is used by the U.S. Patent and Trademark Office is to process and/or examine your submission related to a patent application or patent. If you do not furnish the requested information, the U.S. Patent and Trademark Office may not be able to process and/or examine your submission, which may result in termination of proceedings or abandonment of the application or expiration of the patent.

The information provided by you in this form will be subject to the following routine uses:

1. The information on this form will be treated confidentially to the extent allowed under the Freedom of Information Act (5 U.S.C. 552) and the Privacy Act (5 U.S.C 552a). Records from this system of records may be disclosed to the Department of Justice to determine whether disclosure of these records is required by the Freedom of Information Act.
2. A record from this system of records may be disclosed, as a routine use, in the course of presenting evidence to a court, magistrate, or administrative tribunal, including disclosures to opposing counsel in the course of settlement negotiations.
3. A record in this system of records may be disclosed, as a routine use, to a Member of Congress submitting a request involving an individual, to whom the record pertains, when the individual has requested assistance from the Member with respect to the subject matter of the record.
4. A record in this system of records may be disclosed, as a routine use, to a contractor of the Agency having need for the information in order to perform a contract. Recipients of information shall be required to comply with the requirements of the Privacy Act of 1974, as amended, pursuant to 5 U.S.C. 552a(m).
5. A record related to an International Application filed under the Patent Cooperation Treaty in this system of records may be disclosed, as a routine use, to the International Bureau of the World Intellectual Property Organization, pursuant to the Patent Cooperation Treaty.
6. A record in this system of records may be disclosed, as a routine use, to another federal agency for purposes of National Security review (35 U.S.C. 181) and for review pursuant to the Atomic Energy Act (42 U.S.C. 218(c)).
7. A record from this system of records may be disclosed, as a routine use, to the Administrator, General Services, or his/her designee, during an inspection of records conducted by GSA as part of that agency's responsibility to recommend improvements in records management practices and programs, under authority of 44 U.S.C. 2904 and 2906. Such disclosure shall be made in accordance with the GSA regulations governing inspection of records for this purpose, and any other relevant (i.e., GSA or Commerce) directive. Such disclosure shall not be used to make determinations about individuals.
8. A record from this system of records may be disclosed, as a routine use, to the public after either publication of the application pursuant to 35 U.S.C. 122(b) or issuance of a patent pursuant to 35 U.S.C. 151. Further, a record may be disclosed, subject to the limitations of 37 CFR 1.14, as a routine use, to the public if the record was filed in an application which became abandoned or in which the proceedings were terminated and which application is referenced by either a published application, an application open to public inspection or an issued patent.
9. A record from this system of records may be disclosed, as a routine use, to a Federal, State, or local law enforcement agency, if the USPTO becomes aware of a violation or potential violation of law or regulation.

Notice of Allowability	**Application No.** 17/654,732	**Applicant(s)** Patel, Paresh K.	
	Examiner FRANTZY POINVIL	**Art Unit** 3698	**AIA (FITF) Status** Yes

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address--

All claims being allowable, PROSECUTION ON THE MERITS IS (OR REMAINS) CLOSED in this application. If not included herewith (or previously mailed), a Notice of Allowance (PTOL-85) or other appropriate communication will be mailed in due course. **THIS NOTICE OF ALLOWABILITY IS NOT A GRANT OF PATENT RIGHTS.** This application is subject to withdrawal from issue at the initiative of the Office or upon petition by the applicant. See 37 CFR 1.313 and MPEP 1308.

1.☑ This communication is responsive to the response filed 8/19/2022.
 □ A declaration(s)/affidavit(s) under **37 CFR 1.130(b)** was/were filed on _____.

2.□ An election was made by the applicant in response to a restriction requirement set forth during the interview on _____; the restriction requirement and election have been incorporated into this action.

3.☑ The allowed claim(s) is/are 1-20 . As a result of the allowed claim(s), you may be eligible to benefit from the **Patent Prosecution Highway** program at a participating intellectual property office for the corresponding application. For more information, please see **http://www.uspto.gov/patents/init_events/pph/index.jsp** or send an inquiry to **PPHfeedback@uspto.gov.**

4.□ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

Certified copies:

a) □All b) □ Some* c) □ None of the:

1. □ Certified copies of the priority documents have been received.
2. □ Certified copies of the priority documents have been received in Application No. _____ .
3. □ Copies of the certified copies of the priority documents have been received in this national stage application from the International Bureau (PCT Rule 17.2(a)).

* Certified copies not received: _____ .

Applicant has THREE MONTHS FROM THE "MAILING DATE" of this communication to file a reply complying with the requirements noted below. Failure to timely comply will result in ABANDONMENT of this application.
THIS THREE-MONTH PERIOD IS NOT EXTENDABLE.

5.□ CORRECTED DRAWINGS (as "replacement sheets") must be submitted.
 □ including changes required by the attached Examiner's Amendment / Comment or in the Office action of Paper No./Mail Date _____ .

Identifying indicia such as the application number (see 37 CFR 1.84(c)) should be written on the drawings in the front (not the back) of each sheet. Replacement sheet(s) should be labeled as such in the header according to 37 CFR 1.121(d).

6.□ DEPOSIT OF and/or INFORMATION about the deposit of BIOLOGICAL MATERIAL must be submitted. Note the attached Examiner's comment regarding REQUIREMENT FOR THE DEPOSIT OF BIOLOGICAL MATERIAL.

Attachment(s)

1.☑ Notice of References Cited (PTO-892)
2.☑ Information Disclosure Statements (PTO/SB/08), Paper No./Mail Date 8/21/2022.
3.□ Examiner's Comment Regarding Requirement for Deposit of Biological Material _____.
4.□ Interview Summary (PTO-413), Paper No./Mail Date. _____.

5. □ Examiner's Amendment/Comment
6. ☑ Examiner's Statement of Reasons for Allowance
7. □ Other _____.

/FRANTZY POINVIL/
Primary Examiner, Art Unit 3698

DETAILED ACTION

Notice of Pre-AIA or AIA Status

1. The present application, filed on or after March 16, 2013, is being examined under the first inventor to file provisions of the AIA.

Allowable Subject

2. The following is an examiner's statement of reasons for allowance:

Claims 1-20 are allowable over the art of record.

The prior art taken alone or in combination failed to teach or suggest:

"identifying one or more payment accepting units in proximity to the mobile device that are available to accept payment from a mobile payment application executing on the mobile device, the identifying based at least in part on an identifier corresponding to the one or more payment accepting units, wherein the one or more payment accepting units are payment operated machines that accept payment for dispensing of products and/or services, and displaying a user interface of the mobile payment application on the display of the mobile device, the user interface being configured to display a visual indication of the one or more payment accepting units and accept user input to (i) receive selection by a user of the mobile device of an available payment accepting unit of the one or more payment accepting units and (ii) trigger payment by the mobile payment application for a transaction initiated by the user of the mobile device with the available payment accepting unit of the one or more payment accepting units", as recited in independent claims 1, 13 and 15.

3. The prior following prior art taken alone or in combination fails to teach or suggest the above noted limitations.

Mei (US 20190236586 A1) discloses a payment processing method and apparatus, where the payment processing method includes obtaining by a payee terminal, transaction information of a payment card application, determining, by the payee terminal based on the transaction information, that a running environment of the payment card application is a mobile terminal device; and sending, by the payee terminal, payment voucher information of the payment card application to the mobile terminal device using a first communications technology, where the first communications technology includes a short range communications technology. Hence, a user can conveniently manage and view payment voucher information.

Xu et al (US 20160132870 A1) disclose a method implemented at a server to facilitate secure offline transactions. The server receives, from a client device, an authorization request that includes a user identifier, first financial account information and a secure code. The server authenticates the authorization request, and sends a first transaction approval to the client device. Then, in accordance with the information received in the authorization request, the server facilitates a secure transaction between the client device and a point-of-sale (POS) machine while the client device is offline. Specifically, the server receives, from the POS machine, a transaction request that includes at least the user identifier and the security code. The server retrieves the first financial account information from a memory according to the user identifier and the security code, performs a transaction operation associated with the first financial account information, and sends a second transaction approval to the POS machine.

The above recited limitations provide meaningful limitations that transforms the abstract idea into patent eligible. The claim as a whole effects an improvement to another technology or technical field. These limitations in combination provide meaningful limitations beyond generally linking the use of the abstract idea to a practical application.

Conclusion

4. Any inquiry concerning this communication or earlier communications from the examiner should be directed to FRANTZY POINVIL whose telephone number is (571)272-6797. The examiner can normally be reached M-Th 7:00AM to 5:30PM.

Examiner interviews are available via telephone, in-person, and video conferencing using a USPTO supplied web-based collaboration tool. To schedule an interview, applicant is encouraged to use the USPTO Automated Interview Request (AIR) at http://www.uspto.gov/interviewpractice.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Michael Anderson can be reached on 571-270-0508. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of published or unpublished applications may be obtained from Patent Center. Unpublished application information in Patent Center is available to registered users. To file and manage patent submissions in Patent Center, visit: https://patentcenter.uspto.gov. Visit https://www.uspto.gov/patents/apply/patent-center for more information about Patent Center and https://www.uspto.gov/patents/docx for information about filing in DOCX format. For additional questions, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

/fp/

/FRANTZY POINVIL/
Primary Examiner, Art Unit 3698

August 31, 2022

Notice of References Cited	Application/Control No. 17/654,732	Applicant(s)/Patent Under Reexamination Patel, Paresh K.	
	Examiner FRANTZY POINVIL	Art Unit 3698	Page 1 of 1

U.S. PATENT DOCUMENTS

*		Document Number Country Code-Number-Kind Code	Date MM-YYYY	Name	CPC Classification	US Classification
*	A	US-20190236586-A1	08-2019	Mei; Jingqing	G06Q20/401	1/1
*	B	US-20140337235-A1	11-2014	VAN HEERDEN; Lauren	G06Q20/383	705/71
*	C	US-20080040265-A1	02-2008	Rackley III; Brady Lee	G06Q20/102	705/40
*	D	US-11074577-B1	07-2021	Soccorsy; Benjamin	G06Q20/385	1/1
*	E	US-20130282590-A1	10-2013	Rajarethnam; Rajeshwar	G06Q20/3276	705/71
*	F	US-20120203666-A1	08-2012	Torossian; Arthur	G06Q20/027	705/26.41
*	G	US-20080010193-A1	01-2008	Rackley III; Brady Lee	G06Q20/325	705/39
*	H	US-20160132870-A1	05-2016	Xu; Jiajie	G06Q20/382	705/21
	I					
	J					
	K					
	L					
	M					

FOREIGN PATENT DOCUMENTS

*		Document Number Country Code-Number-Kind Code	Date MM-YYYY	Country	Name	CPC Classification
	N					
	O					
	P					
	Q					
	R					
	S					
	T					

NON-PATENT DOCUMENTS

*		Include as applicable: Author, Title Date, Publisher, Edition or Volume, Pertinent Pages)
	U	Hoffman et al., "New options in Wireless payments", Internet World 7.7:37 Penton Media Inc., Penton Business Media, Inc. and their subsidiaries . (Year: 2001).
	V	Carton et al., "Framework for Mobile Payments Integration', Electronic Journal of Information Systems Evaluation, 15.1: 13-24, Academic Conferences International Limited, January. (Year: 2012).
	W	
	X	

*A copy of this reference is not being furnished with this Office action. (See MPEP § 707.05(a).)
Dates in MM-YYYY format are publication dates. Classifications may be US or foreign.

Innovative VDI Standards: Moving an Industry Forward

Michael L. Kasavana, Ph.D., NAMA Professor in Hospitality Business,
Michigan State University, USA

ABSTRACT

Effective self-service automation requires data sharing among non-proprietary system components. Historically, the original self-service provider, the vending industry, has failed to develop and implement open architecture standards that enable seamless data sharing among disparate devices and servers in a network. This lack of interoperability among the main entities in a sophisticated vending system (e.g. vending machine controller, telemetry device, vending management software, and cashless payment equipment) has contributed to a lack of innovation and creativity in the evolution of unattended point-of-sale retailing. Recently crafted Vending Data Interchange (VDI) standards from the National Automatic Merchandising Association (NAMA) provide a non-proprietary means by which to share machine-level data among diverse technology providers. The VDI standards are designed to ensure reliability, continuity, and longevity among installed hardware, software, and netware. In essence, VDI standards contain technical specifications that bundle vending machine-level data for easy distribution throughout a vending operator's technology network and can be implemented by a qualified provider without operator involvement.

Keywords: *data interchange, vending, self-service technologies, unattended point-of-sale*

INTRODUCTION

The NAMA Technology Leadership Committee, under the direction of the NAMA Board of Directors, formed a specialized technology task force charged with developing data interchange standards that could be implemented industry-wide. The task force was directed to develop a set of data transport protocols enabling the sharing of vending machine data among competing back-end technology providers. These standards had to ensure reliability, continuity, and longevity. Reliability relating to each participating technology provider of a vending operator receiving identical data files, continuity in terms of data retrieval and distribution throughout a vending operator's network, and longevity by providing assurance to vending operators that interfaces between installed applications would remain viable going forward. As a result, the task force produced NAMA VDI (**V**ending **D**ata **I**nterchange) standards. These standards contain technical specifications that bundle vending machine-level data for rapid distribution throughout a vending operator's technology network and can be implemented by technology providers without vending operator intervention.

Simply stated, vending operators desire technology capable of reliably passing data sets from one application service provider to another so that multiple application service providers can contribute to a single networked solution. The essence of the NAMA VDI standards is to enable data movement through a messaging technique that ensures data integrity of a transmitted set of data, regardless of whether it was pulled or pushed to a server. In other words, NAMA VDI standards render vending technology capable of linking together diverse software solutions, from different vending technology providers, into unified applications and likely represent a tipping point in the accelerated adoption of vending technologies as operator concerns related to supplier-dependence are significantly reduced.

Data Sets

NAMA VDI is an innovative set of protocols designed to package vending machine-level data (e.g. DEX and MDB data, alerts data, cashless transaction data, etc.) into a message format that can be shared among diverse supplier systems to enable multiple software applications on the identical data set. For example, consider the situation in which a telemetry provider remotely polls DEX data from a vending machine (e.g. Company "X"). The telemetry provider transfers machine-level generated data file to its server (e.g. "X" Server). The server in turn authenticates the file with a NAMA VDI message wrapper and labels its contents for subsequent communication to any other provider's server in a vending operator's network (e.g. Company "Y" or Company "Z" etc.). Additionally, the vending operator may have machine-installed cashless readers that collecting electronic payment data for transmission to cashless gateway for reconciliation. The polled data set would consist of both DEX data and electronic payment data and packaged into an aggregated data set. Movement of the data set to a host vending management software (VMS) system capable of processing DEX data could occur while simultaneously forwarded data to a cashless gateway system could be applied for processing and settlement. This multiple tasking one a single data set is indicative of the robust nature of VDI messaging.

The functionality of VDI standards is somewhat analogous to an email communication in that the file of machine captured data file forms the content of the message while VDI programming places a wrapper, akin to an email message envelope that enables distribution among any number of file servers (e.g. email recipients), regardless of supplier, provider, or manufacturer. NAMA VDI standards, for example, allow for DEX data to be transmitted by a telemetry device or server in real time. This approach provides a platform for a vending operator's VMS to upload data nightly for use in pre-packaging (also referred to as pre-kitting) and/or dynamic scheduling algorithms that rely on variable replenishment strategies. The goal of NAMA VDI standards is to ensure that a vending operator can confidently implement multiple, diverse vending technology solutions while utilizing operational data in existing application software (regardless of supplier). NAMA VDI specifications are open architecture technology standards designed to be extensible, uniform, stable, and manufacturer neutral.

Vending Technology

More than two decades ago, in an effort to standardize the control of machine-level transaction and event data collection, storage, and transmission technical specification committee members of the National Automatic Merchandizing Association (NAMA)[i] and the European Vending Association (EVA) collaborated to develop a set of protocols necessary for efficient data handling and processing. [i]One of the outcomes of this effort is DEX data. DEX, which is an acronym for Data EXchange standard, is capable of capturing machine-level cash in/out data, product movement data, and financial audit data. DEX data is designed to assist operators with product replenishment strategies, product mix rotations, and cash management safeguards. In order to optimize contribution margins, while controlling operating expenses, DEX data plays an important role in productivity and profitability analysis. Accompanying the advent of DEX, a Data Transfer Standard (DTS) was devised so that the DEX data could be exported from the machine in a decipherable electronic format. Once the data was transmitted, it could be entered into a vending management software system (VMS) and used in combination with product mappings to evaluate route coverage, cash handling procedures, and sales performance. It is for this reason that the DTS protocol is often considered an integral part of the DEX standard; not a separate element.

More recently, the Multi-Drop Bus (MDB) protocol emerged is an internal communication protocol designed to ensure that coin mechanisms, bill validators, and cashless payment devices could be effectively interfaced to a vending machine controller (VMC) without regard to proprietary manufacturing specifications. MDB, often compared to USB standards used in generic computer component interfacing, replaced prior practices built on supplier-specific design connectivity. An MDB cable (also termed a machine harness) provides the physical connectivity for attaching peripheral devices (e.g. card reader, bill validator, etc.) to the VMC of a vending machine. MDB is credited with initiating the movement toward open system architecture in vending technology. Since vending machine-level data capture involves the retrieval of stored audit information (akin to a snapshot) via local or remote transfer, there is a need to apply data in various ways to produce a comprehensive analysis of transactions. In fact, some telemetry providers actively monitor the MDB bus to detect, in real-time, product movement and operational alerts (e.g. bill jam, change shortage, door open, temperature variance, etc.).

As a result of prior developments, vending machine-level data formatting and content derivation conforms to the European Vending Association-Data Transfer Standard (EVA-DTS) and provides access to system status data, transactional data, and machine configuration information. In a typical data connection, a polling device actively surveys the vending machine for stored data then follows DTS standards for transmission to an external device. Once the data transfer is complete, the received vending machine-level data can be wrapped in a NAMA VDI message format for subsequent distribution to installed vending system servers (see Figure 1).

DEX		**DEX**		**DEX**		**NAMA**
Data	>	**Data**	>	**Data**	>	**VDI**
Collection		**Extraction**		**Transmission**		**Standard**

Figure 1. Machine Data Transmission to NAMA VDI Application

MDB Standards

A more recently developed vending machine technology standard than DEX is the multi-drop bus (MDB) standard. Vending machines have one master communication channel and it is labeled the vending machine controller (VMC). The role of the VMC is to define the functionality of peripheral equipment (coin changer, bill validator, card reader, etc.) that must be interfaced with the electronic circuitry of the vender to work properly. MDB is the short form of multi-drop bus/internal communication protocol (MDB/ICP). MDB/ICP is an open global standard governing the interface between a vending machine controller and payment system peripheral devices and is maintained by NAMA and the European Vending Association (EVA). The MDB standard defines a serial bus interface for electronically controlled vending machines. It also standardizes vending machines that employ electronic controls so that all vending and peripheral equipment communicate identically.

Basically, MDB defines and performs as the serial bus interface for electronically controlled vending machines. It also standardizes processes for vending machine interoperability and communication among various peripheral components (i.e. coordination and validation of peripheral equipment interactivity). The serial bus, or MDB, is typically configured as a master-slave arrangement enabling a single master the capability of communicating with up to thirty-two peripheral devices. The VMC serves as the system master unit. The purpose of MDB is to ensure that the necessary functionality of any device on the bus (i.e. interfaced peripheral equipment) is compatible with the capabilities of the VMC. Hence, the software employed by both the peripherals and VMC must be compatible, but not necessarily at the same level of capability in order to establish peripheral functionality. Within the MDB standard, the capability of each peripheral device is designated by a classification Level identifier. Levels of peripheral functionality were established in response to the addition of major extension in the capability of add-on devices. Level designations are used to avoid potential conflicts that may arise when a VMC Level and a peripheral Level are incompatible. Should this occur, neither the VMC nor the peripheral will be able to issue or reply to a command not supported by the device.

For example, connecting a Level 2 MDB peripheral device to a Level 1 VMC creates a condition of incompatibility that prohibits the proper functioning of the peripheral device (e.g. payment acceptor). The current level for the MDB standard is referred to as Version 3/Level 3 (first introduced March 2003) that pioneered the incorporation of features associated with cashless transactions. In essence, the VMC must initially determine the Level of a peripheral device in order to determine the command set to communicate with the device. A VMC can only issue commands that are supported by the peripheral. For example, a Level 3 command may only be issued for a Level 3 or higher peripheral and will not work with a command issued for a Level 1 or 2 peripheral. A cashless payment peripheral, for instance, deciphers the Level of a VMC through a setup command designed to establish compatibility and functionality.

A compilation of component Levels indicates varying functionality among installed peripherals. For each classification of peripheral device there are a set of mandatory requirements that equipment developers must adhere to ensure compatibility and Level designation. It is a fundamental principle of the MDB standard that all peripheral devices must be implementable with both backward and forward compatibility relative to a machine VMC. The VMC typically gravitates to the highest common Level among peripheral devices.

MDB Interfaces

MDB/ICP enables the VMC to determine what coins the coin changer, and what bills a bill validator, can accept as payment. Additionally, MDB/ICP establishes the amount of credit available through a payment card reader. Through coordination of components, the VMC is able to manage and direct the coin changer in determining how much change to pay out; bills to recycle, or outstanding credit balance to return to the card. There are additional peripheral devices, beyond coin changers, bill acceptors, and card readers, like paykey and other closed systems, for which the VMC can be configured via an MDB interface. Historically, the manufacturer of components being placed into a vending machine had to manually define technical functionality among machine components. Since each component manufacturer acted independently, a number of proprietary device interfaces emerged. The problem with a proprietary interface is that its uniqueness adds unnecessary costs and complexity to vending machine configuration and operation. The MDB/IPC standard was adopted to establish a communication method that allowed the devices in a vending machine to use a common interface. Despite the fact several devices can tie into the same MDB interface, each will operate independently on the interface. Since each interfaced device is assigned a unique address, the VMC can determine which device is active and communicating.

A majority of vending machines support the MDB/IPC standard and thereby are capable of allowing a vending operator to choose payment and other devices primarily based on reliability, performance, and price. Since the MDB/IPC standard establishes the manner by which each component device communicates with the VMC, the connection to each device tends to be identical. Every device has basically two MDB/IPC connectors to allow it to both connect to the MDB/IPC bus in the machine while providing a linked connection to another device (if needed). This design reduces the number of

MDB connectors needed as well as allowing for additional devices. Hence, adding an additional peripheral device to a vending machine is simplified since the requisite hardware and bus connectors to add the device are already in the machine. The MDB/IPC is an internationally supported interface. Through the cooperative efforts of the National Automatic Merchandising Association (NAMA) in the U.S. and the European Vending Association (EVA) and the European Vending Machine Manufacturers Association (EVMMA) in Europe, the standard was developed with provisions for varying currency acceptance and payment technologies.

MBD and DEX

Technically, the MDB/IPC standard defines a serial master-slave communication bus used by the internal devices in the machine, like the coin acceptor. MDB allows for instantaneous updating of the current status of the machine (i.e. data changes as each product is sold). It is for this reason that the MDB standard is considered a transaction-based mechanism, unlike DEX, which is a cumulative-based reporting system. The MDB protocol allows for the attachment of an audit (DEX) device that, acting as a passive slave, receives information of all events that happened on the machine (e.g. vends, sold outs, coins and bills accepted, etc.). On the other hand, DEX involves the retrieval of stored information (a snapshot) through a serial plug designed for connectivity with a handheld terminal (HHT) or small PC. The connection conforms to the EVA-DTS standard and provides access to status data, testing routines, and machine setup. In a DEX connection, the connected device actively polls the machine for stored information.

Cashless transactions are not dependent on DEX but rely on MDB processes. The fundamental difference between DEX and MDB is that MDB is the only method for a bill acceptor or a coin changer to report credit deposited to authorize a vending transaction. DEX cannot do this. This fact makes it necessary to have MDB installed; DEX, while needed for sales reporting, is not mandatory for the machine to operate. Hence, from a cashless payment perspective, MDB is more useful than DEX since it details the transaction (card number, transaction value, product(s) sold, date, and time) for reconciliation. The results of the transaction will be posted as an MDB record. For operators not employing cashless vending, DEX data is often sufficient to provide necessary information for a vending management system. It is for this reason, some vending operators only use MDB for cashless transactions, and ignore DEX data. For those operators desiring DEX data, a DEX cable can be used to transfer the DEX file along with the cashless MDB data.

DEX is the key to technological advancements in the vending industry worldwide as it enables data capture at the point of purchase. DEX has earned international recognition and support and can be used to facilitate consistent data formatting throughout the vending channel. In the past, machine manufacturers varied in how data exchange transmissions occurred. In response, DEX designers and equipment engineers have established standards governing data recordation, file formatting, and file exportation through common interface linkages. As a consequence, vending machines are manufactured as DEX-enabled and are often labeled "DEX-compliant." From a sales perspective, DEX provides the vending operator the ability to track brand and/or product preferences at the point of purchase. DEX has been found to improve sales performance, reduce operating expenses, and minimize machine malfunctions. In addition, DEX enables space to sales analysis, for machine-level column allocation optimization, in vending management software. This is an important outcome of a DEX-compliant device. The main benefit of line item tracking is accountability and machine plan-o-gram (i.e. rotating menu of product offering) development.

The fact that vending equipment tends to be strategically placed in disparate locations presents a challenge to efficient replenishment, sales analyses, malfunction notification, and comprehensive audit reporting. Fortunately, machine-level transactional data can be captured through an electronic control board installed within each vending machine. Aggregating machine-level data enables remote review of transactions and inventory without having to have a physical presence at the machine. The fact data can be exported to a remote warehouse, central office, or product fulfillment center extends the opportunity for more thorough, immediate, and frequent analysis. A majority of v-commerce applications are the result of DEX implementation.

In the past, machine manufacturers varied in how data exchanges and transmissions occurred. Recently released DEX software (Edition 6 and higher) tightens the specifications of the protocol to prevent possible misinterpretations in accountability or brand identification. Since there has been a proliferation of diverse vending products, and several variations in the packaging of the same product, the DEX standard has been refined to acknowledge and differentiate between product offerings. While not all vending operators demand identical informational output, vending machine circuit boards are built to possess similar data collection capabilities to ensure the delivery of consistent content. For example, three data elements referenced in the DEX standard are: 1) number of bills held in the bill stacker, 2) quantity and denomination of coins stored in the coin box, and 3) number of vends or products sold.

A DEX-compliant machine relies upon DEX architecture to enable vending machine polling. The vending machine exports its unique identification number and stored data to an external system for analysis and processing. An optional element of this data stream is the machine's service history, including the last date the machine was serviced. Once DEX data is exchanged with a vending management system various transaction audits can be performed. Since captured data is not accessible or editable prior to interfacing to an auxiliary system, cash accountability will be accurate and complete. Also, the ability to track product information at the machine-level enhances productivity, as machine fulfillment is improved and manual data entry eliminated. The DEX protocol enables different makes and models of vending machines to communicate in a consistent manner. DEX data sets include sales mix, cash collection, product movement, and malfunction alerts. Additionally, DEX specifications may soon include a standard for reporting error codes for payment validation, dispensing jams, and other operational problems. Proposed specifications are pending approval.

Since vending machines have an average life of ten years, it may take a generation of new machine installations to fully realize the DEX potential. Basically, DEX provides an indisputable, auditable accounting method for cash collections, units sold, and product price recordation that capable of enhancing route efficiency and improving warehouse operations. For example, how much cash should be in a machine at the close of a sales period? A route driver, unable to view the DEX electronic record, will have cash collections compared against the machine-level electronic record. Balancing cash against collections provides management with a unique level of information and control.

DEX Polling

NAMA and the European Vending Association (EVA) have jointly adopted a communication protocol for the electronic retrieval of machine-level information via data polling. As a consequence, vending machines are now manufactured as DEX-enabled. Each vending machine is given a unique identifying number by which the DEX data extracted is labeled. During a polling session, this unique number and the date and time that the service occurred, are transmitted to the polling device. DEX data is polled an audit can be performed. Since captured data is not accessible or editable by the route driver, cash accountability is assumed accurate and complete. Also, the ability to track product information at the machine level enhances productivity as route time is improved and manual data entry is eliminated. DEX specifies a data format to enable all different types of machines and machine models to communicate electronically in a similar manner. The DEX information available includes: sales, cash collections, product movement and other vending machine activities. Additionally, the DEX specification contains a standard for reporting error codes for payment validation, jams, and other operational problems. Line item tracking is important to both accountability and assistance in future machine menu development. DEX data retrieval can be accomplished via three distinct polling modes: 1-local polling, 2-dial-up polling or 3-wireless polling.

Local Polling – local polling incorporates a hand-held device (or pocket probe) designed to plug connect to a vending machine's DEX-port or to communicate through an IR port. Once the connection is established, the device is used to extract (upload) transactional data from the machine to the handheld device. A typically DEX data upload takes approximately five seconds. Field collected data can be transferred from the handheld device to a central office computer (downloaded) for processing, analysis, and report generation.

Dial-up Polling – dial-up polling involves use of a modem and telephone line. Once a valid connection is established, DEX data can be transported to a remote office or warehouse location for evaluation over an Internet or virtual private network (VPN) connection. This design enables the machine to be remotely monitored with respect to cash, inventory, and machine malfunctions.

Wireless Polling – similar to dial-up polling, wireless polling enables remote access to DEX data via a cellular network. Wireless polling however relies upon cellular network connectivity to establish the proper linkage. The advancement of wireless technology has emerged as an attractive alternative. Wireless applications possess tremendous potential for the vending industry, an industry that desires mobility, flexibility, and reliability in enterprise-wide operations. Vending practitioners dissatisfied with the constraints and complexities of hard wiring are migrating to the convenience of design portability and user mobility that wireless technology solutions provide.

Common network connectivity options include both the Internet and virtual private networks (VPN). Cellular connectivity presents challenges based on the architectural structure surrounding the vending equipment combined with strength of signal requirements. While connectivity to a VPN tends to be more direct and less susceptible to structural infringements, it is likely to be more costly. Historically, vending operators have benefiting from such devices as hand-held terminals, personal digital assistants, smart paging units, global positioning systems, telecommunication links (telemetry), proximity transponders, and related applications.

VDI Standards

The purpose of the NAMA VDI standard is to establish transparent, non-proprietary interfaces that enable transportation of data among the main components of a vending system (e.g. vending machine, telemetry system, cashless payment system, specialty applications, and vending management software). The non-proprietary nature of NAMA VDI renders it an open standard. NAMA VDI relies on messaging standards to satisfy data interchange needs and is not concerned with the entity transmitting or receiving such messages. For example, a messaging standard governing the transmission of machine-level DEX data may originate from the vending machine, an advanced telemetry device, or the file server of another entity. NAMA VDI mandates that the message format conform to the technical specifications of the standard, regardless of the entity creating the message.

VDI Messaging

The NAMA VDI Task Force has identified the following seven elements as important to vending data messaging (interchangeable/exchangeable data files):

1- DEX data messaging – sent or requested captured DEX data file
2- Alert data messaging – may originate from the VMC, DEX, or MDB depending on telemetry provider.
3- Device Status -- device configuration and/or service request
4- Device Configuration – sent device configuration and/or status reporting
5- Security Authorization – defines cooperative agreement partners
6- Machine Message – reconfigures machine to EVA standards
7- Device Messaging - provides confirmation of download instructions

Cooperative Relationships

The NAMA VDI standard incorporates 'cooperative agreements' among competing vending technology suppliers so that interchanged data will be more meaningfully consumed and effectively applied. Cooperative agreements, often referred to as trading partner agreements, involve written documentation that informs both sender (producer) and receiver (consumer) of NAMA VDI messages the specifics of the message(s) being shared. Descriptive elements include such items as: company profile, security authorization, machine identification, location identification, and type of connectivity (server, web-service, email, etc.). For example, if Provider X is to pass a NAMA VDI data message to Provider Y then the cooperating parties must have transaction information to successfully distribute and utilize the desired data messages. User names, passwords, and web-based SSL encryption also can be used to help insure data transfers are secure and accessible by authorized entities.

Early Adopters

Seven major vending technology providers volunteered to implement and fine tune the VDI standards prior to release of Version 1.0. Cantaloupe Systems, CompuVend, Crane Merchandising, InOne Technology, MEI Group, USA Technologies, and Validata worked cooperatively to field test and validate VDI specifications and procedures. This is similar to the replacement of specialized railroad car connectors with non-specialized couplers that enable assembly of cars in any order or sequence (see Figure 2).

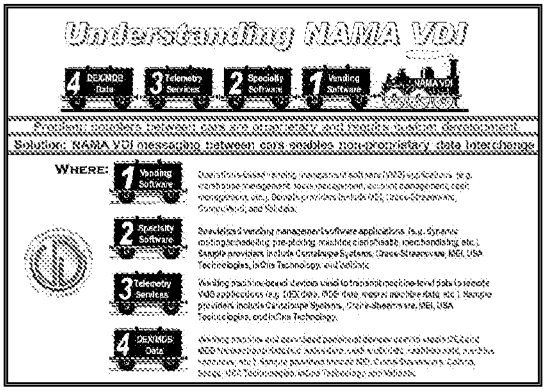

Understanding NAMA VDI
4 DEX/MDB Data
3 Telemetry Services
2 Specialty Software
1 Vending Software
NAMA VDI
Problem: couplers between cars are proprietary and require custom development
Solution: NAMA VDI messaging between cars enables non-proprietary data interchange
WHERE:
1 Vending Software
Operations-based vending management software (VMS) applications (e.g. warehouse management, route management, account management, cash management, etc.). Sample providers include MEI, Crane-Streamware, CompuVend, and Validata.
2 Specialty Software
Specialized vending management software applications (e.g. dynamic routing/scheduling, pre-picking, machine alerts/health, merchandising, etc.). Sample providers include Cantaloupe Systems, Crane-Streamware, MEI, USA Technologies, InOne Technology, and Validata.
3 Telemetry Services
Vending machine-based devices used to transmit machine-level data to remote VMS applications (e.g. DEX data, MDB data, machine data, etc.). Sample providers include Cantaloupe Systems, Crane-Streamware, MEI, USA Technologies, and InOne Technology.
4 DEX/MDB Data

Figure 2. Understanding NAMA VDI Standards

The NAMA VDI standards are being released beginning in first quarter 2010 and are planned to have additional features added and released in sequential stages as developments are completed. Release

1.0 addresses the most critical area among data transactions: DEX messaging. It is anticipated that as the remaining data messages are developed, they will be released and labeled in ascending numerical order (e.g. Release 1.1, 1.2, and so forth).

VDI Benefits

The NAMA VDI standards afford several direct benefits to operators and bottlers, especially those embarking on technology decisions. When purchasing vending technology from a company adhering to NAMA VDI standards, the buyer can be assured that:

1. investment in the compliant technology will be compatible across major suppliers
2. there is no longer a need to rely on the success of a single supplier
3. multiple telemetry devices will work with a variety of VMS providers
4. selling or acquiring VDI compatible components simplifies continued operations

See Figure Three for an illustration of the NAMA VDI standards as applied to data for processing by a vending management software (VMS) program as well as transactional data for processing via a cashless gateway.

SUMMARY

Major vending technology providers participating in NAMA VDI development have created a tipping point for accelerating the implementation of vending technology. Increased interest in addressing operator concerns has resulted in an unprecedented cooperation among vending technology suppliers to enable harmonic data interchange. NAMA VDI ensures that operators can feel confident in technology investment, choice of suppliers, and be assured that hardware and software will work together now and in the future. There has never been a better or safer time to invest in cashless vending, remote machine monitoring, or VMS technology.

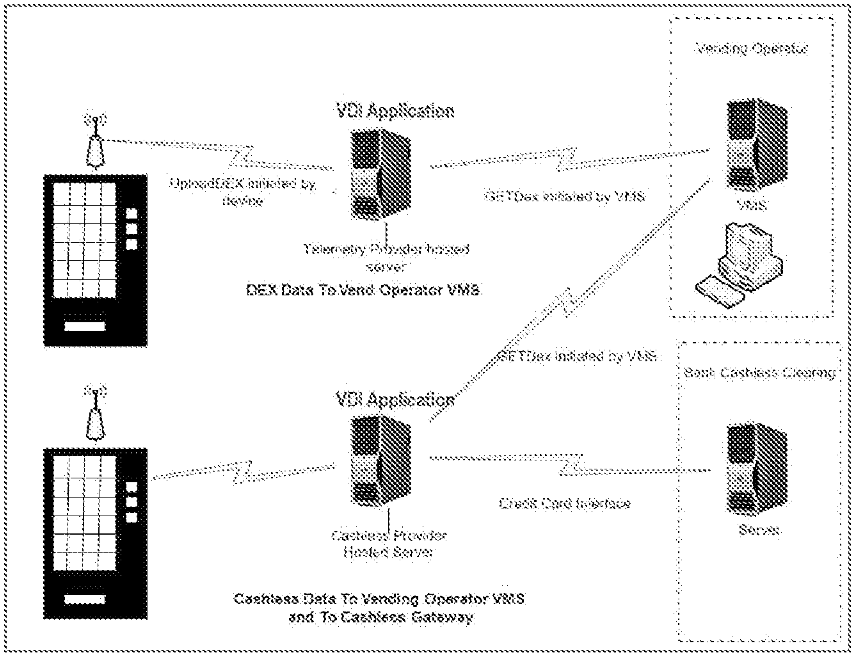

Vending Operator
VDI Application
UploadDEX initiated by device
GETDex initiated by VMS
VMS
Telemetry Provider hosted server
DEX Data To Vend Operator VMS
GETDex initiated by VMS
Bank Cashless Clearing
VDI Application
Credit Card Interface
Server
Cashless Provider Hosted Server
Cashless Data To Vending Operator VMS and To Cashless Gateway

Figure 3. DEX Data transported via telemetry to multiple provider servers reporting into a VMS

REFERENCES

European Vending Association, DEX Standards V 6.1, www.vending-europe.eu/en/standards/eva-dts.html

Kasavana, Michael L. (2006). Understanding MDB, DEX, and DTS, NAMA White Paper (2) 5-17

Kasavana, Michael L., V-Commerce: Understanding Vending Machine Terminology, HFTP Bottomline Magazine, www.hospitalitynet.org/news/4011592.html

MDB/ICP, Version 4.0, April 2009, www.vending.org/technology/MDB_Version_4.pdf

NAMA Vending Data Interchange Standards, www.vending.org/technology/VDI_Standards.pdf

National Automatic Merchandising Association, MDB Standards V 4.0, www.vending.org/technology/MDB_Version_4.pdf

iPhone
User Guide

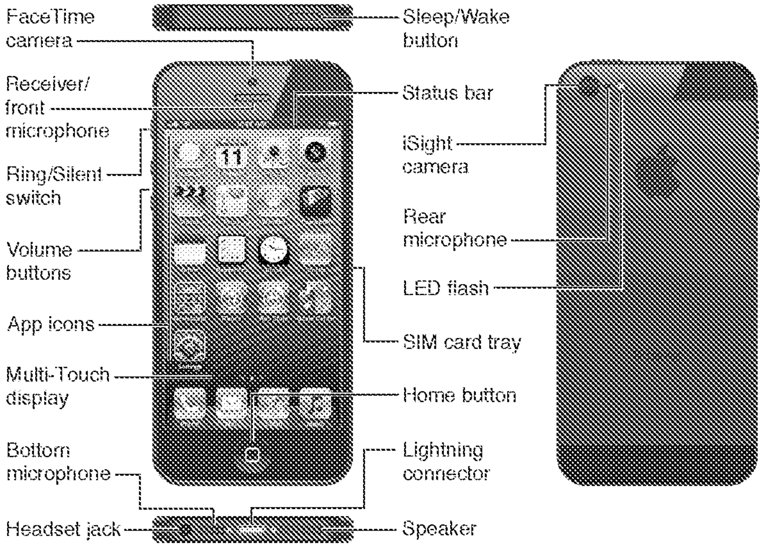
For iOS 6.1 Software

Contents

iPhone at a Glance

1

iPhone 5 overview

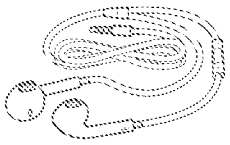


iPhone apps and features may vary based on your location, language, carrier, and model of iPhone. To find out which features are supported in your area, see www.apple.com/ios/feature-availability.

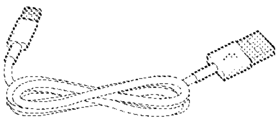
Note: Apps that send or receive data over a cellular network may incur additional fees. Contact your carrier for information about your iPhone service plan and fees.

Accessories

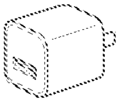
The following accessories are included with iPhone:



Apple headset: Use the Apple EarPods with Remote and Mic (iPhone 5) or the Apple Earphones with Remote and Mic (iPhone 4S or earlier) to listen to music and videos, and make phone calls. See Apple headset on page 31.



Connecting cable: Use the Lightning to USB Cable (iPhone 5) or the 30-pin to USB Cable (iPhone 4S or earlier) to connect iPhone to your computer to sync and charge.



Apple USB power adapter: Use with the Lightning to USB Cable or 30-pin to USB Cable to charge the iPhone battery.

SIM eject tool: Use to eject the SIM card tray. (Not included in all areas.)

Buttons

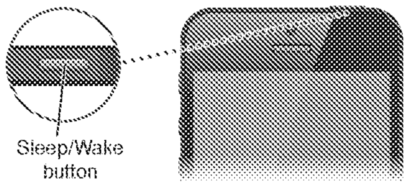
Sleep/Wake button

When you're not using iPhone, you can lock it to turn off the display and save the battery.

Lock iPhone: Press the Sleep/Wake button.

When iPhone is locked, nothing happens if you touch the screen. iPhone can still receive calls, text messages, and other updates. You can also:

- Listen to music
- Adjust the volume
- Use the headset to answer a call or listen to music



Unlock iPhone: Press the Sleep/Wake button or the Home button □, then drag the slider.

Turn iPhone off: Press and hold the Sleep/Wake button for a few seconds until the red slider appears, then drag the slider.

Turn iPhone on: Press and hold the Sleep/Wake button until the Apple logo appears.

Open Camera when iPhone is locked: Press the Sleep/Wake button or the Home button □, then drag 📷 up.

iPhone automatically locks if you don't touch the screen for a minute or so.

Adjust the auto-lock timing or turn it off: See Auto-Lock on page 136.

Require a passcode to unlock iPhone: See Passcode Lock on page 136.

Home button

The Home button □ takes you to the Home screen, no matter what you're doing. It also provides other convenient shortcuts.

Go to the Home screen: Press the Home button □.

On the Home screen, tap an app to open it. See Opening and switching between apps on page 17.

Display recently used apps: With iPhone unlocked, double-click the Home button □. The multitasking bar appears at the bottom of the screen, showing the most recently used apps. Swipe the bar to the left to see more apps.

Display audio playback controls:

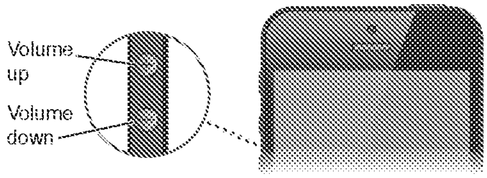
- *When iPhone is locked:* Double-click the Home button □. See Playing music on page 58.
- *When you're using another app:* Double-click the Home button □, then swipe the multitasking bar from left to right.

Use Siri (iPhone 4S or later) or Voice Control: Press and hold the Home button □. See Chapter 4, Siri, on page 36 and Voice Control on page 26.

Volume controls

While you're on the phone or listening to songs, movies, or other media, the buttons on the side of iPhone adjust the audio volume. Otherwise, the buttons control the volume for the ringer, alerts, and other sound effects.

> *WARNING:* For important information about avoiding hearing loss, see Important safety information on page 146.



Lock the ringer and alerts volume: Go to Settings > Sounds and turn off "Change with Buttons."

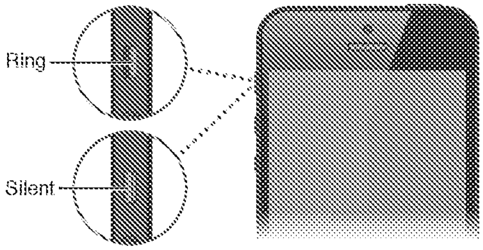
Limit the volume for music and videos: Go to Settings > Music > Volume Limit.

Note: In some countries, iPhone may indicate when you're setting the volume above the European Union hearing safety guidelines. To increase the volume beyond this level, you may need to briefly release the volume control.

You can also use either volume button to take a picture or record a video. See Chapter 12, Camera, on page 74.

Ring/Silent switch

Flip the Ring/Silent switch to put iPhone in ring mode 🔔 or silent mode 🔕.



In ring mode, iPhone plays all sounds. In silent mode, iPhone doesn't ring or play alerts and other sound effects.

Important: Clock alarms, audio apps such as Music, and many games still play sounds through the built-in speaker when iPhone is in silent mode. In some areas, the sound effects for Camera and Voice Memos are played even if the Ring/Silent switch is set to silent.

For information about changing sound and vibrate settings, see Sounds on page 139.

You can also use the Do Not Disturb setting to silence calls, alerts, and notifications.

Set iPhone to Do Not Disturb (☾): Go to Settings and turn on Do Not Disturb. Do Not Disturb keeps calls, alerts, and notifications from making any sounds or lighting up the screen when the screen is locked. Alarms still sound, however, and if the screen is unlocked, Do Not Disturb has no effect.

To schedule quiet hours, allow certain people to call, or enable repeated calls to ring through, go to Settings > Notifications > Do Not Disturb. See Do Not Disturb and Notifications on page 132.

Status icons

The icons in the status bar at the top of the screen give information about iPhone:

Status icon		What it means
▂▄▆	**Cell signal***	Shows whether you're in range of the cellular network and can make and receive calls. The more bars, the stronger the signal. If there's no signal, the bars are replaced with "No service."
✈	**Airplane mode**	Shows that airplane mode is on—you cannot use the phone, access the Internet, or use Bluetooth® devices. Non-wireless features are available. See Airplane mode on page 130.
LTE	LTE	Shows that your carrier's LTE network is available, and iPhone can connect to the Internet over that network. (iPhone 5. Not available in all areas.) See Cellular on page 135.
4G	UMTS	Shows that your carrier's 4G UMTS (GSM) network is available, and iPhone can connect to the Internet over that network. (iPhone 4S or later. Not available in all areas.) See Cellular on page 135.
3G	UMTS/EV-DO	Shows that your carrier's 3G UMTS (GSM) or EV-DO (CDMA) network is available, and iPhone can connect to the Internet over that network. See Cellular on page 135.
E	EDGE	Shows that your carrier's EDGE (GSM) network is available, and iPhone can connect to the Internet over that network. See Cellular on page 135.

Status icon		What it means
○	GPRS/1xRTT	Shows that your carrier's GPRS (GSM) or 1xRTT (CDMA) network is available, and iPhone can connect to the Internet over that network. See Cellular on page 135.
📶	Wi-Fi*	Shows that iPhone is connected to the Internet over a Wi-Fi network. The more bars, the stronger the connection. See Wi-Fi on page 130.
☾	Do Not Disturb	Shows that "Do Not Disturb" is turned on. See Sounds on page 139.
∞	Personal Hotspot	Shows that iPhone is connected to another iPhone providing a Personal Hotspot. See Personal Hotspot on page 132.
⟳	Syncing	Shows that iPhone is syncing with iTunes.
✳	Network activity	Shows network activity. Some third-party apps may also use the icon to show an active process.
☎	Call Forwarding	Shows that Call Forwarding is set up on iPhone. See Call forwarding, call waiting, and caller ID on page 49.
VPN	VPN	Shows that you're connected to a network using VPN. See Cellular on page 135.
🔒	Lock	Shows that iPhone is locked. See Sleep/Wake button on page 8.
☎	TTY	Shows that iPhone is set to work with a TTY machine. See TTY support on page 128.
▶	Play	Shows that a song, audiobook, or podcast is playing. See Playing music on page 58.
🔒	Portrait orientation lock	Shows that the iPhone screen is locked in portrait orientation. See Portrait and landscape orientation on page 19.
⏰	Alarm	Shows that an alarm is set. See Chapter 19, Clock, on page 90.
➤	Location Services	Shows that an item is using Location Services. See Privacy on page 140.
ᛒ	Bluetooth*	*Blue or white icon:* Bluetooth is on and paired with a device. *Gray icon:* Bluetooth is on and paired with a device, but the device is out of range or turned off. *No icon:* Bluetooth is not paired with a device. See Bluetooth devices on page 32.
▭	Bluetooth battery	Shows the battery level of a supported paired Bluetooth device.
▭	Battery	Shows battery level or charging status. See Battery on page 34.

* **Accessories and wireless performance:** The use of certain accessories with iPhone may affect wireless performance. Not all iPod accessories are fully compatible with iPhone. Turning on airplane mode on iPhone may eliminate audio interference between iPhone and an accessory. While airplane mode is on, you cannot make or receive calls or use features that require wireless communication. Reorienting or relocating iPhone and the connected accessory may improve wireless performance.

Getting Started

2

⚠ *WARNING:* To avoid injury, read Important safety information on page 146 before using iPhone.

What you need

To use iPhone, you need:

- A wireless service plan with a carrier that provides iPhone service in your area
- An Internet connection for your computer (broadband is recommended)
- An Apple ID for some features, including iCloud, the App Store and iTunes Store, and online purchases. An Apple ID can be created during setup.

To use iPhone with your computer, you need:

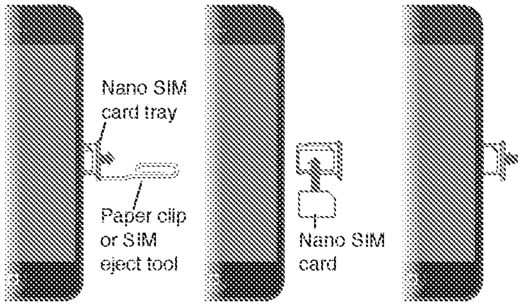
- A Mac with a USB 2.0 or 3.0 port, or a PC with a USB 2.0 port, and one of the following operating systems:
  - Mac OS X version 10.6.8 or later
  - Windows 7, Windows Vista, or Windows XP Home or Professional with Service Pack 3 or later
- iTunes 10.7 or later (for some features), available at www.itunes.com/download

Installing the SIM card

If you were given a SIM card to install, install it before setting up iPhone.

Important: A SIM card is required in order to use cellular services when connecting to GSM networks and some CDMA networks. An iPhone 4S or later that's been activated on a CDMA wireless network may also use a SIM card for connecting to a GSM network, primarily for international roaming. Your iPhone is subject to your wireless service provider's policies, which may include restrictions on switching service providers and roaming, even after conclusion of any required minimum service contract. Contact your wireless service provider for more details. Availability of cellular capabilities depends on the wireless network.

Installing the SIM Card in iPhone 5



Install the SIM card: Insert the end of a small paper clip or SIM eject tool into the hole on the SIM card tray. Pull out the SIM card tray and place the SIM card in the tray as shown. With the tray aligned and the SIM card on top, carefully replace the tray.

Setting up and activating iPhone

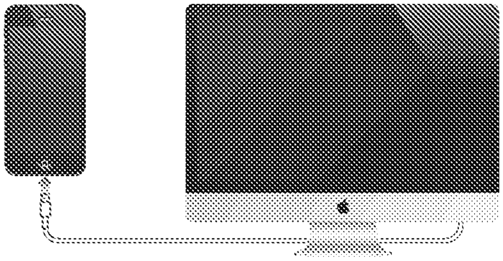
To set up and activate iPhone, turn on iPhone and follow the Setup Assistant. The Setup Assistant steps you through the setup process, including connecting to a Wi-Fi network, signing in with or creating a free Apple ID, setting up iCloud, turning on recommended features such as Location Services and Find My iPhone, and activating iPhone with your carrier. You can also restore from an iCloud or iTunes backup during setup.

Activation can be done over a Wi-Fi network or, with iPhone 4S or later, over your carrier's cellular network (not available in all areas). If neither option is available, you need to connect iPhone to your computer running iTunes for activation.

Connecting iPhone to your computer

You may need to connect iPhone to your computer in order to complete activation. Connecting iPhone to your computer also lets you sync information, music, and other content with iTunes. See Syncing with iTunes on page 16.

Connect iPhone to your computer: Use the Lightning to USB Cable (iPhone 5) or 30-pin to USB Cable (iPhone 4S or earlier) provided with iPhone.



Connecting to the Internet

iPhone connects to the Internet whenever necessary, using a Wi-Fi connection (if available) or your carrier's cellular network. For information about connecting to a Wi-Fi network, see Wi-Fi on page 130.

Note: If a Wi-Fi connection to the Internet isn't available, some iPhone apps and services may transfer data over your carrier's cellular network, which may result in additional fees. Contact your carrier for information about your cellular data plan rates. To manage cellular data usage, see Cellular on page 135.

Setting up mail and other accounts

iPhone works with iCloud, Microsoft Exchange, and many of the most popular Internet-based mail, contacts, and calendar service providers.

If you don't already have a mail account, you can set up a free iCloud account when you first set up iPhone, or later in Settings > iCloud. See iCloud on page 15.

Set up an iCloud account: Go to Settings > iCloud.

Set up some other account: Go to Settings > Mail, Contacts, Calendars.

You can add contacts using an LDAP or CardDAV account, if your company or organization supports it. See Adding contacts on page 101.

You can add calendars using a CalDAV calendar account, and you can subscribe to iCalendar (.ics) calendars or import them from Mail. See Working with multiple calendars on page 68.

Apple ID

An Apple ID is the user name for a free account that lets you access Apple services, such as the iTunes Store, the App Store, and iCloud. You need only one Apple ID for everything you do with Apple. There may be charges for services and products that you use, purchase, or rent.

If you have an Apple ID, use it when you first set up iPhone, and whenever you need to sign in to use an Apple service. If you don't already have an Apple ID, you can create one whenever you're asked to sign in.

For more information, see support.apple.com/kb/he37.

Managing content on your iOS devices

You can transfer information and files between your iOS devices and computers using either iCloud or iTunes.

- *iCloud* stores content such as music, photos, calendars, contacts, documents, and more, and wirelessly pushes it to your other iOS devices and computers, keeping everything up to date. See iCloud below.
- *iTunes* syncs music, video, photos, and more, between your computer and iPhone. Changes you make on one device are copied to the other when you sync. You can also use iTunes to copy a file to iPhone for use with an app, or to copy a document you've created on iPhone to your computer. See Syncing with iTunes on page 16.

You can use iCloud or iTunes, or both, depending on your needs. For example, you can use iCloud Photo Stream to automatically get photos you take on iPhone to your other devices, and use iTunes to sync photo albums from your computer to iPhone.

Important: Don't sync items in the Info pane of iTunes (such as contacts, calendars, and notes) and also use iCloud to keep that information up to date on your devices. Otherwise, duplicated data may result.

iCloud

iCloud stores your content, including music, photos, contacts, calendars, and supported documents. Content stored in iCloud is pushed wirelessly to your other iOS devices and computers set up with the same iCloud account.

iCloud is available on devices with iOS 5 or later, on Mac computers with OS X Lion v10.7.2 or later, and on PCs with the iCloud Control Panel for Windows (Windows Vista Service Pack 2 or Windows 7 required).

iCloud features include:

- *iTunes in the Cloud*—Download previous iTunes music and TV show purchases to iPhone for free, anytime.
- *Apps and Books*—Download previous App Store and iBookstore purchases to iPhone for free, anytime.
- *Photo Stream*—Photos you take appear on all your devices. You can also create photo streams to share with others. See Photo Stream on page 71.
- *Documents in the Cloud*—For iCloud-enabled apps, keep documents and app data up to date across all your devices.
- *Mail, Contacts, Calendars*—Keep your mail contacts, calendars, notes, and reminders up to date across all your devices.
- *Backup*—Back up iPhone to iCloud automatically when connected to power and Wi-Fi. See Backing up iPhone on page 150.
- *Find My iPhone*—Locate your iPhone on a map, display a message, play a sound, lock the screen, or remotely wipe the data. See Find My iPhone on page 34.
- *Find My Friends*—Share your location with people who are important to you. Download the free app from the App Store.
- *iTunes Match*—With an iTunes Match subscription, all your music—including music you've imported from CDs or purchased somewhere other than iTunes—appears on all of your devices and can be downloaded and played on demand. See iTunes Match on page 62.
- *iCloud Tabs*—See the webpages you have open on your other iOS devices and OS X computers. See Chapter 7, Safari, on page 55.

With iCloud, you get a free email account and 5 GB of storage for your mail, documents, and backups. Your purchased music, apps, TV shows, and books, as well as your photo streams, don't count against your free space.

Sign in or create an iCloud account, and set iCloud options: Go to Settings > iCloud.

Purchase additional iCloud storage: Go to Settings > iCloud > Storage & Backup, then tap Manage Storage. For information about purchasing iCloud storage, go to help.apple.com/icloud.

View and download previous purchases:

- *iTunes Store purchases:* Go to iTunes, tap More, then tap Purchased.
- *App Store purchases:* Go to App Store, tap Updates, then tap Purchased.
- *iBookstore purchases:* Go to iBooks, tap Store, then tap Purchased.

Turn on Automatic Downloads for music, apps, or books: Go to Settings > iTunes & App Stores.

For more information about iCloud, go to www.apple.com/icloud. For support information, go to www.apple.com/support/icloud.

Syncing with iTunes

Syncing with iTunes copies information from your computer to iPhone, and vice versa. You can sync by connecting iPhone to your computer, or you can set up iTunes to sync wirelessly with Wi-Fi. You can set iTunes to sync music, photos, videos, podcasts, apps, and more. For information about syncing iPhone with your computer, open iTunes, then choose iTunes Help from the Help menu.

Set up wireless iTunes syncing: Connect iPhone to your computer. In iTunes on the computer, select your iPhone, click Summary, then turn on "Sync with this iPhone over Wi-Fi."

When Wi-Fi syncing is turned on, iPhone syncs every day. iPhone must be connected to a power source, iPhone and your computer must both be on the same wireless network, and iTunes must be open on your computer. For more information, see iTunes Wi-Fi Sync on page 136.

Tips for syncing with iTunes

- If you use iCloud to store your contacts, calendars, bookmarks, and notes, don't also sync them to your device using iTunes.
- Purchases you make from the iTunes Store or the App Store on iPhone are synced back to your iTunes library. You can also purchase or download content and apps from the iTunes Store on your computer, and then sync them to iPhone.
- In the device's Summary pane, you can set iTunes to automatically sync when your device is attached to your computer. To temporarily override this setting, hold down Command and Option (Mac) or Shift and Control (PC) until you see iPhone appear in the iTunes window.
- In the device's Summary pane, select "Encrypt iPhone backup" if you want to encrypt the information stored on your computer when iTunes makes a backup. Encrypted backups are indicated by a lock icon 🔒, and a separate password is required to restore the backup. If you don't select this option, other passwords (such as those for mail accounts) aren't included in the backup and will have to be reentered if you use the backup to restore the device.
- In the device's Info pane, when you sync mail accounts, only the settings are transferred from your computer to iPhone. Changes you make to an email account on iPhone don't affect the account on your computer.
- In the device's Info pane, click Advanced to select options to let you *replace* the information on iPhone with the information from your computer during the next sync.
- If you listen to part of a podcast or audiobook, the place you left off is included if you sync the content with iTunes. If you started listening on iPhone, you can pick up where you left off using iTunes on your computer—or vice versa.
- In the device's Photo pane, you can sync photos and videos from a folder on your computer.

Viewing this user guide on iPhone

You can view the *iPhone User Guide* on iPhone in Safari, and in the free iBooks app.

View the user guide in Safari: Tap 📖, then tap the iPhone User Guide bookmark.

- *Add an icon for the guide to the Home screen:* Tap ↗, then tap "Add to Home Screen."
- *View the guide in a different language:* Tap "Change Language" on the main contents page.

View the user guide in iBooks: If you haven't installed iBooks, open App Store, then search for and install "iBooks." Open iBooks and tap Store. Search for "iPhone User," then select and download the guide.

For more information about iBooks, see Chapter 30, iBooks, on page 109.

Basics 3

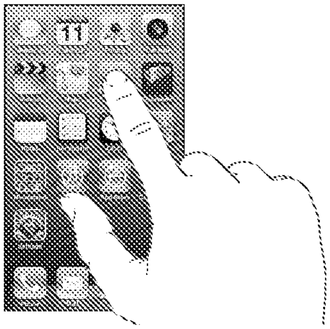
Using apps

You interact with iPhone using your fingers to tap, double-tap, swipe, and pinch objects on the touchscreen.

Opening and switching between apps

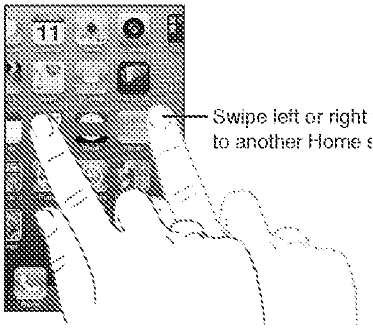
To go to the Home screen, press the Home button ▢.

Open an app: Tap it.



To return to the Home screen, press the Home button ▢ again.

See another Home screen: Swipe left or right.

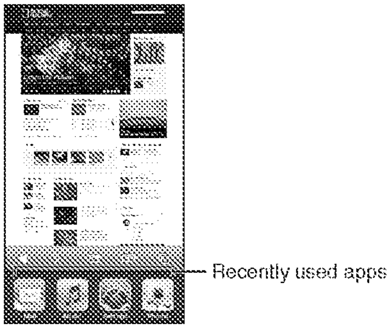


Swipe left or right to switch to another Home screen.

Go to the first Home screen: Press the Home button ▢.

View recently used apps: Double-click the Home button ▢ to reveal the multitasking bar.

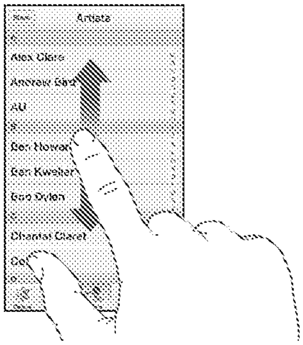
Tap an app to use it again. Swipe left to see more apps.



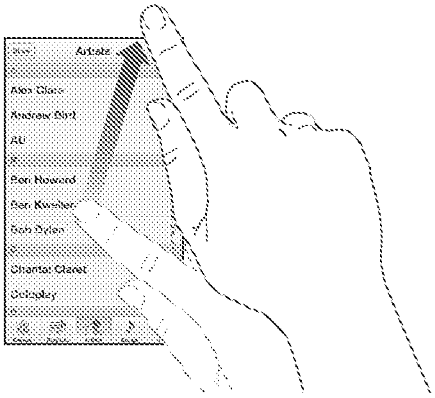
If you have a lot of apps, you might want to use Spotlight to locate and open them. See Searching on page 27.

Scrolling

Drag up or down to scroll. On some screens, such as webpages, you can also scroll side to side. Dragging your finger to scroll won't choose or activate anything on the screen.



Flick to scroll quickly.



You can wait for the scrolling to come to a stop, or touch the screen to stop it immediately.

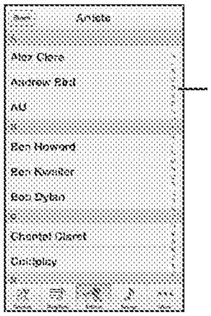
To quickly scroll to the top of a page, tap the status bar at the top of the screen.

Lists

Depending on the list, choosing an item can do different things—for example, it may open another list, play a song, open an email, or show someone's contact information.

Choose an item in a list: Tap it.

Some lists have an index along the side to help you navigate quickly.

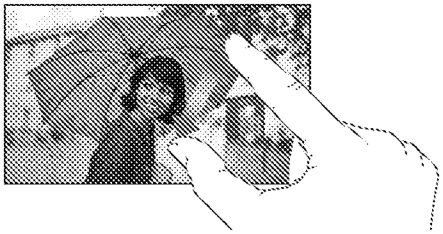


Drag your finger along the index to scroll quickly. Tap a letter to jump to a section.

Return to a previous list: Tap the back button in the upper-left corner.

Zooming in or out

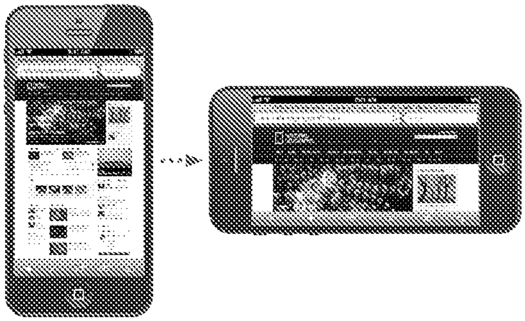
Depending on the app, you may be able to zoom in to enlarge, or zoom out to reduce the image on the screen. When viewing photos, webpages, mail, or maps, for example, pinch two fingers together to zoom out or spread them apart to zoom in. For photos and webpages, you can also double-tap (tap twice quickly) to zoom in, then double-tap again to zoom out. For maps, double-tap to zoom in and tap once with two fingers to zoom out.



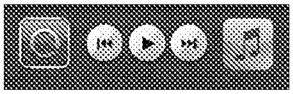
Zoom is also an accessibility feature that lets you magnify the screen with any app you're using, to help you see what's on the display. See Zoom on page 125.

Portrait and landscape orientation

You can view many iPhone apps in either portrait or landscape orientation. Rotate iPhone and the display rotates too, adjusting to fit the new orientation.



Lock the screen in portrait orientation: Double-click the Home button ▢, swipe the multitasking bar from left to right, then tap 🔒.



The orientation lock icon 🔒 appears in the status bar when the screen orientation is locked.

Adjusting brightness

You can manually adjust the brightness of the screen, or turn on Auto-Brightness to have iPhone use the built-in ambient light sensor to automatically adjust the brightness.

Adjust the screen brightness: Go to Settings > Brightness & Wallpaper, then drag the slider.

Turn Auto-Brightness on or off: Go to Settings > Brightness & Wallpaper.

See Brightness & Wallpaper on page 139.

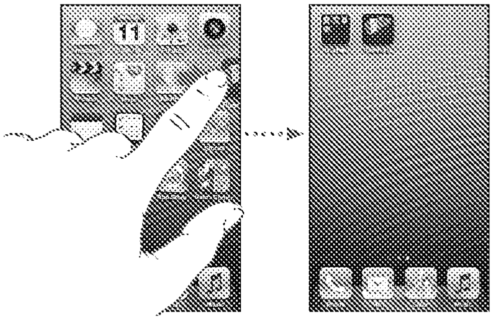
Customizing iPhone

You can customize the layout of your apps on the Home screen, organize them in folders, and change the wallpaper.

Rearranging apps

Customize your Home screen by rearranging apps, moving apps to the Dock along the bottom of the screen, and creating additional Home screens.

Rearrange apps: Touch and hold any app on the Home screen until it jiggles, then move apps around by dragging them. Press the Home button □ to save your arrangement.



Create a new Home screen: While arranging apps, drag an app to the right edge of the rightmost screen, until a new screen appears.

You can create up to 11 Home screens. The dots above the Dock show the number of screens you have, and which screen you're viewing.

Swipe left or right to switch between screens. To go to the first Home screen, press the Home button □.

Move an app to another screen: While it's jiggling, drag an app to the side of the screen.

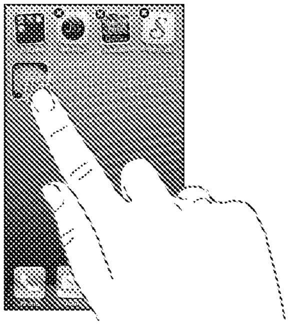
Customize the Home screen using iTunes: Connect iPhone to your computer. In iTunes on your computer, select iPhone, then click the Apps button to see the image of the iPhone Home screen.

Reset the Home screen to its original layout: In Settings, go to General > Reset, then tap Reset Home Screen Layout. Resetting the Home screen removes any folders you've created and applies the default wallpaper to your Home screen.

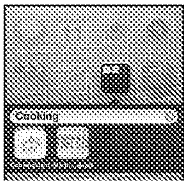
Organizing with folders

You can use folders to organize the apps on your Home screens. Rearrange folders—just as you do apps—by dragging them around your Home screens or to the Dock.

Create a folder: Touch an app until the Home screen icons begin to jiggle, then drag the app onto another.



iPhone creates a new folder that includes the two apps, and names the folder based on the type of apps. To enter a different name, tap the name field.



Open a folder: Tap the folder. To close a folder, tap outside the folder or press the Home button □.

Organize with folders: While arranging apps (the icons are jiggling):

- *Add an app to a folder:* Drag the app onto the folder.
- *Remove an app from a folder:* Open the folder if necessary, then drag the app out.
- *Delete a folder:* Move all apps out of the folder. The folder is automatically deleted.
- *Rename a folder:* Tap to open the folder, then tap the name and enter a new one.

When you finish, press the Home button □.

Changing the wallpaper

You can customize both the Lock screen and the Home screen by choosing an image or photo to use as wallpaper. Choose one of the supplied images, or a photo from your Camera Roll or another album on iPhone.

Change the wallpaper: Go to Settings > Brightness & Wallpaper.

Typing

The onscreen keyboard lets you type when you need to enter text.

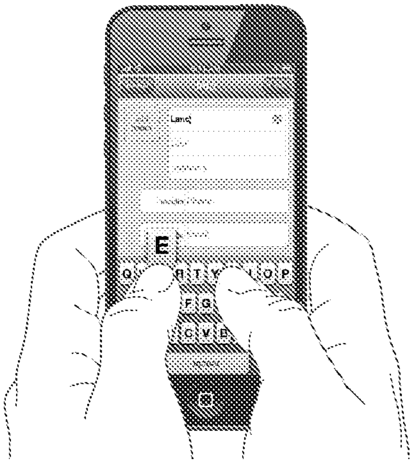
Entering text

Use the onscreen keyboard to enter text, such as contact information, mail, and web addresses. Depending on the app and the language you're using, the keyboard may correct misspellings, predict what you're typing, and even learn as you use it.

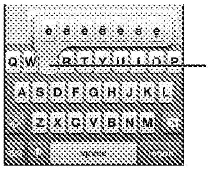
You can also use an Apple Wireless Keyboard to type. See Apple Wireless Keyboard on page 24. To use dictation instead of typing, see Dictation on page 25.

Enter text: Tap a text field to bring up the keyboard, then tap keys on the keyboard.

As you type, each letter appears above your thumb or finger. If you touch the wrong key, you can slide your finger to the correct key. The letter isn't entered until you release your finger from the key.



- *Type uppercase:* Tap the Shift key ⇧ before tapping a letter. Or touch and hold the Shift key, then slide to a letter.
- *Quickly type a period and space:* Double-tap the space bar.
- *Turn on caps lock:* Double-tap the Shift key ⇧. To turn caps lock off, tap the Shift key.
- *Enter numbers, punctuation, or symbols:* Tap the Number key 123. To see additional punctuation and symbols, tap the Symbol key #+=.
- *Enter accented letters or other alternate characters:* Touch and hold a key, then slide to choose one of the options.



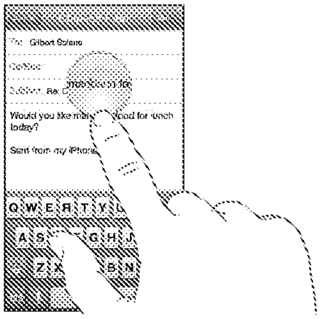
To type an alternate character, touch and hold a key, then slide to choose one of the options.

Set options for typing: Go to Settings > General > Keyboard.

Editing text

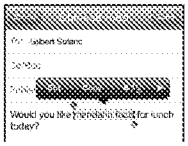
If you need to edit text, an onscreen magnifying glass lets you position the insertion point where you need it. You can select text, and cut, copy, and paste text. In some apps, you can also cut, copy, and paste photos and videos.

Position the insertion point: Touch and hold to bring up the magnifying glass, then drag to position the insertion point.



Select text: Tap the insertion point to display the selection buttons. Tap Select to select the adjacent word, or tap Select All to select all text.

You can also double-tap a word to select it. Drag the grab points to select more or less text. In read-only documents, such as webpages, touch and hold to select a word.



Cut or copy text: Select text, then tap Cut or Copy.

Paste text: Tap the insertion point, then tap Paste to insert the last text that you cut or copied. To replace text, select it before tapping Paste.

Undo the last edit: Shake iPhone, then tap Undo.

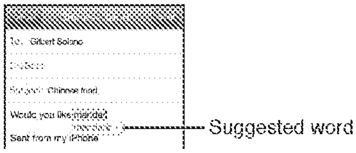
Make text bold, italic, or underlined: Select text, tap ▶, then tap B/I/U (not always available).

Get the definition of a word: Select the word, then tap Define (not always available).

Get alternative words: Select a word, then tap Suggest (not always available).

Auto-correction and spell checking

For many languages, iPhone uses the active dictionary to correct misspellings or make suggestions as you type. When iPhone suggests a word, you can accept the suggestion without interrupting your typing. For a list of supported languages, see www.apple.com/iphone/specs.html.

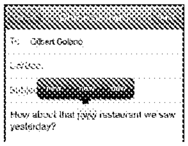


Accept the suggestion: Type a space, punctuation mark, or return character.

Reject a suggestion: Tap the "x" next to the suggestion.

Each time you reject a suggestion for the same word, iPhone becomes more likely to accept the word.

iPhone may also underline words you've already typed that might be misspelled.



To: Gilbert Golano

Cc/Bcc:

Subject:

How about that [illegible] restaurant we saw yesterday?

Replace a misspelled word: Tap the underlined word, then tap the correct spelling. If the word you want doesn't appear, just retype it.

Turn auto-correction or spell checking on or off: Go to Settings > General > Keyboard.

Shortcuts and your personal dictionary

Shortcuts lets you type just a few characters instead of a longer word or phrase. The expanded text appears whenever you type the shortcut. For example, the shortcut "omw" expands to "On my way!"

Create a shortcut: Go to Settings > General > Keyboard, then tap Add New Shortcut.

Prevent iPhone from trying to correct a word or phrase: Create a shortcut, but leave the Shortcut field blank.

Edit a shortcut: Go to Settings > General > Keyboard, then tap the shortcut.

Use iCloud to keep your personal dictionary up to date on your other iOS devices: Go to Settings > iCloud and turn on "Documents & Data."

Keyboard layouts

You can use Settings to set the layouts for the onscreen keyboard or for an Apple Wireless Keyboard that you use with iPhone. The available layouts depend on the keyboard language. See Apple Wireless Keyboard below and Appendix B, International Keyboards, on page 143.

Select keyboard layouts: Go to Settings > General > International > Keyboards, select a language, then choose the layouts.

Apple Wireless Keyboard

You can use an Apple Wireless Keyboard (available separately) for typing on iPhone. The Apple Wireless Keyboard connects via Bluetooth, so you must first pair it with iPhone. See Pairing Bluetooth devices on page 32.

Once the keyboard is paired, it connects whenever the keyboard is within range of iPhone—up to about 33 feet (10 meters). When a wireless keyboard is connected, the onscreen keyboard doesn't appear when you tap a text field. To save the battery, turn off the keyboard when not in use.

Switch the language when using a wireless keyboard: Press Command–Space bar to display a list of available languages. Press the Space bar again while holding down the Command key to choose a different language.

Turn off a wireless keyboard: Hold down the On/off switch on the keyboard until the green light goes off.

iPhone disconnects the keyboard when the keyboard is turned off or out of range.

Unpair a wireless keyboard: Go to Settings > Bluetooth, tap ⓘ next to the keyboard name, then tap "Forget this Device."

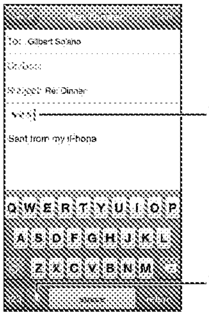
Dictation

On iPhone 4S or later, you can dictate text instead of typing. To use dictation, Siri must be turned on and iPhone must be connected to the Internet. You can include punctuation and give commands to format your text.

Note: Cellular data charges may apply.

Turn on dictation: Go to Settings > General > Siri, then turn on Siri.

Dictate text: From the onscreen keyboard, tap 🎙, then speak. When you finish, tap Done.


To: Gilbert Solano

Cc/Bcc:

Subject: Re: Dinner

Sent from my iPhone

These appear while Siri composes the text from your dictation.

Tap to begin dictation.

To add text, tap 🎙 again and continuing dictating. To insert text, tap to place the insertion point first. You can also replace selected text by dictating.

You can bring iPhone to your ear to start dictation, instead of tapping 🎙 on the keyboard. To finish, move iPhone back down in front of you.

Add punctuation or format text: Say the punctuation or formatting command.

For example, "Dear Mary comma the check is in the mail exclamation mark" results in "Dear Mary, the check is in the mail!"

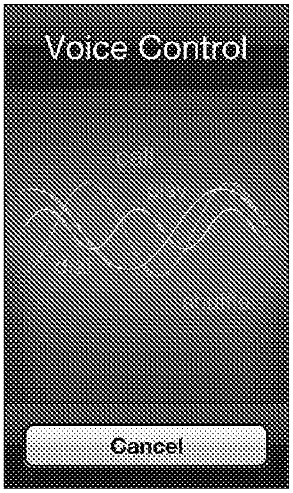
Punctuation and formatting commands include:

- quote … end quote
- new paragraph
- cap—to capitalize the next word
- caps on … caps off—to capitalize the first character of each word
- all caps—to make the next word all uppercase
- all caps on … all caps off—to make the enclosed words all uppercase
- no caps on … no caps off—to make the enclosed words all lowercase
- no space on … no space off—to run a series of words together
- smiley—to insert :-)
- frowny—to insert :-(
- winky—to insert ;-)

Voice Control

Voice Control lets you make phone calls and control music playback using voice commands. On iPhone 4S or later, you can also use Siri to control iPhone by voice. See Chapter 4, Siri, on page 36.

Note: Voice Control and Voice Control settings are not available when Siri is turned on.



Use Voice Control: Press and hold the Home button ○ until the Voice Control screen appears and you hear a beep. You can also press and hold the center button on your headset. See Apple headset on page 31.

For best results:

- Speak clearly and naturally.
- Say only iPhone commands, names, and numbers. Pause slightly between commands.
- Use full names.

Voice Control normally expects you to speak voice commands in the language that's set for iPhone (in Settings > General > International > Language). Voice Control settings let you change the language for speaking voice commands. Some languages are available in different dialects or accents.

Change the language or country: Go to Settings > General > International > Voice Control, then tap the language or country.

Voice Control for the Music app is always on, but you can prevent voice dialing when iPhone is locked.

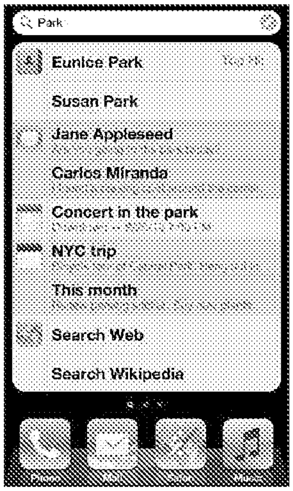
Prevent voice dialing when iPhone is locked: Go to Settings > General > Passcode Lock, then turn off Voice Dial (available only when Siri is turned off in Settings > General > Siri). To use voice dialing, you must first unlock iPhone.

For specific commands, see Making calls on page 43 and Siri and Voice Control on page 62.

For more about using Voice Control, including information about using Voice Control in different languages, go to support.apple.com/kb/HT3597.

Searching

You can search many of the apps on iPhone, as well as Wikipedia and the web. Search an individual app, or search all the apps at once using Spotlight. Spotlight also searches the names of apps on iPhone—if you have a lot of apps, you might want to use Spotlight to locate and open them.



Search an individual app: Enter text in the search field.

Search iPhone using Spotlight: Swipe right from the first Home screen, or press the Home button □ from any Home screen. Enter text in the search field.

Search results appear as you type. To dismiss the keyboard and see more results, tap Search. Tap an item in the list to open it. The icons let you know which apps the results are from.

iPhone may display a top hit for you, based on previous searches.

Spotlight searches the following:

- Contacts—All content
- Apps—Titles
- Music—Names of songs, artists, and albums, and the titles of podcasts and videos
- Podcasts—Titles
- Videos—Titles
- Audiobooks—Titles
- Notes—Text of notes
- Calendar (Events)—Event titles, invitees, locations, and notes
- Mail—To, From, and Subject fields of all accounts (the text of messages isn't searched)
- Reminders—Titles
- Messages—Names and text of messages

Search the web or Wikipedia from Spotlight: Scroll to the bottom of the search results, then tap Search Web or Search Wikipedia.

Open an app from Search: Enter all or part of the app name, then tap the app.

Choose which items are searched, and the order they're searched: Go to Settings > General > Spotlight Search.

Notifications

To help make sure you don't miss important events, many iPhone apps can provide alerts. An alert can appear briefly as a banner at the top of the screen, which goes away if you don't respond to it, or as a notice in the center of the screen that remains until you acknowledge it. Some apps can also display badges on their icons on the Home screen, to let you know how many new items await—for example, how many new email messages you have. If there's a problem—such as a message that couldn't be sent—an exclamation mark ❗ appears on the badge. A numbered badge on a folder shows the total number of alerts for all the apps in the folder.

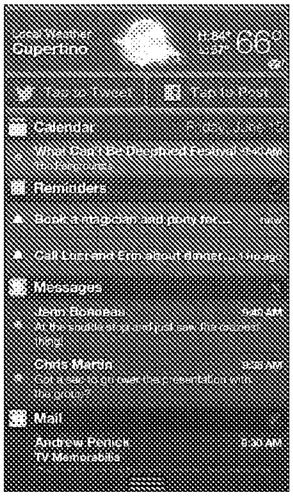
Alerts can also appear on the Lock screen.

Respond to an alert when iPhone is locked: Swipe the alert from left to right.

Notification Center displays all your alerts in one place. So if you weren't able to respond when you first received an alert, you can respond to them in Notification Center when you're ready. Alerts can include:

- Missed phone calls and voice messages
- New email
- New text messages
- Reminders
- Calendar events
- Friend requests (Game Center)

You can also get the local weather, and display your personal stock ticker. If you're signed in to your Twitter or Facebook account, you can post or tweet to your account from Notification Center.



Local Weather
Cupertino H: 84° L: 57° 66°

Calendar

What Can't Be Deepfried Festival 9:40 AM
The Fairgrounds

Reminders

Book a magician and pony for ... now

Call Lucy and Erin about dinner... 1 hr ago

Messages

Jenn Ronneau 9:40 AM
At the shuttle stop and just saw the craziest thing!

Chris Martin 9:35 AM
Got a sec to go over the presentation with the group?

Mail

Andrew Penick 8:30 AM
TV Memorabilia

View Notification Center: Swipe down from the top of the screen. Scroll the list to see additional alerts.

- *Respond to an alert:* Tap it.
- *Remove an alert:* Tap ⊗, then tap Clear.

Manage alerts for your apps: Go to Settings > Notifications. See Do Not Disturb and Notifications on page 132.

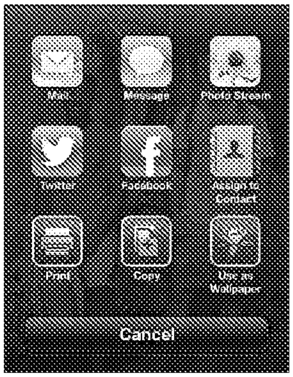
Choose alert sounds, adjust the alert volume, or turn vibrate on or off: Go to Settings > Sounds.

Sharing

iPhone gives you lots of ways to share with other people.

Sharing within apps

In many apps, tapping ↪ displays options for sharing, as well as other actions such as printing or copying. The options vary depending on the app you're using.



Facebook

Sign in to your Facebook account (or create a new account) in Settings to enable posting directly from many of the apps on iPhone.

Sign in to or create a Facebook account: Go to Settings > Facebook.

Post from Notification Center: Tap "Tap to Post."

Post using Siri: Say "Post to Facebook …."

Post an item from an app: In most apps, tap ↪. In Maps, tap ◎, tap Share Location, then tap Facebook.

Set options for Facebook: Go to Settings > Facebook to:

- Update Contacts on iPhone with Facebook names and photos
- Allow apps (such as Calendar and Contacts) to use your account

Install the Facebook app: Go to Settings > Facebook, then tap Install.

Twitter

Sign in to your Twitter account (or create a new account) in Settings to enable Tweets with attachments from many of the apps on iPhone.

Sign in to or create a Twitter account: Go to Settings > Twitter.

Tweet from Notification Center: Tap "Tap to Tweet."

Tweet using Siri: Say "Tweet …."

Tweet an item from an app: View the item, tap ↪, then tap Twitter. If ↪ isn't showing, tap the screen. To include your location, tap Add Location.

Tweet a location in Maps: Tap the location pin, tap ◎, tap Share Location, then tap Twitter.

When you're composing a Tweet, the number in the lower-right corner of the Tweet screen shows the number of characters remaining that you can enter. Attachments use some of a Tweet's 140 characters.

Add Twitter user names and photos to your contacts: Go to Settings > Twitter, then tap Update Contacts.

Install the Twitter app: Go to Settings > Twitter, then tap Install.

To learn how to use the Twitter app, open the app, tap Me, then tap Help.

Connecting iPhone to a TV or other device

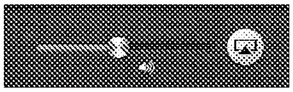
You can use AirPlay with Apple TV to stream content to an HDTV, or connect iPhone to your TV using cables.

AirPlay

With AirPlay, you can stream music, photos, and video wirelessly to Apple TV and other AirPlay-enabled devices. The AirPlay controls appear when an AirPlay-enabled device is available on the same Wi-Fi network that iPhone is connected to. You can also mirror the contents of your iPhone screen on a TV.

Stream content to an AirPlay-enabled device: Tap ▭, then choose the device.

Access the AirPlay and volume controls while using any app: When the screen is on, double-click the Home button □ and scroll to the left end of the multitasking bar.



Switch playback back to iPhone: Tap ▭, then choose iPhone

Mirror the iPhone screen on a TV: Tap ▭ at the left end of the multitasking bar, choose an Apple TV, then tap Mirroring. A blue bar appears at the top of the iPhone screen when AirPlay mirroring is turned on. Everything on the iPhone screen appears on the TV.

Connecting iPhone to a TV using a cable

Apple cables and adapters (available separately) may be used to connect iPhone to a TV, projector, or other external display. For more information, go to support.apple.com/kb/HT4108.

Printing with AirPrint

AirPrint lets you print wirelessly to AirPrint-enabled printers from the following iOS apps:

- Mail—email messages and attachments that can be viewed in Quick Look
- Photos and Camera—photos
- Safari—webpages, PDFs, and other attachments that can be viewed in Quick Look
- iBooks—PDFs
- Maps—the portion of the map showing on the screen
- Notes—the currently displayed note

Other apps available from the App Store may also support AirPrint.

iPhone and the printer must be on the same Wi-Fi network. For more information about AirPrint, go to support.apple.com/kb/HT4356.

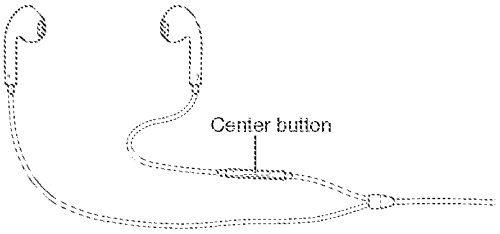
Print a document: Tap ↩ or ✎ (depending on the app you're using), then tap Print.

See the status of a print job: Double-click the Home button □, then tap Print Center in the multitasking bar. The badge on the icon shows how many documents are ready to print, including the current one.

Cancel a print job: In Print Center, select the print job, if necessary, then tap Cancel Printing.

Apple headset

The Apple EarPods with Remote and Mic (iPhone 5) and the Apple Earphones with Remote and Mic (iPhone 4S or earlier) feature a microphone, volume buttons, and an integrated button that allows you to answer and end calls, and control audio and video playback.



Plug in the headset to listen to music or make a phone call. Press the center button to control music playback and answer or end calls, even when iPhone is locked.

Adjust the volume: Press the + or − button.

Use the center button to control music playback:

- *Pause a song or video:* Press the center button. Press again to resume playback.
- *Skip to the next song:* Press the center button twice quickly.
- *Return to the previous song:* Press the center button three times quickly.
- *Fast-forward:* Press the center button twice quickly and hold.
- *Rewind:* Press the center button three times quickly and hold.

Use the center button to answer or make phone calls:

- *Answer an incoming call:* Press the center button.
- *End the current call:* Press the center button.
- *Decline an incoming call:* Press and hold the center button for about two seconds, then let go. Two low beeps confirm you declined the call.
- *Switch to an incoming or on-hold call, and put the current call on hold:* Press the center button. Press again to switch back to the first call.
- *Switch to an incoming or on-hold call, and end the current call:* Press and hold the center button for about two seconds, then let go. Two low beeps confirm you ended the first call.

Use Siri or Voice Control: Press and hold the center button.

See Chapter 4, Siri, on page 36 or Voice Control on page 26.

If you get a call while the headset is plugged in, you can hear the ringtone through both the iPhone speaker and the headset.

Bluetooth devices

You can use iPhone with the Apple Wireless Keyboard and other Bluetooth devices, such as Bluetooth headsets, car kits, and stereo headphones. For supported Bluetooth profiles, go to support.apple.com/kb/HT3647.

Pairing Bluetooth devices

WARNING: For important information about avoiding hearing loss and avoiding distraction while driving, see Important safety information on page 146.

Before you can use a Bluetooth device with iPhone, you must first pair them.

Pair a Bluetooth device with iPhone:

1 Make the device discoverable.

See the documentation that came with the device. For an Apple Wireless Keyboard, press the On/off switch.

2 Go to Settings > Bluetooth and turn Bluetooth on.

3 Select the device and, if prompted, enter the passkey or PIN. See the instructions about the passkey or PIN that came with the device.

For information about using an Apple Wireless Keyboard, see Apple Wireless Keyboard on page 24.

To use a Bluetooth headset with iPhone, see the documentation that came with the device.

Return audio output to iPhone when a Bluetooth headset is connected: Turn off or unpair the device, or turn off Bluetooth in Settings > Bluetooth. Audio output returns to iPhone whenever the device is out of range. You can also use AirPlay ▭ to switch audio output to iPhone. See AirPlay on page 30.

Bluetooth status

After you pair a device with iPhone, the Bluetooth icon appears in the status bar at the top of the screen:

- ᛒ *or* ᛒ*:* Bluetooth is on and paired with a device. (The color depends on the current color of the status bar.)
- ᛒ: Bluetooth is on and paired with a device, but the device is out of range or turned off.
- *No Bluetooth icon:* Bluetooth is not paired with a device.

Unpairing a Bluetooth device from iPhone

You can unpair a Bluetooth device if you don't want to use it with iPhone any more.

Unpair a Bluetooth device: Go to Settings > Bluetooth and turn on Bluetooth. Tap ⊙ next to the device name, then tap "Forget this Device."

File sharing

You can use iTunes to transfer files between iPhone and your computer. You can also view files received as email attachments on iPhone. See Reading mail on page 51. If you have the same apps that work with iCloud on more than one device, you can use iCloud to automatically keep your documents up to date across all your devices. See iCloud on page 15.

Transfer files using iTunes: Connect iPhone to your computer using the included cable. In iTunes on your computer, select iPhone, then click the Apps button. Use the File Sharing section to transfer documents between iPhone and your computer. Apps that support file sharing appear in the File Sharing Apps list in iTunes. To delete a file, select the file in the Files list, then press the Delete key.

Security features

Security features help protect the information on iPhone from being accessed by others.

Passcodes and data protection

For security, you can set a passcode that you must enter each time you turn on or wake up iPhone, or when you access the passcode lock settings.

Setting a passcode turns on data protection, which uses your passcode as the key for encrypting mail messages and attachments stored on iPhone. (Some apps available from the App Store may also use data protection.) A notice at the bottom of the Passcode Lock screen in Settings shows that data protection is enabled.

Important: On an iPhone 3GS that didn't ship with iOS 4 or later, you must also restore iOS software to enable data protection. See Updating and restoring iPhone software on page 152.

Set a passcode: Go to Settings > General > Passcode Lock, then tap Turn Passcode On and enter a 4-digit passcode.

Use a more secure passcode: To increase security, turn off Simple Passcode and use a longer passcode with a combination of numbers, letters, punctuation, and special characters.

To unlock iPhone when it's protected by a combination passcode, you enter the passcode using the keyboard. If you prefer to unlock iPhone using the numeric keypad, you can set up a longer passcode using numbers only.

Prevent access to Siri when iPhone is locked: Go to Settings > General > Passcode Lock, then turn Siri off.

Prevent voice dialing when iPhone is locked: Go to Settings > General > Passcode Lock, then turn Voice Dial off. (Available only when Siri is turned off in Settings > General > Siri.)

See Passcode Lock on page 136.

Find My iPhone

Find My iPhone can help you locate and secure your iPhone using the free Find My iPhone app on another iPhone, iPad, or iPod touch, or using a Mac or PC web browser signed in to www.icloud.com.

Find My iPhone includes:

- *Play Sound:* Play a sound for two minutes.
- *Lost Mode:* You can immediately lock your missing iPhone with a passcode and send it a message displaying a contact number. iPhone also tracks and reports its location, so you can see where it's been when you check the Find My iPhone app.
- *Erase iPhone:* Protects your privacy by erasing all the information and media on your iPhone and restoring iPhone to its original factory settings.

Important: To use these features, Find My iPhone must have been turned on in iCloud settings on your iPhone before it was lost, and iPhone must be connected to the Internet.

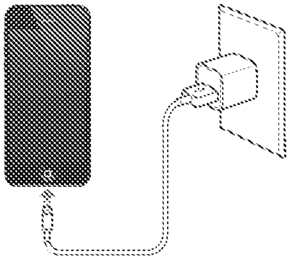
Turn on Find My iPhone: Go to Settings > iCloud, then turn on Find My iPhone.

Battery

iPhone has an internal, lithium-ion rechargeable battery. For more information about the battery—including tips for maximizing battery life—go to www.apple.com/batteries.

WARNING: For important safety information about the battery and charging iPhone, see Important safety information on page 146.

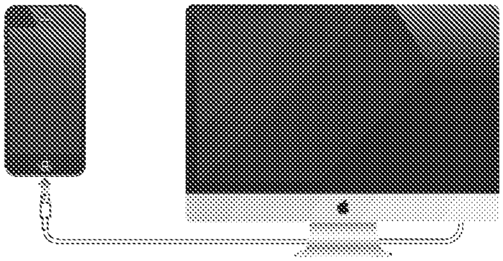
Charge the battery: Connect iPhone to a power outlet using the included cable and USB power adapter.



Note: Connecting iPhone to a power outlet can start an iCloud backup or wireless iTunes syncing. See Backing up iPhone on page 150 and Syncing with iTunes on page 16.

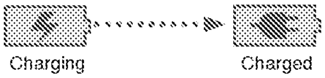
Charge the battery and sync iPhone using a computer: Connect iPhone to your computer using the included cable.

Unless your keyboard has a high-power USB 2.0 or 3.0 port, you must connect iPhone to a USB 2.0 or 3.0 port on your computer.



Important: The iPhone battery may drain instead of charge if iPhone is connected to a computer that's turned off or is in sleep or standby mode.

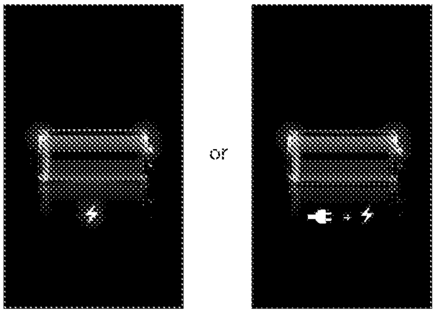
The battery icon in the upper-right corner shows the battery level or charging status.



Display the percentage of battery charge: Go to Settings > General > Usage and turn on the setting under Battery Usage.

If you charge the battery while syncing or using iPhone, it may take longer to charge.

Important: If iPhone is very low on power, it may display one of the following images, indicating that iPhone needs to charge for up to ten minutes before you can use it. If iPhone is extremely low on power, the display may be blank for up to two minutes before one of the low-battery images appears.



Rechargeable batteries have a limited number of charge cycles and may eventually need to be replaced.

Replace the battery: The iPhone battery isn't user replaceable; it can be replaced only by an authorized service provider. See www.apple.com/batteries/replacements.html.

Siri

4

What is Siri?

Siri is the intelligent personal assistant that helps you get things done just by talking. Siri understands natural speech, so you don't have to learn specific commands or remember keywords. You can ask things in different ways. For example, you can say "Set the alarm for 6:30 a.m." or "Wake me at 6:30 in the morning." Either way, Siri gets it.

WARNING: For important information about avoiding distraction while driving, see Important safety information on page 146.

Note: Siri is available on iPhone 4S or later, and requires Internet access. Cellular data charges may apply.

Siri lets you write and send a message, schedule a meeting, place a phone call, get directions, set a reminder, search the web, and much more—simply by talking naturally. Siri asks a question if it needs clarification or more information. Siri also uses information from your contacts, music library, calendars, reminders, and so forth to understand what you're talking about.

Siri works seamlessly with most of the built-in apps on iPhone, and uses Search and Location Services when needed. You can also ask Siri to open an app for you.

There's so much you can say to Siri—here are some more examples, for starters:

- Call Joe
- Set the timer for 30 minutes
- Directions to the nearest Apple store
- Is it going to rain tomorrow?
- Open Passbook
- Post to Facebook
- Tweet

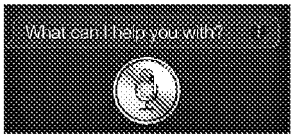
Using Siri

Starting Siri

Siri comes to life with the press of a button.

Start Siri: Press the Home button ○ until Siri appears. If you didn't turn Siri on when you set up iPhone, go to Settings > General > Siri.

You'll hear two quick beeps and see "What can I help you with?" on the screen.

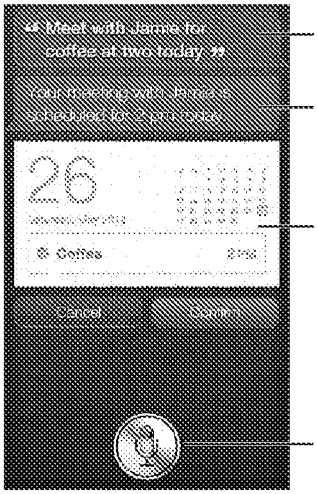


What can I help you with?

Just start speaking. The microphone icon lights up to let you know that Siri hears you talking. Once you've started a dialogue with Siri, tap the microphone icon to talk to it again.

Siri waits for you to stop speaking, but you can also tap the microphone icon to tell Siri you're done. This is useful when there's a lot of background noise. It can also speed up your conversation with Siri, since Siri won't have to wait for your pause.

When you stop speaking, Siri displays what it heard and provides a response. Siri often includes related info that might be useful. If the info is related to an app—for example, a text message you've composed, or a location you asked for—just tap the display to open the app for details and further action.



"Meet with Jamie for coffee at two today"

Your meeting with Jamie is scheduled for 2 pm today.

What Siri heard you say

Siri's response

Related info—tap to open the app.

Tap to speak to Siri.

Siri may ask you for clarification in order to complete a request. For example, tell Siri to "Remind me to call mom," and Siri may ask "What time would you like me to remind you?"

Cancel a request: Say "cancel," tap 🎤, or press the Home button ○.

Stop a phone call you started with Siri: Before the Phone app opens, press the Home button ○. If Phone is already open, tap End.

Telling Siri about yourself

The more Siri knows about you, the more it can use your information to help you. Siri gets your information from your personal info card ("My Info") in Contacts.

Tell Siri who you are: Go to Settings > General > Siri > My Info, then tap your name.

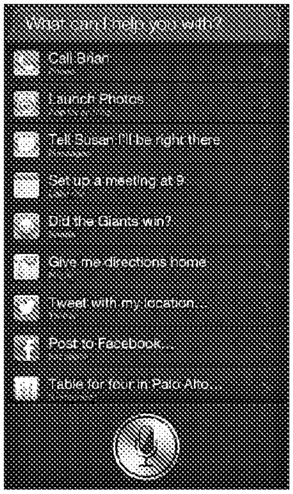
Put your home and work addresses on your card, so you can say things like "How do I get home?" and "Remind me to call Bob when I get to work."

Siri also wants to know about the important people in your life, so put those relationships on your personal info card—Siri can help you. For example, the first time you tell Siri to call your sister, Siri asks you who your sister is (if you don't already have that info on your card). Siri adds that relationship to your personal info card so it doesn't have to ask next time.

Create cards in Contacts for all your important relationships, and include information such as phone numbers, email addresses, home and work addresses, and nicknames you like to use.

Onscreen guide

Siri prompts you with examples of things you can say, right on screen. Ask Siri "what can you do" or tap ⓘ when Siri first appears. Siri displays a list of the apps it supports, with an example request. Tap an item in the list to see more examples.



What can I help you with?

Call Brian

Launch Photos

Tell Susan I'll be right there

Set up a meeting at 9

Did the Giants win?

Give me directions home

Tweet with my location...

Post to Facebook...

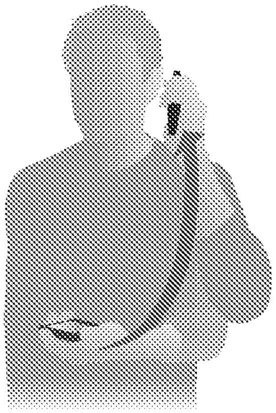
Table for four in Palo Alto...

Raise to Speak

You can start talking to Siri just by bringing iPhone to your ear, like making a phone call. If the screen isn't on, first press the Sleep/Wake or Home button. You'll hear two quick beeps to indicate Siri is listening. Then start talking.

Turn on Raise to Speak: Go to Settings > General > Siri.

If Siri doesn't respond when you bring iPhone to your ear, start with the screen facing you, so your hand rotates on the way up.



Handsfree Siri

You can use Siri with the headset that came with iPhone, and with other compatible wired or Bluetooth headsets.

Talk to Siri using a headset: Press and hold the center button (or the call button on a Bluetooth headset).

To continue a conversation with Siri, press and hold the button each time you want to talk.

When you use a headset, Siri speaks its responses to you. Siri reads back text messages and email messages that you've dictated before sending them. This gives you a chance to change the message if you want. Siri also reads back the subjects of reminders before creating them.

Location Services

Because Siri knows locations (iPhone 4S or later) like "current," "home," and "work," it can remind you to do a certain task when you leave a location or arrive at a location. Tell Siri "Remind me to call my daughter when I leave the office," and Siri does just that.

Location information isn't tracked or stored outside iPhone. You can still use Siri if you turn Location Services off, but Siri won't do anything that requires location information.

Turn off Location Services for Siri: Go to Settings > Privacy > Location Services.

Accessibility

Siri is accessible to blind and visually impaired users through VoiceOver, the screen reader built into iOS. VoiceOver describes aloud what's onscreen—including any text in Siri's responses—so you can use iPhone without seeing it.

Turn on VoiceOver: Go to Settings > General > Accessibility.

Turning on VoiceOver causes even your notifications to be read aloud for you. For more information, see VoiceOver on page 115.

Setting options for Siri

Turn Siri on or off: Go to Settings > General > Siri.

Note: Turning Siri off resets Siri, and Siri forgets what it's learned about your voice.

Set options for Siri: Go to Settings > General > Siri.

- *Language:* Select the language you want to use with Siri.
- *Voice Feedback:* By default, Siri speaks its responses only when you hold iPhone to your ear or use Siri with a headset. If you want Siri to always speak its responses, set this option to Always.
- *My Info:* Let Siri know which card in Contacts contains your personal info. See Telling Siri about yourself on page 37.
- *Raise to Speak:* Talk to Siri by bringing iPhone to your ear when the screen is on. To turn this feature on or off, go to Settings > General > Siri.

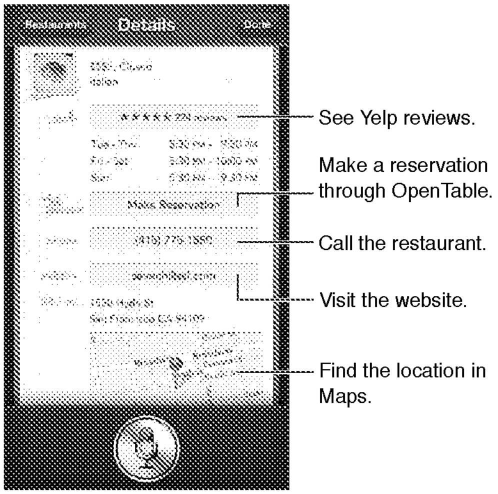
Allow or prevent access to Siri when iPhone is locked with a passcode: Go to Settings > General > Passcode Lock.

You can also disable Siri by turning on restrictions. See Restrictions on page 137.

Restaurants

Siri works with Yelp, OpenTable, and others to provide information about restaurants and help you make reservations. Ask to find restaurants by cuisine, price, location, outdoor seating, or a combination of options. Siri can show you available photos, Yelp stars, price range, and reviews. Get more information by using the Yelp and OpenTable apps—iPhone prompts you to download them if you don't already have them installed.

See detailed info about a restaurant: Tap a restaurant that Siri suggests.

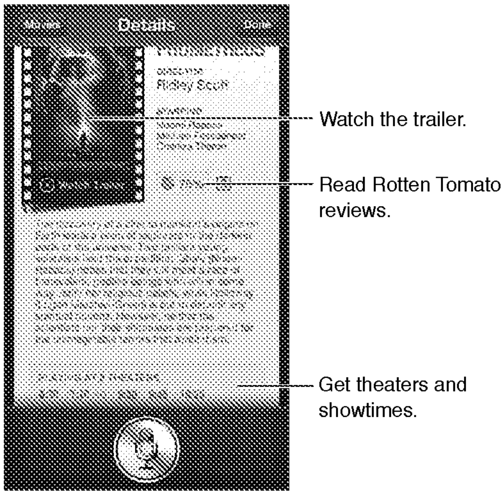


Movies

Ask Siri about what movies are playing, or where you can see a specific movie. Find out when a film premiered, who directed it and what awards it won. Siri gives theater locations, show times, and Rotten Tomato reviews.

Siri works with Fandango to help you purchase tickets (for theaters that support it). Ask about showtimes for a movie, or tell Siri you want to buy tickets. When you tap Buy Tickets, Fandango opens if it's installed, or you'll be prompted to install the app from the App Store.

See detailed info about a movie: Tap a movie that Siri suggests.



Sports

Siri knows a lot about sports—including baseball, basketball, football, soccer, and hockey. Ask Siri for game schedules, scores from the current season's games, or up-to-the minute scores from live games. Tell Siri to show you player stats and compare them against other players' stats. Siri tracks team records, too. Here are some things you might ask:

- What was the score of the last Giants game?
- What are the National League standings?
- When is the Chicago Cubs first game of the season?

Dictation

When Siri is turned on, you can also dictate text. See Dictation on page 25.

Although you can compose email, text messages, and other text by talking directly with Siri, you might prefer dictation. Dictation lets you edit a message instead of replacing the entire text. Dictation also gives you more time to think while composing.

Siri understands a pause to mean you finished talking for the moment, and takes that opportunity to respond. While this lets you have a natural conversation with Siri, Siri might interrupt you before you're really done if you pause too long. With dictation, you can pause as much as you like, and resume talking when you're ready.

You can also start composing text using Siri, then continue using dictation. For example, you can create an email with Siri, then tap the draft to open the message in Mail. In Mail, you can complete or edit the message and make other changes, such as adding or removing recipients, revising the subject, or changing the account you're sending the email from.

Correcting Siri

If Siri is having trouble

Siri may sometimes have trouble understanding you—in a noisy environment, for example. If you speak with an accent, it can take Siri some time to get used to your voice. If Siri doesn't hear you exactly right, you can make corrections.

Siri shows what it heard you say, along with its response.

Correct what Siri hears you say: Tap the bubble showing what Siri heard you say. Edit your request by typing, or tap 🎤 on the keyboard to dictate.

For information about using dictation, see Dictation on page 41.

If some of the text is underlined in blue, tap it and Siri suggests some alternatives. Tap one of the suggestions, or replace the text by typing or dictating.

Correct Siri by voice: Tap 🎤, then restate or clarify your request. For example, "I meant Boston."

When correcting Siri, don't say what you *don't* want—just tell Siri what you *do* want.

Correct a mail or text message: If Siri asks if you want to send the message, say something like:

- Change it to: Call me tomorrow.
- Add: See you there question mark.
- No, send it to Bob.
- No. (to keep the message without sending it)
- Cancel.

To have Siri read the message to you, say "Read it back to me" or "Read me the message." If it's correct, say something like "Yes, send it."

Noisy environments

In a noisy environment, hold iPhone close to your mouth, but don't talk directly into the bottom edge. Continue to speak clearly and naturally. Tap 🎙 when you finish speaking.

You can also try holding iPhone to your ear to speak to Siri.

Network connection

Siri might tell you it's having trouble connecting to the network. Because Siri relies on Apple servers for voice recognition and other services, you need to have a good 3G, 4G, or LTE cellular connection or a Wi-Fi connection to the Internet.

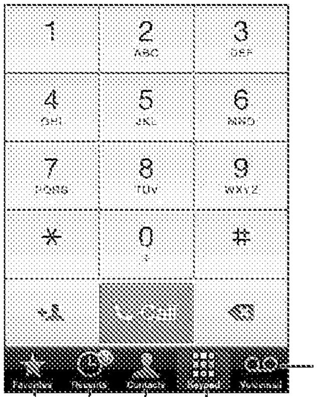
Phone

5

Phone calls

Making calls

Making a call on iPhone is as simple as tapping a name or number in your contacts, using Siri to say "call Bob" (iPhone 4S or later), tapping one of your favorites, or tapping a recent call to return it.



View a list of your voicemail messages.

Dial manually.

Call, email, or text someone in your contacts list.

View your recent incoming and outgoing calls to return a call or get more info. The red badge indicates the number of missed calls.

Call a favorite with a single tap.

WARNING: For important information about avoiding distraction, see Important safety information on page 146.

Buttons at the bottom of the Phone screen give you quick access to your favorites, recent calls, your contacts, and a numeric keypad for dialing manually.

Manually dial a number: Tap Keypad, enter the number, then tap Call.

- *Paste a number to the keypad:* Tap the screen above the keyboard, then tap Paste.
- *Enter a soft (2-second) pause:* Touch the "*" key until a comma appears.
- *Enter a hard pause (to pause dialing until you tap the Dial button):* Touch the "#" key until a semicolon appears.
- *Redial the last number:* Tap Keypad, tap Call to display the number, then tap Call again.

Add a contact to Favorites: In Contacts, tap "Add to Favorites" at the bottom of a contact card. To delete or rearrange your favorites list, tap Edit.

Use Siri or Voice Control: Press and hold the Home button ◻, say *call* or *dial*, then say the name or number. You can add *at home, work,* or *mobile.* See Chapter 4, Siri, on page 36 and Voice Control on page 26.

For best results, speak the full name of the person you're calling. When voice dialing a number, speak each digit separately—for example, *four one five, five five five, one two one two.* For the 800 area code in the U.S., you can say *eight hundred.*

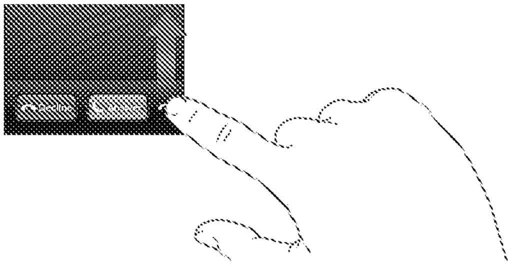
Receiving calls

Answer a call: Tap Answer. If iPhone is locked, drag the slider. You can also press the center button on your headset.

Silence a call: Press the Sleep/Wake button or either volume button. You can still answer the call after silencing it, until it goes to voicemail.

Reply to an incoming call with a text message: Swipe ☎ up, tap "Reply with Message," then choose a reply or tap Custom. To create your own default replies, go to Settings > Phone > "Reply with Message" and replace any of the default messages.

Remind yourself to return an incoming call: Swipe ☎ up, tap Remind Me Later, then choose when you want to be reminded.



Decline a call and send it directly to voicemail: Do one of the following:

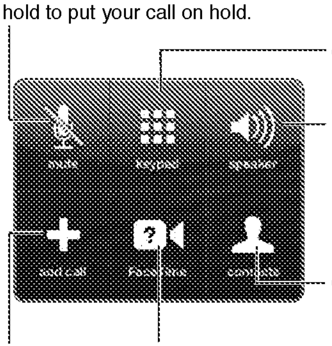
- Press the Sleep/Wake button twice quickly.
- Press and hold the center button on your headset for about two seconds. Two low beeps confirm that the call was declined.
- Tap Decline (if iPhone is awake when the call comes in).

Block calls and maintain Wi-Fi access to the Internet: Go to Settings and turn on Airplane Mode, then tap Wi-Fi to turn it on.

Set iPhone to Do Not Disturb (☾): Go to Settings and turn on Do No Disturb. See Do Not Disturb and Notifications on page 132.

While on a call

When you're on a call, the screen shows call options.



Mute your line. *iPhone 4 or later:* Touch and hold to put your call on hold.

Dial a number or enter numbers.

Use the speakerphone or a Bluetooth device.

Get contact info.

Make another call.

Make a FaceTime call.

Use another app during a call: Press the Home button ◻, then open the app. To return to the call, tap the green bar at the top of the screen.

End a call: Tap End. Or press the center button on your headset.

Respond to a second incoming call:

- *Ignore the call and send it to voicemail:* Tap Ignore.
- *Put the first call on hold and answer the new one:* Tap Hold Call + Answer.
- *End the first call and answer the new one:* When using a GSM network, tap End Call + Answer. With a CDMA network, tap End Call and when the second call rings back, tap Answer, or drag the slider if the phone is locked.

If you're on a FaceTime video call, you can either end the video call and answer the incoming call, or decline the incoming call.

Switch between calls: Tap Swap. The active call is put on hold. With CDMA, you can't switch between calls if the second call was outgoing, but you can merge the calls. If you end the second call or the merged call, both calls are terminated.

Merge calls: Tap Merge Calls. With CDMA, you can't merge calls if the second call was incoming.

Conference calls

With GSM, you can set up a conference call with up to five people at a time, depending on your carrier.

Create a conference call: While on a call, tap Add Call, make another call, then tap Merge Calls. Repeat to add more people to the conference.

- *Drop one person:* Tap Conference, tap ⊖ next to a person, then tap End Call.
- *Talk privately with one person:* Tap Conference, then tap Private next to the person. Tap Merge Calls to resume the conference.
- *Add an incoming caller:* Tap Hold Call + Answer, then tap Merge Calls.

Note: You can't make a FaceTime video call when you're on a conference call.

Using a Bluetooth device

For information about using a Bluetooth device, see the documentation that came with the device. See Pairing Bluetooth devices on page 32.

Bypass your Bluetooth device:

- Answer a call by tapping the iPhone screen.
- During a call, tap Audio and choose iPhone or Speaker Phone.
- Turn off Bluetooth in Settings > Bluetooth.
- Turn off the Bluetooth device, or move out of range. You must be within about 30 feet (10 meters) of a Bluetooth device for it to be connected to iPhone.

Emergency calls

Make an emergency call when iPhone is locked: On the Enter Passcode screen, tap Emergency Call.

Important: iPhone can be used to make an emergency call in many locations, provided that cellular service is available, but it should not be relied on for emergencies. Some cellular networks may not accept an emergency call from iPhone if iPhone is not activated, if iPhone is not compatible with or configured to operate on a particular cellular network, or (when applicable) if iPhone does not have a SIM card or if the SIM card is PIN-locked.

In the U.S., location information (if available) is provided to emergency service providers when you dial 911.

With CDMA, when an emergency call ends, iPhone enters *emergency call mode* for a few minutes to allow a call back from emergency services. During this time, data transmission and text messages are blocked.

Exit emergency call mode (CDMA): Do one of the following:

- Tap the back button.
- Press the Sleep/Wake button or the Home button □.
- Use the keypad to dial a non-emergency number.

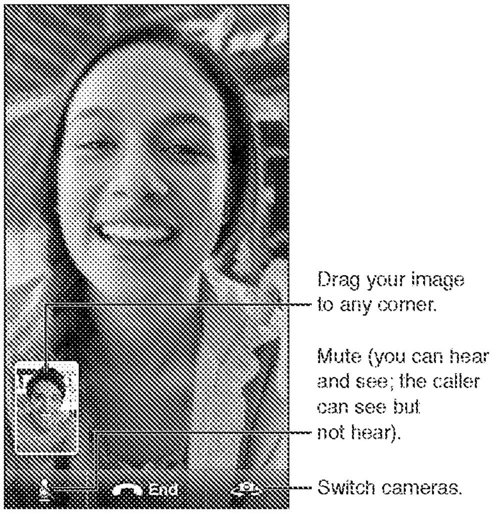
FaceTime

With iPhone 4 or later, you can make a video call to someone with a Mac or other iOS device that supports FaceTime. The FaceTime camera lets you talk face-to-face; switch to the iSight camera on the back to share what you see around you.

Note: On iPhone 3GS or iPhone 4, you need a Wi-Fi connection to the Internet. On iPhone 4S or later, you can also make FaceTime calls over a cellular data connection. Cellular data charges may apply. To turn off FaceTime using cellular data, go to Settings > General > Cellular.

Make a FaceTime call: In Contacts, choose a name, tap FaceTime, then tap the phone number or email address that the person uses for FaceTime.

To call someone who has an iPhone 4 or later, you can start by making a voice call, then tap FaceTime.


End
Drag your image to any corner.
Mute (you can hear and see; the caller can see but not hear).
Switch cameras.

Note: With FaceTime, your phone number is displayed even if caller ID is blocked or turned off.

Use Siri or Voice Control: Press and hold the Home button ○, then say "FaceTime," followed by the name of the person to call.

Set FaceTime options: Go to Settings > FaceTime to:

- Turn FaceTime on or off
- Specify your Apple ID or an email address for receiving FaceTime calls

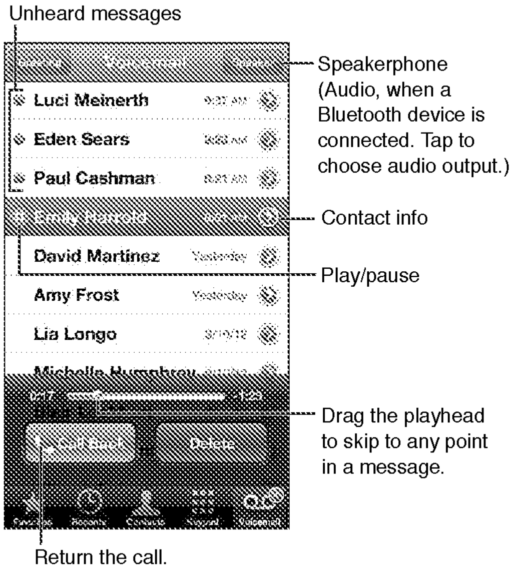
Visual voicemail

Visual voicemail lets you see a list of your messages and choose which ones to listen to or delete, without having to listen to instructions or prior messages. The badge on the Voicemail icon tells you how many unheard messages you have.

Set up visual voicemail: The first time you tap Voicemail, you're prompted to create a voicemail password and record your voicemail greeting.

Listen to a voicemail message: Tap Voicemail, then tap a message. To listen again, select the message and tap ▶. If visual voicemail isn't available with your service, tap Voicemail and follow the voice prompts.

Check voicemail from another phone: Dial your own number or your carrier's remote access number.



Messages are saved until you delete them or your carrier erases them.

Delete a message: Swipe or tap the message, then tap Delete.

Note: In some areas, deleted messages may be permanently erased by your carrier.

Manage deleted messages: Tap Deleted Messages (at the end of the messages list), then:

- *Listen to a deleted message:* Tap the message.
- *Undelete a message:* Tap the message and tap Undelete.
- *Delete messages permanently:* Tap Clear All.

Change your greeting: Tap Voicemail, tap Greeting, tap Custom, then tap Record and say your greeting. Or, to use your carrier's generic greeting, tap Default.

Set an alert sound for new voicemail: Go to Settings > Sounds, then tap New Voicemail.

Note: If the Ring/Silent switch is off, iPhone won't sound alerts.

Change the voicemail password: Go to Settings > Phone > Change Voicemail Password.

Contacts

From a contact's info screen, a quick tap lets you make a phone call, create an email message, find the contact's location, and more. See Chapter 25, Contacts, on page 100.

Call forwarding, call waiting, and caller ID

The following information applies only to GSM networks. For CDMA networks, contact your carrier for information about enabling and using these features. See support.apple.com/kb/HT4515.

Turn call forwarding on or off: Go to Settings > Phone > Call Forwarding. The Call Forwarding icon (↪) appears in the status bar when call forwarding is on. You must be in range of the cellular network when you set iPhone to forward calls, or calls won't be forwarded. FaceTime calls are not forwarded.

Turn call waiting on or off: Go to Settings > Phone > Call Waiting. If you're on a call and call waiting is turned off, incoming calls go directly to voicemail.

Turn caller ID on or off: Go to Settings > Phone > Show My Caller ID.

Note: For FaceTime calls, your phone number is displayed even if caller ID is turned off.

Ringtones, Ring/Silent switch, and vibrate

iPhone comes with ringtones that sound for incoming calls, Clock alarms, and the Clock timer. You can also purchase ringtones from songs in iTunes. See Chapter 22, iTunes Store, on page 94.

Set the default ringtone: Go to Settings > Sounds > Ringtone.

Turn the ringer on or off: Flip the switch on the side of iPhone.

Important: Clock alarms still sound even if you set the Ring/Silent switch to silent.

Turn vibrate on or off: Go to Settings > Sounds.

Assign a different ringtone for a contact: In Contacts, choose a contact, tap edit, then tap Ringtone and choose a ringtone.

For more information, see Sounds on page 139.

International calls

For information about making international calls from your home area, including rates and other charges that may apply, contact your carrier or go to your carrier's website.

When traveling abroad, you may be able to use iPhone to make calls, send and receive text messages, and use apps that access the Internet, depending on available networks.

Enable international roaming: Contact your carrier for information about availability and fees.

Important: Voice, text message, and data roaming charges may apply. To avoid charges when roaming, turn off Voice Roaming and Data Roaming.

If you have an iPhone 4S or later that's been activated to work on a CDMA network, you may be able to roam on GSM networks if the phone has a SIM card installed. When roaming on a GSM network, iPhone has access to GSM network features. Charges may apply. Contact your carrier for more information.

Set network options: Go to Settings > General > Cellular to:

- Turn data roaming on or off.
- Turn cellular data on or off.
- Turn voice roaming on or off (CDMA).
- Use GSM networks abroad (CDMA).

Turn off cellular services: Go to Settings, turn on Airplane Mode, then tap Wi-Fi and turn Wi-Fi on. Incoming phone calls are sent to voicemail. To resume cellular service, turn Airplane Mode off.

Automatically add the prefix or country code for calls to the U.S.: (GSM) Go to Settings > Phone, then turn on Dial Assist. This lets you use contacts and favorites to make calls while abroad.

Choose a carrier: Go to Settings > Carrier. This option is available only when you're traveling outside your service provider's network, and for carriers that have roaming agreements with your provider. See Carrier on page 133.

Get voicemail when visual voicemail isn't available: Dial your own number (with CDMA, add # after your number), or touch and hold "1" on the numeric keypad.

Setting options for Phone

Go to Settings > Phone to:

- See the phone number for your iPhone
- Change the default text message replies for incoming calls
- Turn call forwarding, call waiting, and caller ID on or off (GSM)
- Turn TTY on or off
- Change your voicemail password (GSM)
- Require a PIN to unlock your SIM when you turn iPhone on (required by some carriers)

Go to Settings > FaceTime to:

- Turn FaceTime on or off
- Use your Apple ID for FaceTime
- Add an email address for FaceTime
- Turn cellular data on or off

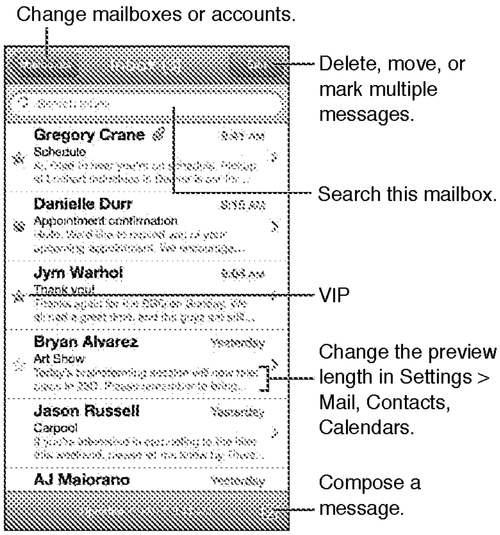
Go to Settings > Sounds to:

- Set ringtones and volume
- Set vibration options
- Set the sound for new voicemail

Mail

6

Reading mail


Change mailboxes or accounts.
Delete, move, or mark multiple messages.
Search this mailbox.
VIP
Change the preview length in Settings > Mail, Contacts, Calendars.
Compose a message.

Gregory Crane
Schedule

Danielle Durr
Appointment confirmation

Jym Warhol
Thank you!

Bryan Alvarez — Yesterday
Art Show

Jason Russell — Yesterday
Carpool

AJ Maiorano — Yesterday

Flag a message or mark it as unread: Tap 🏴. To mark multiple messages at once, tap Edit while viewing the message list.

Identify messages addressed specifically to you: Go to Settings > Mail, Contacts, Calendars, then turn Show To/Cc Label on or off. Messages with your address in the To or Cc field are indicated with an icon in the message list.

See all the recipients of a message: Tap the word Details in the From field. Tap a recipient's name or email address to view the recipient's contact information or add them to Contacts or your VIP list.

Prevent downloading remote images: Go to Settings > Mail, Contacts, Calendars, then turn Load Remote Images off.

Open a link: Tap the link to use its default action, or touch and hold to see other actions. For example, for an address, you can show its location in Maps or add it to Contacts. For a web link, you can add it to Reading List.

Open a meeting invitation or attachment: Tap the item. If the attachment can be used by multiple apps, touch and hold to choose an app that works with the file.

Save an attached photo or video: Touch and hold the photo or video, then tap Save Image or Video. It's saved to your Camera Roll in the Photos app.

Load new messages: Pull the message list or mailbox list downward to refresh the list.

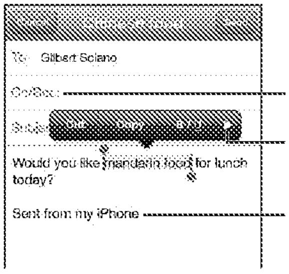
- *Set the number of older messages retrieved:* Go to Settings > Mail, Contacts, Calendars > Show.

Turn off new message notifications for an account: Go to Settings > Notifications > Mail > *account name*, then turn Notification Center off.

Change the tones played by Mail: Go to Settings > Sound.

- *Change the tone played for new mail in each account:* Go to Settings > Notifications > Mail > *account name* > New Mail Sound.
- *Change the tone played for new mail from VIPs:* Go to Settings > Notifications > Mail > VIP > New Mail Sound.

Sending mail



To: Gilbert Solano

Cc/Bcc:

Subject:

Would you like mandarin food for lunch today?

Sent from my iPhone

Tap to change From, Cc, or Bcc.

Tap to attach a photo or video.

Change your signature in Settings > Mail, Contacts, Calendars.

Compose a message: Tap ✎, then type a name or email address. After you enter recipients, you can drag to move them between fields, such as from To to Cc. If you have multiple mail accounts, tap From to change the account you're sending from.

Automatically Bcc yourself on outgoing messages: Go to Settings > Mail, Contacts, Calendars > Always Bcc Myself.

Save a draft of a message: Tap Cancel, then tap Save. The message is saved in the account's Drafts mailbox. Touch and hold ✎ to see your saved drafts.

Reply to a message: Tap ↩, then tap Reply. Files or images attached to the initial message aren't sent back. To include the attachments, forward the message instead of replying.

Forward a message: Open a message and tap ↩, then tap Forward. This also forwards the message's attachments.

Quote a portion of the message you're replying to or forwarding: Touch and hold to select text. Drag the grab points to select the text you want to include in your reply, then tap ↩.

- *Change the quote level:* Select the text to indent, tap ▶ at least twice, then tap Quote Level.
- *Automatically increase the quote level when replying:* Go to Settings > Mail, Contacts, Calendars, then turn on Increase Quote Level.

Send a photo or video in a message: Tap the insertion point to display the selection buttons. Tap ▶, tap Insert Photo or Video, then choose a photo or video from an album. You can also email multiple photos using Photos—see Sharing photos and videos on page 72.

Change your email signature: Go to Settings > Mail, Contacts, Calendars > Signature. If you have more than one mail account, tap Per Account to specify a different signature for each account.

Organizing mail

See messages from VIPs: Go to the mailbox list (tap Mailboxes to get there), then tap VIP.

- *Add a person to the VIP list:* Tap the person's name or address in a From, To, or Cc/Bcc field, then tap Add to VIP.

Group related messages together: Go to Settings > Mail, Contacts, Calendars, then turn Organize by Thread on or off.

Search messages: Open a mailbox, scroll to the top, then enter text in the Search field. You can search the From, To, or the Subject field in the mailbox that's currently open. For mail accounts that support searching messages on the server, tap All to search From, To, Subject, and the message body.

Delete a message: If the message is open, Tap 🗑.

- *Delete a message without opening it:* Swipe over the message title, then tap Delete.
- *Delete multiple messages:* While viewing the message list, tap Edit.
- *Turn off deletion confirmation:* Go to Settings > Mail, Contacts, Calendars > Ask Before Deleting.

Recover a message: Go to the account's Trash mailbox, open the message, tap 📁, then move the message to the account's inbox or other folder.

- *Set how long your messages stay in Trash before being permanently deleted:* Go to Settings > Mail, Contacts, Calendars > *account name* > Account > Advanced.

Turn archiving on or off: Go to Settings > Mail, Contacts, Calendars > *account name* > Account > Advanced. Not all mail accounts support archiving. When you archive a message, it moves to the All Mail mailbox. To delete the message instead of archiving it, touch and hold Archive, then tap Delete.

Move a message to a different mailbox: While viewing the message, tap 📁, then choose a destination. To move multiple messages at once, tap Edit while viewing the message list.

Add, rename, or delete a mailbox: In the mailbox list, tap Edit. Some mailboxes can't be renamed or deleted.

Printing messages and attachments

Print a message: Tap ↩, then tap Print.

Print an inline image: Touch and hold the image, then tap Save Image. Go to Photos and print the image from your Camera Roll album.

Print an attachment: Tap the attachment to open it in Quick Look, tap ⇪, then tap Print.

For more information about printing, see Printing with AirPrint on page 30.

Mail accounts and settings

Change Mail and mail account settings: Go to Settings > Mail, Contacts, Calendars. You can set up:

- iCloud
- Microsoft Exchange and Outlook
- Google
- Yahoo!
- AOL
- Microsoft Hotmail
- Other POP and IMAP accounts

Settings vary based on the type of account you're setting up. Your Internet service provider or system administrator can provide the information you need to enter.

Temporarily stop using an account: Go to Settings > Mail, Contacts, Calendars, choose an account, then turn off mail service for the account. When the service is turned off, iPhone doesn't display or sync that information until you turn it back on. This is a good way to stop receiving work email while on vacation, for example.

Delete an account: Go to Settings > Mail, Contacts, Calendars, choose an account, then scroll down and tap Delete Account. All information synced with that account, such as bookmarks, mail, and notes, is removed.

Set Push settings: Go to Settings > Mail, Contacts, Calendars > Fetch New Data. Push delivers new information whenever it appears on the server and there's an Internet connection (some delays may occur). When Push is turned off, use the Fetch New Data setting to determine how often data is requested. The setting you choose here overrides individual account settings. For optimal battery life, don't fetch too often. Not all accounts support push.

Send signed and encrypted messages: Go to Settings > Mail, Contacts, Calendars > *account name* > Account > Advanced. Turn on S/MIME, then select certificates for signing and encrypting outgoing messages. To install certificates, you may get a configuration profile from your system administrator, download the certificates from the issuer's website using Safari, or receive them as mail attachments.

Set advanced options: Go to Settings > Mail, Contacts, Calendars > *account name* > Account > Advanced. Options vary depending on the account, and may include:

- Store drafts, sent messages, and deleted messages on iPhone
- Set how long deleted messages are kept before being permanently removed
- Adjust mail server settings
- Adjust SSL and password settings

Ask your Internet service provider or system administrator if you're not sure what the appropriate settings are for your account.

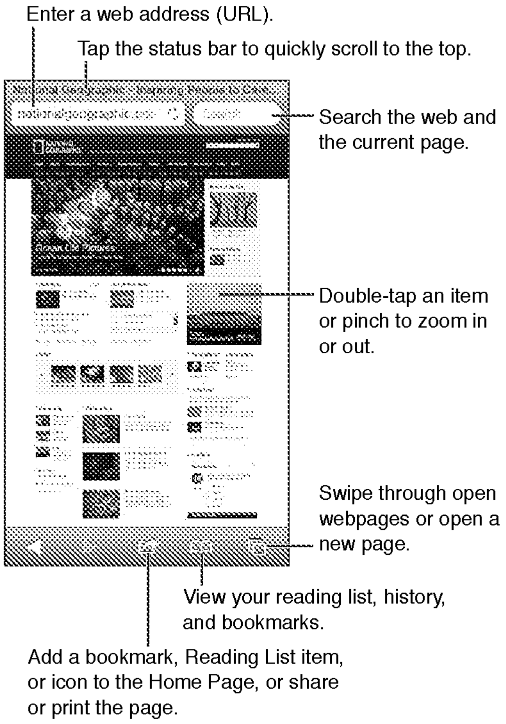
Safari

7

Safari features include:

- Reader—view articles without ads or clutter
- Reading list—collect articles to read later
- Full-screen mode—when viewing webpages in landscape orientation

Use iCloud to see pages you have open on other devices, and to keep your bookmarks and reading list up to date on your other devices.


Enter a web address (URL).
Tap the status bar to quickly scroll to the top.
Search the web and the current page.
Double-tap an item or pinch to zoom in or out.
Swipe through open webpages or open a new page.
View your reading list, history, and bookmarks.
Add a bookmark, Reading List item, or icon to the Home Page, or share or print the page.

View a webpage: Tap the address field (in the title bar), enter the URL, then tap Go.

- *Scroll a webpage:* Drag up, down, or sideways.
- *Scroll within a frame:* Drag two fingers inside the frame.
- *View in full-screen landscape:* Rotate iPhone, then tap ⤡.
- *Reload a webpage:* Tap ↻ in the address field.

Close a webpage: Tap ⧉, then tap ⊗ by the page.

See webpages you have open on your other devices: Tap 📖, then tap iCloud Tabs. To share webpages you have open on iPhone with your other devices using iCloud Tabs, go to Settings > iCloud and turn on Safari.

Follow a link on a webpage: Tap the link.

- *See a link's destination:* Touch and hold the link.
- *Open a link in a new tab:* Touch and hold the link, then tap "Open in New Page."

Detected data—such as phone numbers and email addresses—may also appear as links in webpages. Touch and hold a link to see the available options.

View an article in Reader: Tap the Reader button, if it appears in the address field.

- *Adjust the font size:* Tap AA.
- *Share the article:* Tap 📤.

  Note: When you email an article from Reader, the full text of the article is sent, in addition to the link.
- *Return to normal view:* Tap Done.

Use Reading List to collect webpages and read them later:

- *Add the current webpage:* Tap 📤, then tap "Add to Reading List." With iPhone 4 or later, the webpage is saved as well as the link, so you can read it even when you can't connect to the Internet.
- *Add the destination of a link:* Touch and hold the link, then tap "Add to Reading List."
- *View your reading list:* Tap 📖, then tap Reading List.
- *Delete an item from your reading list:* Swipe the item, then tap Delete.

Fill out a form: Tap a text field to bring up the keyboard.

- *Move to a different text field:* Tap the text field, or tap Next or Previous.
- *Submit a form:* Tap Go, Search, or the link on the webpage to submit the form.
- *Enable AutoFill:* Go to Settings > Safari > AutoFill.

Search the web, the current webpage, or a searchable PDF: Enter text in the search field.

- *Search the web:* Tap one of the suggestions that appear, or tap Search.
- *Find the search text on the current webpage or PDF:* Scroll to the bottom of the screen, then tap the entry below On This Page.

  The first instance is highlighted. To find later instances, tap ▶.

Bookmark the current webpage: Tap 📤, then tap Bookmark.

When you save a bookmark, you can edit its title. By default, bookmarks are saved at the top level of Bookmarks. To choose a different folder, tap Bookmarks on the Add Bookmarks screen.

Create an icon on the Home screen: Tap 📤, then tap "Add to Home Screen." Safari adds an icon for the current webpage to your Home Screen. Unless the webpage has a custom icon, that image is also used for the web clip icon on the Home screen. Web clips are backed up by iCloud and iTunes, but they aren't pushed to other devices by iCloud or synced by iTunes.

Share or copy a link for the current webpage: Tap 📤, then tap Mail, Message, Twitter, Facebook, or Copy.

Print the current webpage: Tap 📤, then tap Print. See Printing with AirPrint on page 30.

Use iCloud to keep your bookmarks and reading list up to date on your other devices: Go to Settings > iCloud and turn on Safari. See iCloud on page 15.

Set options for Safari: Go to Settings > Safari. Options include:

- Search engine
- AutoFill for filling out forms
- Opening links in a new page or in the background
- Private browsing to help protect private information and block some websites from tracking your behavior
- Clearing history, cookies, and data
- Cellular data for Reading List
- Fraud warning

Music

8

Getting music

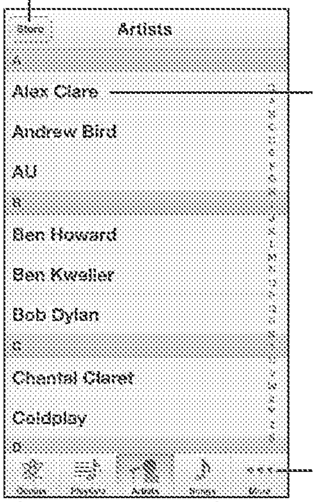
Get music and other audio content onto iPhone:

- *Purchase and download from the iTunes Store:* In Music, tap Store. See Chapter 22, iTunes Store, on page 94.
- *Automatically download music purchased on your other iOS devices and computers:* See iCloud on page 15.
- *Sync content with iTunes on your computer:* See Syncing with iTunes on page 16.
- *Use iTunes Match to store your music library in iCloud:* See iTunes Match on page 62.

Playing music

WARNING: For important information about avoiding hearing loss, see Important safety information on page 146.

You can listen to audio from the built-in speaker, headphones attached to the headset jack, or wireless Bluetooth stereo headphones paired with iPhone. When headphones are attached or paired, no sound comes from the speaker.



Open iTunes Store.

Tap to listen.

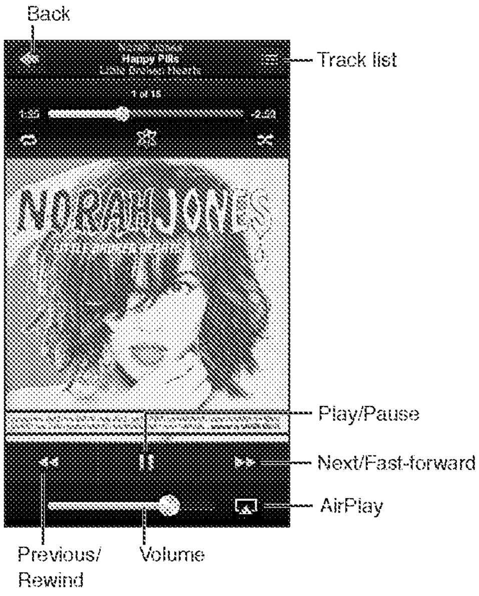
See additional browse buttons.

Choose how to browse.

Play a track: Browse by playlist, artist, song, or other category, then tap the track.

- *See additional browse buttons:* Tap More.
- *Change which browse buttons appear at the bottom:* Tap More, tap Edit, then drag an icon over the button you want to replace.

The Now Playing screen shows you what's playing, and provides playback controls.



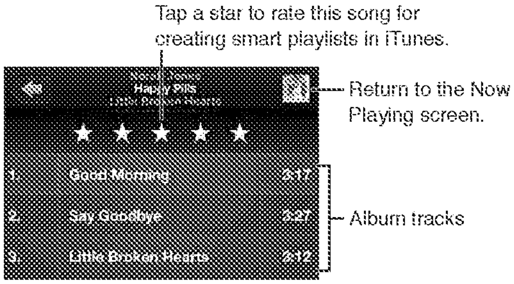
Lyrics appear on the Now Playing screen if you've added them to the song using the song's Info window in iTunes and you've synced music using iTunes. (Lyrics aren't supported by iTunes Match.)

Display additional controls (iPhone 4S or earlier): Tap the album artwork on the Now Playing screen to display the scrubber bar and playhead, and the Repeat, Genius, and Shuffle buttons.

Skip to any point in a song: Drag the playhead along the scrubber bar. Slide your finger down to slow down the scrub rate.

Shake to shuffle: Shake iPhone to turn on shuffle, and to change songs. To turn Shake to Shuffle on or off, go to Settings > Music.

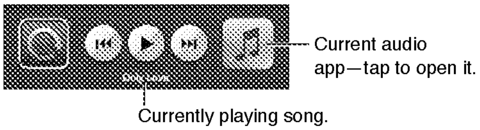
See all tracks on the album containing the current song: Tap ▤. To play a track, tap it.



1. Good Morning 2:17
2. Say Goodbye 3:27
3. Little Broken Hearts 2:12

Search music (titles, artists, albums, and composers): While browsing, tap the status bar to reveal the search field at the top of the screen, then enter your search text. You can also search audio content from the Home screen. See Searching on page 27.

Display audio controls while in another app: Double-click the Home button □, then swipe the multitasking bar to the right. Swipe right again to display a volume control and the AirPlay button ▭ (when in range of an Apple TV or AirPlay speakers).

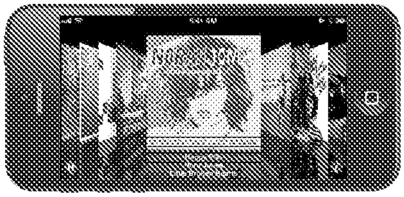


Display audio controls while the screen is locked: Double-click the Home button □.

Play music on AirPlay speakers or Apple TV: Tap ▭. See AirPlay on page 30.

Cover Flow

When you rotate iPhone, your music content appears in Cover Flow.



Browse albums in Cover Flow: Drag left or right.

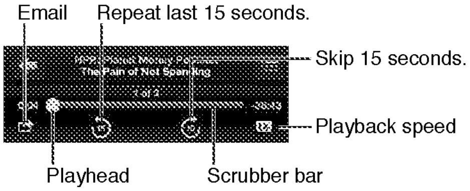
- *See the tracks on an album:* Tap the album artwork or ⓘ. Drag up or down to scroll; tap a track to play it.
- *Return to the artwork:* Tap the title bar, or tap ⓘ again.

Podcasts and audiobooks

On iPhone 5, podcast and audiobook controls and info appear on the Now Playing screen when you begin playback.

Note: The Podcasts app is available for free in the App Store. See Chapter 31, Podcasts, on page 113. If you install the Podcasts app, podcast content and controls are removed from Music.

Show or hide the controls and info (iPhone 4S or earlier): Tap the center of the screen.



Get more podcast episodes: Tap Podcasts (tap More first, if Podcasts isn't visible), then tap a podcast to see available episodes. To download more episodes, tap Get More Episodes.

Hide lyrics and podcast info: Go to Settings > Music, then turn off Lyrics & Podcasts Info.

Playlists

Create a playlist: View Playlists, tap Add Playlist near the top of the list, then enter a title. Tap ⊕ to add songs and videos, then tap Done.

Edit a playlist: Select the playlist to edit, then tap Edit.

- *Add more songs:* Tap +.
- *Delete a song:* Tap ⊖. Deleting a song from a playlist doesn't delete it from iPhone.
- *Change the song order:* Drag ≡.

New and changed playlists are copied to your iTunes library the next time you sync iPhone with your computer, or via iCloud if you've subscribed to iTunes Match.

Clear or delete a playlist: Select the playlist, then tap Clear or Delete.

Delete a song from iPhone: In Songs, swipe the song, then tap Delete.

The song is deleted from iPhone, but not from your iTunes library on your Mac or PC, or from iCloud.

Genius

A Genius playlist is a collection of songs from your library that go together. Genius is a free service, but it requires an Apple ID.

A Genius Mix is a selection of songs of the same kind of music, recreated from your library each time you listen to the mix.

Use Genius on iPhone: Turn on Genius in iTunes on your computer, then sync iPhone with iTunes. Genius Mixes are synced automatically, unless you manually manage your music. You can also sync Genius playlists.

Browse and play Genius Mixes: Tap Genius (tap More first, if Genius isn't visible). Swipe left or right to access other mixes. To play a mix, tap ▶.

Make a Genius playlist: View Playlists, then tap Genius Playlist and choose a song. Or, from the Now Playing screen, tap the screen to display the controls, then tap ⚛.

- *Replace the playlist using a different song:* Tap New and pick a song.
- *Refresh the playlist:* Tap Refresh.
- *Save the playlist:* Tap Save. The playlist is saved with the title of the song you picked and marked by ⚛.

Edit a saved Genius playlist: Tap the playlist, then tap Edit.

- *Delete a song:* Tap ⊖.
- *Change the song order:* Drag ≡.

Delete a saved Genius playlist: Tap the Genius playlist, then tap Delete.

Genius playlists created on iPhone are copied to your computer when you sync with iTunes.

Note: Once a Genius playlist is synced to iTunes, you can't delete it directly from iPhone. Use iTunes to edit the playlist name, stop syncing, or delete the playlist.

Siri and Voice Control

You can use Siri (iPhone 4S or later) or Voice Control to control music playback. See Chapter 4, Siri, on page 36 and Voice Control on page 26.

Use Siri or Voice Control: Press and hold the Home button □.

- *Play or pause music:* Say "play" or "play music." To pause, say "pause," "pause music," or "stop." You can also say "next song" or "previous song."
- *Play an album, artist, or playlist:* Say "play," then say "album," "artist," or "playlist" and the name.
- *Shuffle the current playlist:* Say "shuffle."
- *Find out more about the current song:* Say "what's playing," "who sings this song," or "who is this song by."
- *Use Genius to play similar songs:* Say "Genius" or "play more songs like this."

iTunes Match

iTunes Match stores your music library in iCloud—including songs imported from CDs—and lets you play your collection on iPhone and your other iOS devices and computers. iTunes Match is offered as a paid subscription. To find out where it's available, see support.apple.com/kb/HT5085.

Subscribe to iTunes Match: In iTunes on your computer, go to Store > Turn On iTunes Match, then click the Subscribe button.

Once you subscribe, iTunes adds your music, playlists, and Genius Mixes to iCloud. Your songs that match music already in the iTunes Store are automatically available in iCloud. Other songs are uploaded. You can download and play matched songs at up to iTunes Plus quality (256 kbps DRM-free AAC), even if your original was of lower quality. For more information, see www.apple.com/icloud/features.

Turn on iTunes Match: Go to Settings > Music.

Turning on iTunes Match removes synced music from iPhone, and disables Genius Mixes and Genius Playlists.

Note: If "Use Cellular Data for iTunes" in Settings > General > Cellular is on, cellular data charges may apply.

Songs are downloaded to iPhone when you play them. You can also download songs and albums manually.

Download a song or album to iPhone: While browsing, tap ☁.

Note: When iTunes Match is on, downloaded music is automatically removed from iPhone when space is needed, starting with the oldest and least played songs. iCloud icons (☁) reappear for removed songs and albums, showing that the songs and albums are still available through iCloud, but not stored locally on iPhone.

Manually remove a song or album: Swipe sideways across the song or album, then tap Delete.

Show only music that's been downloaded from iCloud: Go to Settings > Music, then turn off Show All Music (available only when iTunes Match is turned on).

Manage your devices using iTunes Match or Automatic Downloads: In iTunes on your computer, go to Store > View My Apple ID. Sign in, then click Manage Devices in the "iTunes in the Cloud" section.

Home Sharing

Home Sharing lets you play music, movies, and TV shows from the iTunes library on your Mac or PC. iPhone and your computer must be on the same Wi-Fi network.

Note: Home Sharing requires iTunes 10.2 or later, available at www.itunes.com/download. Bonus content, such as digital booklets and iTunes Extras, can't be shared.

Play music from your iTunes library on iPhone:

1. In iTunes on your computer, choose File > Home Sharing > Turn On Home Sharing. Log in, then click Create Home Share.
2. On iPhone, go to Settings > Music, then log in to Home Sharing using the same Apple ID and password.
3. In Music, tap More, then tap Shared and choose your computer's library.

Return to content on iPhone: Tap Shared and choose My iPhone.

Music settings

Go to Settings > Music to set options for Music, including:

- Shake to Shuffle
- Sound Check (to normalize the volume level of your audio content)
- Equalization (EQ)

  Note: EQ affects all sound output, including the headset jack and AirPlay. EQ settings generally apply only to music played from the Music app.

  The Late Night setting applies to all audio output—video as well as music. Late Night compresses the dynamic range of the audio output, reducing the volume of loud passages and increasing the volume of quiet passages. You might want to use this setting when listening to music on an airplane or in some other noisy environment, for example.
- Lyrics and podcast info
- Grouping by album artist
- iTunes Match
- Home Sharing

Set the volume limit: Go to Settings > Music > Volume Limit, then adjust the volume slider.

Note: In European Union countries, you can limit the maximum headset volume to the European Union recommended level. Go to Settings > Music > Volume Limit, then turn on EU Volume Limit.

Restrict changes to the volume limit: Go to Settings > General > Restrictions > Volume Limit, then tap Don't Allow Changes.

Messages

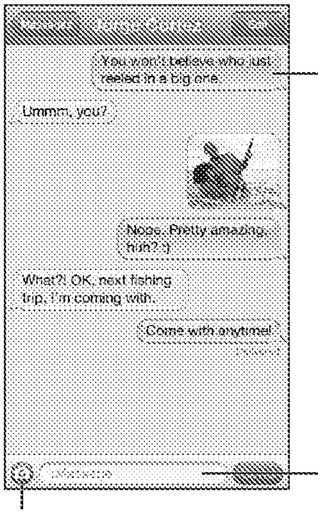
9

Sending and receiving messages

WARNING: For important information about avoiding distraction while driving, see Important safety information on page 146.

Messages lets you exchange text messages with other SMS and MMS devices via your cellular connection, and with other iOS devices using iMessage.

iMessage is an Apple service that lets you send unlimited messages over Wi-Fi (as well as cellular connections) to other iOS and OS X Mountain Lion users. With iMessage, you can see when other people are typing, and let them know when you've read their messages. iMessages are displayed on all of your iOS devices logged in to the same account, so you can start a conversation on one of your devices, and continue it on another device. iMessages are encrypted for security.



You won't believe who just reeled in a big one.

Ummm, you?

Nope. Pretty amazing, huh? :)

What?! OK, next fishing trip, I'm coming with.

Come with anytime!

Blue indicates an iMessage conversation.

Tap to enter text.

Tap the attach media button to include a photo or video.

Start a text conversation: Tap ✎, then tap ⊕ and choose a contact, search your contacts by entering a name, or enter a phone number or email address manually. Enter a message, then tap Send.

An alert badge ❗ appears if a message can't be sent. Tap the alert in a conversation to try sending the message again. Double-tap to send the message as an SMS text message.

Resume a conversation: Tap the conversation in the Messages list.

Use picture characters: Go to Settings > General > Keyboard > Keyboards > Add New Keyboard, then tap Emoji to make that keyboard available. Then while typing a message, tap 🌐 to bring up the Emoji keyboard. See Special input methods on page 144.

See a person's contact info: Scroll to the top (tap the status bar) to see actions you can perform, such as making a FaceTime call.

See earlier messages in the conversation: Scroll to the top (tap the status bar). Tap Load Earlier Messages if needed.

Send messages to a group (iMessage and MMS): Tap ✎, then enter multiple recipients. With MMS, group messaging must also be turned on in Settings > Messages, and replies are sent only to you—they aren't copied to the other people in the group.

Managing conversations

Conversations are saved in the Messages list. A blue dot ● indicates unread messages. Tap a conversation to view or continue it.

Forward a conversation: Tap Edit, select parts to include, then tap Forward.

Edit a conversation: Tap Edit, select the parts to delete, then tap Delete. To clear all text and attachments without deleting the conversation, tap Clear All.

Delete a conversation: In the Messages list, swipe the conversation, then tap Delete.

Search a conversation: Tap the top of the screen to display the search field, then enter the text you're looking for. You can also search conversations from the Home screen. See Searching on page 27.

Add someone to your contacts list: Tap a phone number in the Messages list, then tap "Add to Contacts."

Sharing photos, videos, and other info

With iMessage or MMS, you can send and receive photos and videos, and send locations, contact info, and voice memos. The size limit of attachments is determined by your service provider—iPhone may compress photo and video attachments when needed.

Send a photo or video: Tap ◎.

Send a location: In Maps, tap ◎ for a location, tap Share Location, then tap Message.

Send contact info: In Contacts, choose a contact, tap Share Contact, then tap Message.

Send a voice memo: In Voice Memos, tap ☰, tap the voice memo, tap Share, then tap Message.

Save a photo or video you receive to your Camera Roll album: Tap the photo or video, then tap ⬆.

Copy a photo or video: Touch and hold the attachment, then tap Copy.

Add someone to your contacts from the Messages list: Tap the phone number or email address, tap the status bar to scroll to the top, then tap "Add Contact."

Save contact info you receive: Tap the contact bubble, then tap Create New Contact or "Add to Existing Contact."

Messages settings

Go to Settings > Messages to set options for Messages, including:

- Turning iMessage on or off
- Notifying others when you've read their messages
- Specifying an Apple ID or email address to use with Messages
- SMS and MMS options
- Showing the Subject field
- Showing the character count

Manage notifications for messages: See Do Not Disturb and Notifications on page 132.

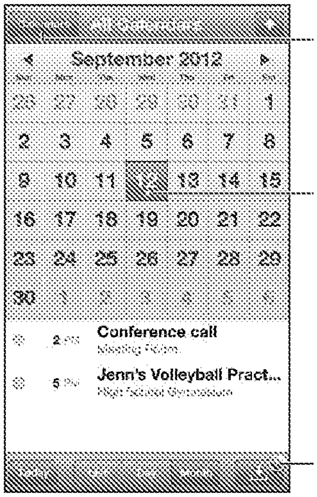
Set the alert sound for incoming text messages: See Sounds on page 139.

Calendar

10

At a glance

iPhone makes it easy to stay on schedule. You can view calendars individually, or view several calendars at once.



Change calendars or accounts.

A day with a dot has events.

View invitations.

View or edit an event: Tap the event. You can:

- Set a primary and secondary alert
- Change the event's date, time, or duration
- Move an event to a different calendar
- Invite others to attend events on iCloud, Microsoft Exchange, and CalDAV calendars
- Delete the event

You can also move an event by holding it down and dragging it to a new time, or by adjusting the grab points.

Add an event: Tap + and enter event information, then tap Done.

- *Set the default calendar for new events:* Go to Settings > Mail, Contacts, Calendars > Default Calendar.
- *Set default alert times for birthdays and events:* Go to Settings > Mail, Contacts, Calendars > Default Alert Times.

Search for events: Tap List, then enter text in the search field. The titles, invitees, locations, and notes for the calendars you're viewing are searched. You can also search calendar events from the Home screen. See Searching on page 27.

Set the calendar alert tone: Go to Settings > Sounds > Calendar Alerts.

View by week: Rotate iPhone sideways.

Import events from a calendar file: If you receive an .ics calendar file in Mail, open the message and tap the calendar file to import all of the events it contains. You can also import an .ics file published on the web by tapping a link to the file. Some .ics files subscribe you to a calendar instead of adding events to your calendar. See Working with multiple calendars below.

If you have an iCloud account, a Microsoft Exchange account, or a supported CalDAV account, you can send and receive meeting invitations.

Invite others to an event: Tap an event, tap Edit, then tap Invitees to select people from Contacts.

Respond to an invitation: Tap an invitation in the calendar. Or tap ✉ to display the Event screen, then tap an invitation. You can view information about the organizer and other invitees. If you add comments (which may not be available for all calendars) your comments can be seen by the organizer but not other attendees.

Accept an event without marking the time as reserved: Tap the event, then tap Availability and select "free." The event stays on your calendar, but doesn't appear as busy to others who send you invitations.

Working with multiple calendars

You can view individual calendars, or several calendars at once. You can subscribe to iCloud, Google, Yahoo!, or iCalendar calendars, as well as your Facebook events and birthdays.

Turn on iCloud, Google, Exchange, or Yahoo! calendars: Go to Settings > Mail, Contacts, Calendars, tap an account, then turn on Calendar.

Add a CalDAV account: Go to Settings > Mail, Contacts, Calendars, tap Add an Account, then tap Other. Under Calendars, tap Add CalDAV Account.

View Facebook events: Go to Settings > Facebook, then sign in to your Facebook account and turn on access to Calendar.

Select calendars to view: Tap Calendars, then tap to select the calendars you want to view. The events for all selected calendars appear in one view.

View the Birthdays calendar: Tap Calendars, then tap Birthdays to include birthdays from your Contacts with your events. If you've set up a Facebook account, you can also include your Facebook friends' birthdays.

You can subscribe to any calendar that uses the iCalendar (.ics) format. Supported calendar-based services include iCloud, Yahoo!, Google, and the Calendar application in OS X. You can read events from a subscribed calendar on iPhone, but you can't edit events or create new ones.

Subscribe to a calendar: Go to Settings > Mail, Contacts, Calendars, then tap Add Account. Tap Other, then tap Add Subscribed Calendar. Enter the server and filename of the .ics file to subscribe to. You can also subscribe to an iCalendar (.ics) calendar published on the web, by tapping a link to the calendar.

Sharing iCloud calendars

You can share an iCloud calendar with other iCloud users. When you share a calendar, others can view it, and you can let them add or change events, too. You can also share a read-only version that anyone can view.

Create an iCloud calendar: Tap Calendars, tap Edit, then tap Add Calendar.

Share an iCloud calendar: Tap Calendars, tap Edit, then tap the iCloud calendar you want to share. Tap Add Person, then choose someone from Contacts. The person will receive an email invitation to join the calendar, but they need an Apple ID and iCloud account in order to accept.

Turn off notifications for shared calendars: Go to Settings > Mail, Contacts, Calendars and turn off Shared Calendar Alerts.

Change a person's access to a shared calendar: Tap Calendars, tap Edit, tap the shared calendar, then tap a person you're sharing with. You can turn off their ability to edit the calendar, resend the invitation to join the calendar, or stop sharing the calendar with them.

Share a read-only calendar with anyone: Tap Calendars, tap Edit, then tap the iCloud calendar you want to share. Turn on Public Calendar, then tap Share Link to copy or send the URL for the calendar. Anyone can use the URL to subscribe to your calendar using a compatible app, such as Calendar for iOS or OS X.

Calendar settings

There are several settings in Settings > Mail, Contacts, Calendars that affect Calendar and your calendar accounts. These include:

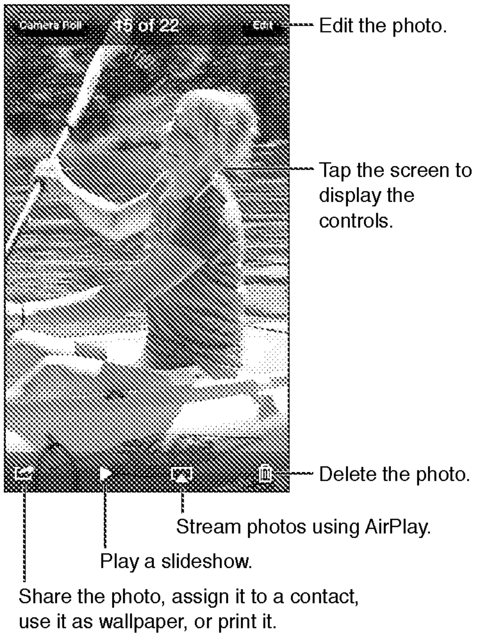
- Syncing of past events (future events are always synced)
- Alert tone played for new meeting invitations
- Calendar time zone support, to show dates and times using a different time zone

Photos 11

Viewing photos and videos

Photos lets you view photos and videos on iPhone, in your:

- Camera Roll album—photos and videos you took on iPhone, or saved from an email, text message, webpage, or screenshot
- Photo Stream albums—photos in My Photo Stream and your shared photo streams (see Photo Stream on page 71)
- Photo Library and other albums synced from your computer (see Syncing with iTunes on page 16)


Camera Roll
15 of 22
Edit
Edit the photo.
Tap the screen to display the controls.
Delete the photo.
Stream photos using AirPlay.
Play a slideshow.
Share the photo, assign it to a contact, use it as wallpaper, or print it.

View photos and videos: Tap an album, then tap a thumbnail.

- *See the next or previous photo or video:* Swipe left or right.
- *Zoom in or out:* Double-tap or pinch.
- *Pan a photo:* Drag it.
- *Play a video:* Tap ▶ in the center of the screen. To change between full-screen and fit-to-screen viewing, double-tap the screen.

Albums you sync with iPhoto 8.0 (iLife '09) or later, or Aperture v3.0.2 or later, can be viewed by events or by faces. You can also view photos by location, if they were taken with a camera that supports geotagging.

View a slideshow: Tap a thumbnail, then tap ▶. Select options, then tap Start Slideshow. To stop the slideshow, tap the screen. To set other options, go to Settings > Photos & Camera.

Stream a slideshow or video to a TV: See AirPlay on page 30.

Organizing photos and videos

Create an album: Tap Albums, tap +, enter a name, then tap Save. Select items to add to the album, then tap Done.

Note: Albums created on iPhone aren't synced back to your computer.

Add items to an album: When viewing thumbnails, tap Edit, select items, then tap Add To.

Manage albums: Tap Edit:

- *Rename an album:* Select the album, then enter a new name.
- *Rearrange albums:* Drag ≡.
- *Delete an album:* Tap ⊖.

Only albums created on iPhone can be renamed or deleted.

Photo Stream

With Photo Stream, a feature of iCloud (see iCloud on page 15), photos you take on iPhone automatically appear on your other devices set up with Photo Stream, including your Mac or PC. Photo Stream also lets you share select photos with friends and family, directly to their devices or on the web.

About Photo Stream

When Photo Stream is turned on, photos you take on iPhone (as well as any other photos added to your Camera Roll) appear in your photo stream after you leave the Camera app and iPhone is connected to the Internet via Wi-Fi. These photos appear in the My Photo Stream album on iPhone and on your other devices set up with Photo Stream.

Turn on Photo Stream: Go to Settings > iCloud > Photo Stream.

Photos added to your photo stream from your other iCloud devices also appear in My Photo Stream. iPhone and other iOS devices can keep up to 1000 of your most recent photos in My Photo Stream. Your computers can keep all your Photo Stream photos permanently.

Note: Photo Stream photos don't count against your iCloud storage.

Manage photo stream contents: In a photo stream album, tap Edit.

- *Save photos to iPhone:* Select the photos, then tap Save.
- *Share, print, copy, or save photos to your Camera Roll album:* Select the photos, then tap Share.
- *Delete photos:* Select the photos, then tap Delete.

Note: Although deleted photos are removed from photo streams on your devices, the original photos remain in the Camera Roll album on the device they originated from. Photos saved to a device or computer from a photo stream are also not deleted. To delete photos from Photo Stream, you need iOS 5.1 or later on iPhone and your other iOS devices. See support.apple.com/kb/HT4486.

Shared photo streams

Shared photo streams let you share selected photos with just the people you choose. iOS 6 and OS X Mountain Lion users can subscribe to your shared photo streams, view the latest photos you've added, "like" individual photos, and leave comments—right from their devices. You can also create a public website for a shared photo stream, to share your photos with others over the web.

Note: Shared photo streams work over both Wi-Fi and cellular networks. Cellular data charges may apply.

Turn on Shared Photo Streams: Go to Settings > iCloud > Photo Stream.

Create a shared photo stream: Tap Photo Stream, then tap +. To invite other iOS 6 or OS X Mountain Lion users to subscribe to your shared photo stream, enter their email addresses. To post the photo stream on icloud.com, turn on Public Website. Name the album, then tap Create.

Add photos to a shared photo stream: Select a photo, tap ⇪, tap Photo Stream, then select the shared photo stream. To add several photos from an album, tap Edit, select the photos, then tap Share.

Delete photos from a shared photo stream: Tap the shared photo stream, tap Edit, select the photos, then tap Delete.

Edit a shared photo stream: Tap Photo Stream, then tap ⚙. You can:

- Rename the photo stream
- Add or remove subscribers, and resend an invitation
- Create a public website, and share the link
- Delete the photo stream

Sharing photos and videos

You can share photos in email, text messages (MMS or iMessage), photo streams, Twitter posts, and Facebook. Videos can be shared in email and text messages (MMS or iMessage), and on YouTube.

Share or copy a photo or video: Choose a photo or video, then tap ⇪. If you don't see ⇪, tap the screen to show the controls.

The size limit of attachments is determined by your service provider. iPhone may compress photo and video attachments, if necessary.

You can also copy photos and videos, and then paste them into an email or text message (MMS or iMessage).

Share or copy multiple photos and videos: While viewing thumbnails, tap Edit, select the photos or videos, then tap Share.

Save a photo or video from:

- *Email:* Tap to download it if necessary, tap the photo or touch and hold the video, then tap Save.
- *Text message:* Tap the item in the conversation, tap ⇪, than tap Save to Camera Roll.
- *Webpage (photo only):* Touch and hold the photo, then tap Save Image.

Photos and videos that you receive, or that you save from a webpage, are saved to your Camera Roll album.

Printing photos

Print to AirPrint-enabled printers:

- *Print a single photo:* Tap ↪, then tap Print.
- *Print multiple photos:* While viewing a photo album, tap Edit, select the photos, tap Share, then tap Print.

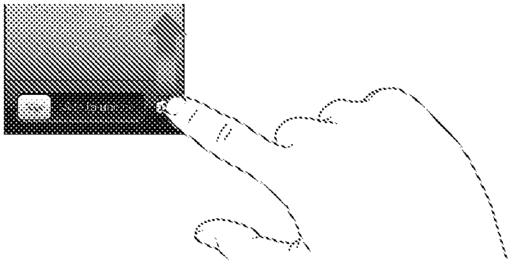
See Printing with AirPrint on page 30.

Camera

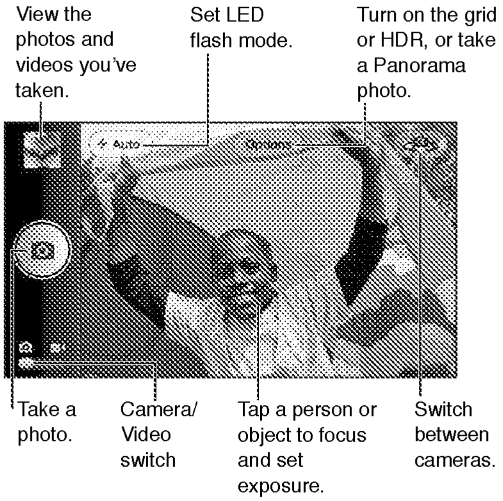
12

At a glance

To quickly open Camera when iPhone is locked, swipe 📷 up.



With iPhone, you can take both still photos and videos. In addition to the iSight camera on the back, there's a FaceTime camera on the front for FaceTime calls and self-portraits. An LED flash on the back gives you extra light when you need it.


View the photos and videos you've taken.
Set LED flash mode.
Turn on the grid or HDR, or take a Panorama photo.
Auto
Options
Take a photo.
Camera/ Video switch
Tap a person or object to focus and set exposure.
Switch between cameras.

A rectangle briefly appears where the camera is focused and setting the exposure. When you photograph people with iPhone 4S or later, iPhone uses face detection to automatically focus on and balance the exposure across up to 10 faces. A rectangle appears for each face detected.

Take a photo: Tap 📷 or press either volume button.

- *Zoom in or out:* Pinch the screen (iSight camera only).

Take a panorama photo (iPhone 4S or later): Tap Options, then tap Panorama. Point iPhone where you want to start, then tap 📷. Pan slowly in the direction of the arrow, holding iPhone steady. Try to keep the arrow directly on top of the horizontal line. When you finish, tap Done.

- *Reverse the panning direction:* Tap the arrow.

Record a video: Switch to 📹, then tap ● or press either volume button to start or stop recording.

- *Capture a still photo while recording:* Tap 📷.

When you take a photo or start a video recording, iPhone makes a shutter sound. You can control the volume with the volume buttons, or mute the sound using the Ring/Silent switch.

Note: In some countries, muting iPhone does not prevent the shutter sound.

If Location Services is turned on, photos and videos are tagged with location data that can be used by other apps and photo-sharing websites. See Privacy on page 140.

Set the focus and exposure:

- *Set the focus and exposure for the next shot:* Tap the object on the screen. Face detection is temporarily turned off.
- *Lock the focus and exposure:* Touch and hold the screen until the rectangle pulses. AE/AF Lock is displayed at the bottom of the screen, and the focus and exposure remain locked until you tap the screen again.

Take a screenshot: Press and release the Sleep/Wake button and the Home button □ at the same time. The screenshot is added to your Camera Roll album.

HDR photos

HDR (iPhone 4 or later) combines three separate exposures into a single "high dynamic range" photo. For best results, iPhone and the subject should be stationary.

Turn on HDR: Tap Option, then set HDR. When HDR is on, the flash is turned off.

Keep the normal photo in addition to the HDR version: Go to Settings > Photos & Camera. When you keep both versions, ✓ HDR appears in the upper-left corner of the HDR photo when viewed in your Camera Roll album with the controls visible.

Viewing, sharing, and printing

The photos and videos you take with Camera are saved in your Camera Roll album. If you have Photo Stream turned on, new photos also appear in your Photo Stream album and are streamed to your other iOS devices and computers. See Photo Stream on page 71.

View your Camera Roll album: Swipe to the right, or tap the thumbnail image. You can also view your Camera Roll album in the Photos app.

- *Show or hide the controls while viewing a photo or video:* Tap the screen.
- *Share a photo or video:* Tap ↗. To send multiple photos or videos, tap ↗ while viewing thumbnails, select the items, then tap Share.
- *Print a photo:* Tap ↗. See Printing with AirPrint on page 30.
- *Delete a photo or video:* Tap 🗑.

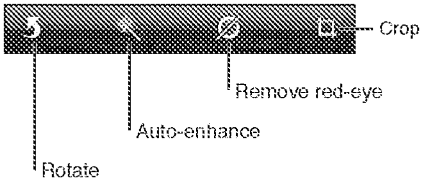
Return to the camera: Tap 📷.

Upload photos and videos to your computer: Connect iPhone to your computer.

- *Mac:* Select the photos and videos you want, then click the Import or Download button in iPhoto or other supported photo application on your computer.
- *PC:* Follow the instructions that came with your photo application.

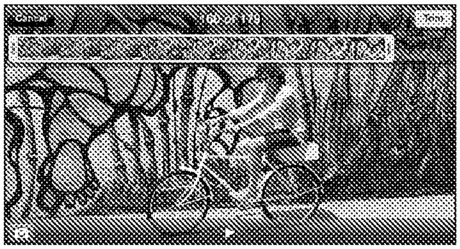
If you delete photos or videos from iPhone when you upload them to your computer, they're removed from your Camera Roll album. You can use the Photos settings pane in iTunes to sync photos and videos to the Photos app on iPhone (videos can be synced only with a Mac). See Syncing with iTunes on page 16.

Editing photos and trimming videos



Edit a photo: While viewing a photo in full screen, tap Edit, then tap a tool.

- *Auto-enhance:* Enhancing improves a photo's overall darkness or lightness, color saturation, and other qualities. If you decide against the enhancement, tap the tool again (even if you saved the changes).
- *Remove red-eye:* Tap each eye that needs correcting.
- *Crop:* Drag the corners of the grid, drag the photo to reposition it, then tap Crop. To set a specific ratio, tap Constrain.



Trim a video: While viewing a video, tap the screen to display the controls. Drag either end of the frame viewer at the top, then tap Trim.

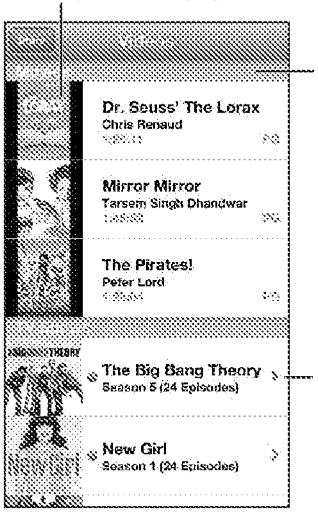
Important: If you choose Trim Original, the trimmed frames are permanently deleted from the original video. If you choose "Save as New Clip," a new trimmed video clip is saved in your Camera Roll album and the original video is unaffected.

Videos

13

Use the Videos app to watch movies, TV shows, and music videos. To watch video podcasts, install the free Podcasts app from the App Store. See Chapter 31, Podcasts, on page 113. To watch videos you record using Camera on iPhone, open the Photos app.

Tap a video to play it.



Dr. Seuss' The Lorax
Chris Renaud

Mirror Mirror
Tarsem Singh Dhandwar

The Pirates!
Peter Lord

The Big Bang Theory
Season 5 (24 Episodes)

New Girl
Season 1 (24 Episodes)

Swipe down to search.

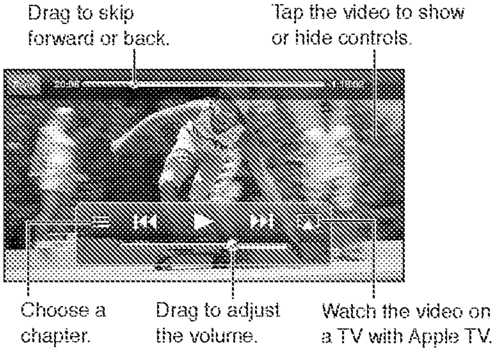
See additional episodes of a series.

> *WARNING:* For important information about avoiding hearing loss, see Important safety information on page 146.

Get videos:

- *Buy or rent videos from the iTunes store (not available in all areas):* Open the iTunes app on iPhone and tap Videos. See Chapter 22, iTunes Store, on page 94.
- *Transfer videos from your computer:* Connect iPhone, then sync videos in iTunes on your computer. See Syncing with iTunes on page 16.
- *Stream videos from your computer:* Turn on Home Sharing in iTunes on your computer. Then, on iPhone, go to Settings > Videos and enter the Apple ID and password you used to set up Home Sharing on your computer. Then, open Videos on iPhone and tap Shared at the top of the list of videos.

Convert a video to work with iPhone: If you try to add a video from iTunes to iPhone and a message says the video can't play on iPhone, you can convert the video. Select the video in your iTunes library and choose File > Create New Version > "Create iPod or iPhone Version." Then add the converted video to iPhone.


Drag to skip forward or back.
Tap the video to show or hide controls.
Choose a chapter.
Drag to adjust the volume.
Watch the video on a TV with Apple TV.

Watch a video: Tap the video in the list of videos.

- *Scale the video to fill the screen or fit to the screen:* Tap ▣ or ▣. Or, double-tap the video to scale without showing the controls.
- *Start over from the beginning:* If the video contains chapters, drag the playhead along the scrubber bar all the way to the left. If there are no chapters, tap ⏮.
- *Skip to the next or previous chapter (if available):* Tap ⏭ or ⏮. You can also press the center button or equivalent on a compatible headset two times (skip to next) or three times (skip to previous).
- *Rewind or fast-forward:* Touch and hold ⏮ or ⏭.
- *Select a different audio language (if available):* Tap 💬, then choose a language from the Audio list.
- *Show or hide subtitles (if available):* Tap 💬, then choose a language, or Off, from the Subtitles list.
- *Show or hide closed captioning (if available):* Go to Settings > Videos.
- *Watch the video on a TV:* See Connecting iPhone to a TV or other device on page 30.

Set a sleep timer: Open the Clock app and tap Timer, then swipe to set the number of hours and minutes. Tap When Timer Ends and choose Stop Playing, tap Set, then tap Start to start the timer. When the timer ends, iPhone stops playing music or video, closes any other open app, and then locks itself.

Delete a video: Swipe left or right over the video in the list. Deleting a video (other than a rented movie) from iPhone doesn't delete it from your iTunes library.

Important: If you delete a rented movie from iPhone, it's deleted permanently and cannot be transferred back to your computer.

When you delete a video (other than a rented movie) from iPhone, it isn't deleted from your iTunes library on your computer, and you can sync the video back to iPhone later. If you don't want to sync the video back to iPhone, set iTunes to not sync the video. See Syncing with iTunes on page 16.

Maps

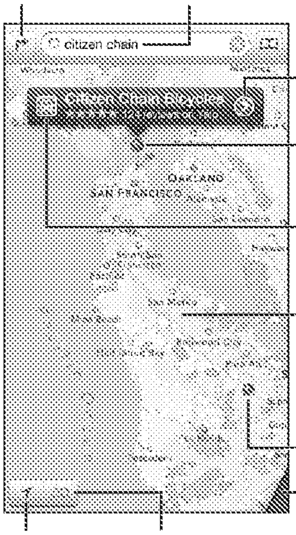
14

Finding locations

WARNING: For important information about navigating safely and avoiding distraction while driving, see Important safety information on page 146.

Get directions.

Enter a search.



Get more info.

Tap a pin to display the info banner.

Quick driving directions

Double-tap to zoom in; tap with two fingers to zoom out. Or, pinch.

Current location

Show your current location.

Flyover (3D in standard view)

Print, show traffic, list results, or choose the view.

Important: Maps, directions, 3D, Flyover, and location-based apps depend on data services. These data services are subject to change and may not be available in all areas, resulting in maps, directions, 3D, Flyover, or location-based information that may be unavailable, inaccurate, or incomplete. Compare the information provided on iPhone to your surroundings, and defer to posted signs to resolve any discrepancies. Some Maps features require Location Services. See Privacy on page 140.

Find a location: Tap the search field, then type an address or other information, such as:

- Intersection ("8th and market")
- Area ("greenwich village")
- Landmark ("guggenheim")
- Zip code
- Business ("movies,""restaurants san francisco ca,""apple inc new york")

Or, tap one of the suggestions in the list below the search field.

Navigate maps:

- *Move up or down, left or right:* Drag the screen.
- *Rotate the map:* Rotate two fingers on the screen. A compass appears in the upper-right corner to show the map's orientation.
- *Return to the north-facing orientation:* Tap ◉.

Find the location of a contact, or of a bookmarked or recent search: Tap ▤.

Get and share info about a location: Tap the pin to display the info banner, then tap ⓘ. When available, you can get reviews and photos from Yelp. You can also get directions, contact the business, visit the home page, add the business to your contacts, share the location, or bookmark the location.

- *Read reviews:* Tap Reviews. To use other Yelp features, tap the buttons beneath the reviews.
- *See photos:* Tap Photos.
- *Email, text, tweet, or post a location to Facebook:* Tap Share Location. To tweet or post to Facebook, you must be signed in to your accounts. See Sharing on page 29.

Use the drop pin to mark a location: Touch and hold the map until the drop pin appears.

Choose standard, hybrid, or satellite view: Tap the lower-right corner.

Report a problem: Tap the lower-right corner.

Getting directions

Get driving directions: Tap ↱, tap 🚗, enter the starting and ending locations, then tap Route. Or, choose a location or a route from the list, when available. If multiple routes appear, tap the one you want to take. Tap Start to begin.

- *Hear turn-by-turn directions (iPhone 4S or later):* Tap Start.

  Maps follows your progress and speaks turn-by-turn directions to your destination. To show or hide the controls, tap the screen.

  If iPhone auto-locks, Maps stays onscreen and continues to announce instructions. You can also open another app and continue to get turn-by-turn directions. To return to Maps, tap the banner across the top of the screen.
- *View turn-by-turn directions (iPhone 4 or earlier):* Tap Start, then swipe left to see the next instruction.
- *Return to the route overview:* Tap Overview.
- *View the directions as a list:* Tap ☰ on the Overview screen.
- *Stop turn-by-turn directions:* Tap End.

Get quick driving directions from your current location: Tap ↱ on the banner of your destination, then tap Directions To Here.

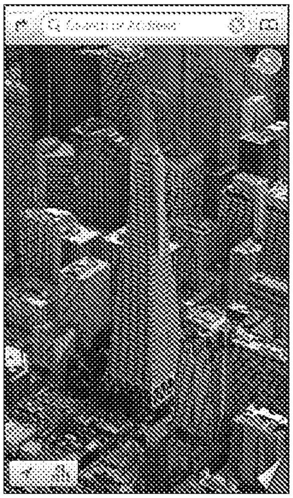
Get walking directions: Tap ↱, tap 🚶, enter the starting and ending locations, then tap Route. Or, choose a location or a route from the list, when available. Tap Start, then swipe left to see the next instruction.

Get public transit directions: Tap ↱, tap 🚌, enter the starting and ending locations, then tap Route. Or, choose a location or a route from the list, when available. Download and open the routing apps for the transit services you want to use.

Show traffic conditions: Tap the bottom-right corner of the screen, then tap Show Traffic. Orange dots show slowdowns, and red dots show stop-and-go traffic. To see an incident report, tap a marker.

3D and Flyover

On iPhone 4S or later, use 3D (standard view) or Flyover (satellite or hybrid view) for three-dimensional views of many cities around the world. You can navigate in the usual ways, and zoom in to see buildings. You can also adjust the camera angle.



The Transamerica Pyramid Building is a registered service mark of Transamerica Corporation.

Use 3D or Flyover: Zoom in until **3D** or 🏙 becomes active, then tap the button. Or, drag two fingers up. You can switch between 3D and Flyover by tapping the lower-right corner and changing views.

Adjust the camera angle: Drag two fingers up or down.

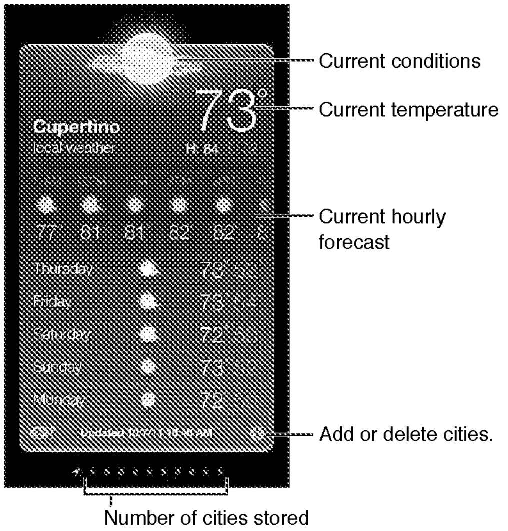
Maps settings

Set options for Maps: Go to Settings > Maps. Settings include:

- Navigation voice volume (iPhone 4S or later)
- Miles or kilometers for distance
- Language and size of labels

Weather 15

Get the current temperature and six-day forecast for one or more cities around the world, with hourly forecasts for the next 12 hours. Weather also uses Location Services to get the forecast for your current location.



If the weather board is light blue, it's daytime in that city. Dark purple indicates nighttime.

Manage your list of cities: Tap ⓘ, then add a city or make other changes. Tap Done when you finish.

- *Add a city:* Tap +. Enter a city or zip code, then tap Search.
- *Rearrange the order of cities:* Drag ≡ up or down.
- *Delete a city:* Tap ⊖, then tap Delete.
- *Choose Fahrenheit or Celsius:* Tap °F or °C.

See weather for another city: Swipe left or right.

The leftmost screen shows your local weather.

View the current hourly forecast:

- *iPhone 5:* Swipe the hourly display left or right.
- *iPhone 4S or earlier:* Tap Hourly.

Turn local weather on or off: Go to Settings > Privacy > Location Services. See Privacy on page 140.

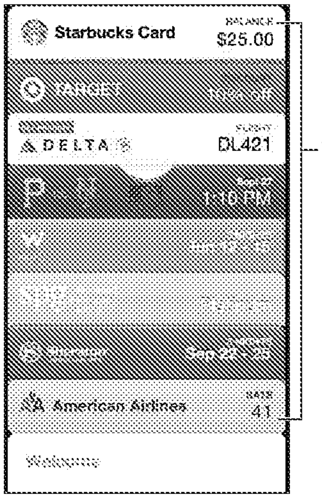
See information about a city at yahoo.com: Tap Ⓨ!.

Use iCloud to push your list of cities to your other iOS devices: Go to Settings > iCloud > Documents & Data, then turn on Documents & Data (it's on by default). See iCloud on page 15.

Passbook

16

Use Passbook to keep boarding passes, movie tickets, coupons, gift cards, and more, all in one place. Add passes from airlines, theaters, stores, and other participating merchants. Scan a pass on iPhone to check in for a flight, get in to a movie, or redeem a coupon.



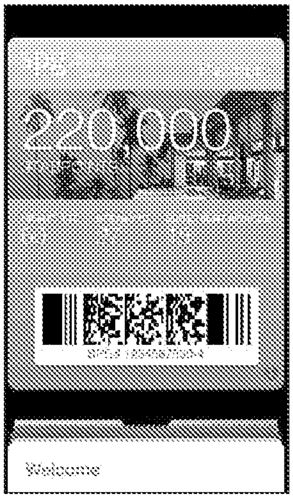
Tap a pass to view it.

Passes can include useful information, such as the balance on your coffee card, a coupon's expiration date, or your seat number for a concert. Some passes may also appear on your Lock screen when you wake iPhone at the right time or place—for example, when you reach the airport for a flight you're taking. (Location Services must be on in Settings > Privacy > Location Services.)

Add a pass to Passbook: You can add a pass from an app, email or Messages message, or website when you make a purchase or receive a coupon or gift. For example, tap Add to Passbook in the Fandango app when you purchase a ticket for a theater that supports scannable passes.

Find apps that support Passbook in the App Store: Tap "Apps for Passbook" on the Welcome pass. See www.itunes.com/passbookapps.

Use a pass: If an alert for a pass appears on the lock screen, slide the alert to display the pass. Or, open Passbook, select the pass, then present the barcode on the pass to the scanner.



View more information: Tap ⓘ.

Passes are usually updated automatically. To refresh a pass manually, tap ⓘ, then pull the pass downward.

Delete a pass: Tap ⓘ, then tap 🗑.

Prevent passes from appearing on your Lock screen: Go to Settings > General > Passcode Lock and tap Turn Passcode On. Then go to Allow Access When Locked and turn Passbook off. To prevent a specific pass from appearing on your Lock screen, tap ⓘ, then turn off Show On Lock Screen.

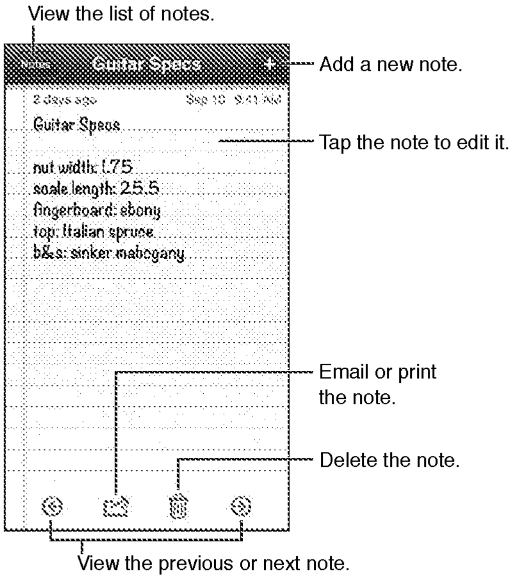
Set notification options: Go to Settings > Notifications > Passbook.

Include passes on your other iPhone or iPod touch: Go to Settings > iCloud and turn on Passbook.

Notes

17

Type notes on iPhone, and iCloud makes them available on your other iOS devices and Mac computers. You can also read and create notes in other accounts, such as Gmail or Yahoo!.



Notes

Guitar Specs

2 days ago

Sep 10 9:41 AM

Guitar Specs

nut width: 1.75
scale length: 25.5
fingerboard: ebony
top: Italian spruce
b&s: sinker mahogany

Use iCloud to keep your notes up to date on your iOS devices and Mac computers:

- *If you use an icloud.com, me.com, or mac.com email address for iCloud:* Go to Settings > iCloud and turn on Notes.
- *If you use a Gmail or other IMAP account for iCloud:* Go to Settings > Mail, Contacts, Calendars and turn on Notes for the account.

Choose the default account for new notes: Go to Settings > Notes.

Create a note in a specific account: Tap Accounts and select the account, then tap + to create the note. If you don't see the Accounts button, tap the Notes button first.

See only notes in a specific account: Tap Accounts and choose the account. If you don't see the Accounts button, tap Notes first.

Delete a note while viewing the list of notes: Swipe left or right across the note in the list.

Search for notes: While viewing the list of notes, scroll to the top of the list to reveal the search field. Tap in the field and type what you're looking for. You can also search for notes from the Home screen. See Searching on page 27.

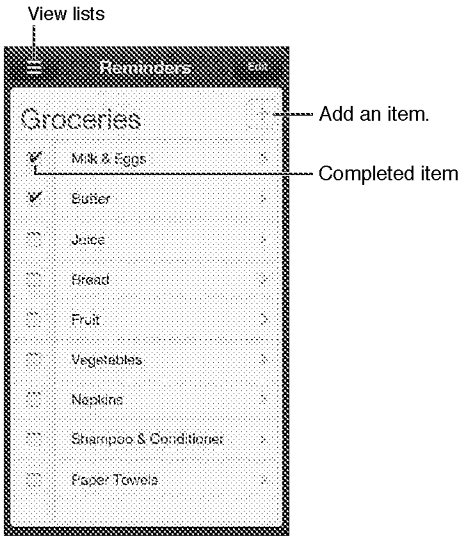
Print or email a note: While reading the note, tap ➦. To email the note, iPhone must be set up for email. See Setting up mail and other accounts on page 14.

Change the font: Go to Settings > Notes.

Reminders

18

Reminders lets you keep track of all the things you need to do.



Reminders Edit

Groceries

- ✓ Milk & Eggs
- ✓ Butter
- Juice
- Bread
- Fruit
- Vegetables
- Napkins
- Shampoo & Conditioner
- Paper Towels

See reminder details: Tap a reminder. You can:

- Change or delete it
- Set a due date
- Set a priority
- Add notes
- Move it to a different list

Reminders can alert you when you arrive at or leave a location.

Add a location alert: While entering a reminder, tap ⓘ, then turn on "Remind Me At a Location."

To use a different location, tap your current location. Locations in the list include addresses from your personal info card in Contacts, such as the home and work addresses you've added. To use a different address, tap Enter an Address.

Note: Location reminders are not available on iPhone 3GS. You cannot set locations for reminders in Microsoft Exchange and Outlook accounts.

Search your reminders: Tap ≡ to see the search field, or search from the Home screen. Reminders are searched by name. You can also use Siri to find or add reminders.

Turn off reminder notifications: Go to Settings > Notifications. For information, see Do Not Disturb and Notifications on page 132.

Set the tone played for notifications: Go to Settings > Sounds.

Keep your reminders up to date on other devices: Go to Settings > iCloud, then turn on Reminders. To keep up to date with Reminders on OS X Mountain Lion, turn on iCloud on your Mac, too. Some other types of accounts, such as Exchange, also support Reminders. Go to Settings > Mail, Contacts, Calendars and turn on Reminders for the accounts you want to use.

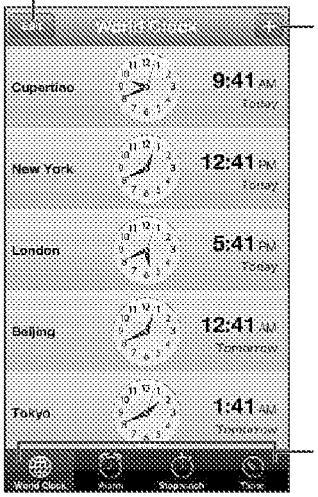
Set a default list for new reminders: Go to Settings > Mail, Contacts, Calendars, then under Reminders, tap Default List.

Clock

19

You can add clocks to show the time in other major cities and time zones around the world.

Delete clocks or change their order.



Cupertino 9:41 AM Today

New York 12:41 PM Today

London 5:41 PM Today

Beijing 12:41 AM Tomorrow

Tokyo 1:41 AM Tomorrow

Add a clock.

View clocks, set an alarm, time an event, or set a timer.

Add a clock: Tap ✚, then type the name of a city or choose a city from the list. If you don't see the city you're looking for, try a major city in the same time zone.

Organize clocks: Tap Edit, then drag ≡ to move or tap ⊖ to delete.

Set an alarm: Tap Alarm, then tap ✚.

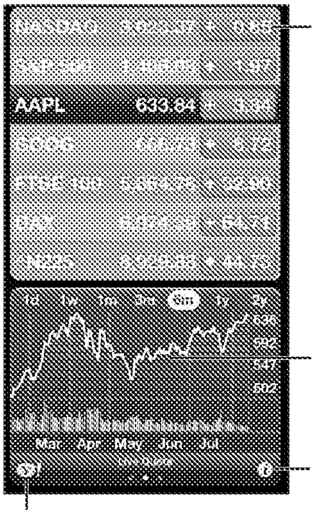
Change an alarm: Tap Edit, then tap > to change settings or tap ⊖ to delete.

Set a sleep timer for iPhone: Set a timer, tap When Timer Ends, and choose Stop Playing.

Stocks

20

Keep track of your stocks, see the change in value over time, and get news about your investments.



Tap to see percent change. Tap again to see market capitalization.

Swipe left or right to see stats or news articles.

Customize your stock list.

Go to yahoo.com for more info.

Manage your stock list: Tap ⓘ, then add stocks or make other changes. When you finish, tap Done.

- *Add an item:* Tap +. Enter a symbol, company name, fund name, or index, then tap Search.
- *Delete an item:* Tap ⊖.
- *Rearrange the order of items:* Drag ≡ up or down.

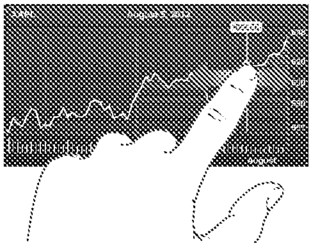
View stock info:

- *Switch the display to percentage change, price change, or market capitalization:* Tap any of the values along the right side of the screen.
- *See the summary, chart, or news:* Swipe the info beneath the stock list. Tap a news headline to view the article in Safari. To change the chart's time period, tap 1d, 1w, 1m, 3m, 6m, 1y, or 2y.
- *Add a news article to your reading list:* Touch and hold the news headline, then tap Add to Reading List.
- *See more stock information at yahoo.com:* Tap Y!.

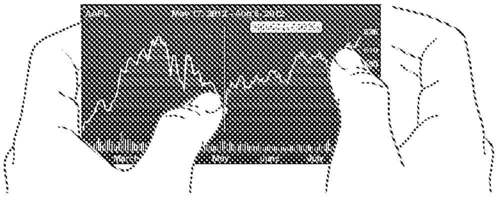
Quotes may be delayed 20 minutes or more, depending upon the reporting service. To display your stocks as a ticker in Notification Center, see Notifications on page 28.

View a full-screen chart: Rotate iPhone to landscape orientation.

- *See the value at a specific date or time:* Touch the chart with one finger.



- *See the difference in value over time:* Touch the chart with two fingers.

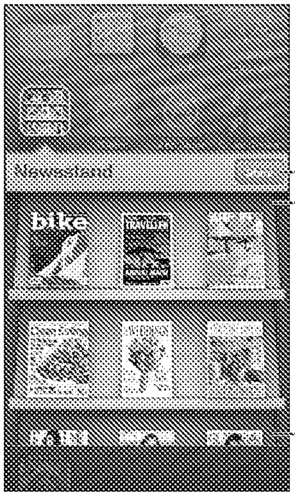


Use iCloud to keep your stock list up to date on your iOS devices: Go to Settings > iCloud > Documents & Data, then turn on Documents & Data (it's on by default). See iCloud on page 15.

Newsstand

21

Newsstand organizes your magazine and newspaper apps and lets you know when new issues are ready for reading.



Find Newsstand apps.

Touch and hold a publication to rearrange.

Newsstand organizes magazine and newspaper apps with a shelf for easy access.

Find Newsstand apps: Tap Newsstand to reveal the shelf, then tap Store. When you purchase a newsstand app, it's added to your shelf. After the app is downloaded, open it to view its issues and subscription options. Subscriptions are In-App purchases, billed to your store account.

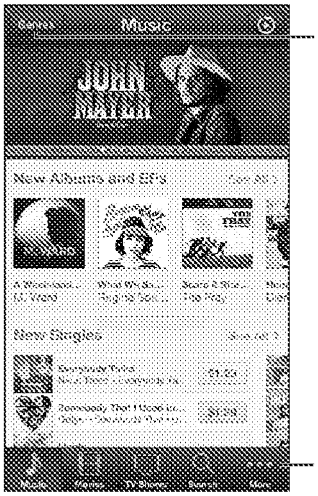
Turn off automatically downloading new issues: Go to Settings > Newsstand. If an app supports it, Newsstand downloads new issues when connected to Wi-Fi.

iTunes Store

22

At a glance

Use the iTunes Store to add music, movies, and TV shows to iPhone.



Browse

See purchases, downloads, and more.

Use iTunes Store to:

- Find music, TV shows, movies, tones, and more, by browsing or searching
- See your personal Genius recommendations
- Download previous purchases

Note: You need an Internet connection and an Apple ID to use the iTunes Store.

Browse content: Tap one of the categories. Tap Genres to refine the listings. To see more information about an item, tap it.

Search for content: Tap Search, then tap the search field and enter one or more words, then tap Search.

Preview an item: Tap a song or video to play a sample.

Purchase an item: Tap the item's price (or tap Free), then tap again to buy it. If you already purchased the item, "Download" appears instead of the price and you won't be charged again. To see the progress of items being downloaded, tap Downloads at the bottom of the screen.

Rent a movie: In some areas, certain movies are available to rent. You have 30 days to begin viewing a rented movie. Once you've started playing it, you can watch it as many times as you want in 24 hours. After these time limits, the movie is deleted.

Download a previous purchase: Tap More, then tap Purchased. To automatically download purchases made on other devices, go to Settings > iTunes & App Stores.

Redeem a gift card or code: Tap any category (such as music), scroll to the bottom, then tap Redeem.

Send a gift: While viewing the item you want to give as a gift, tap ↗, then tap Gift.

View or edit your account: Go to Settings > iTunes & App Stores, tap your Apple ID, then tap View Apple ID. Tap an item to edit it. To change your password, tap the Apple ID field.

Turn iTunes Match on or off: Go to Settings > iTunes & App Stores. iTunes Match is a subscription service that stores all of your music in iCloud so you can access it from wherever you are.

Sign in using a different Apple ID: Go to Settings > iTunes & App Stores, tap your account name, then tap Sign Out. The next time you download an app, you can enter a different Apple ID.

Download purchases using the cellular network: Go to Settings > iTunes & App Stores > Use Cellular Data. Downloading purchases and using iTunes Match over the cellular network may incur charges from your carrier.

Changing the browse buttons

You can replace and rearrange the buttons at the bottom of the screen. For example, if you often download tones but don't watch many TV shows, you could replace those buttons.

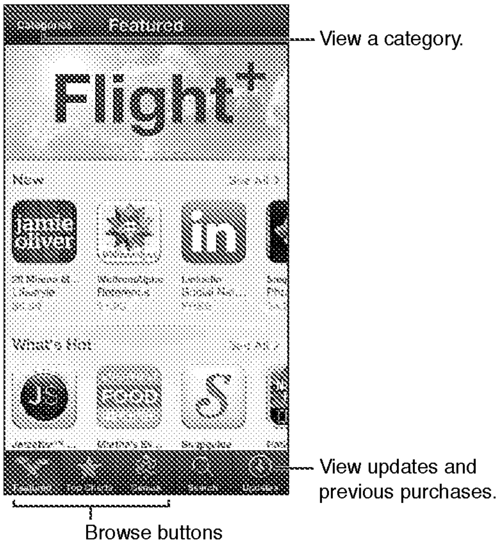
Change the browse buttons: Tap More, tap Edit, then drag a button to the bottom of the screen, over the button you want to replace. When you finish, tap Done.

App Store

23

At a glance

Use the App Store to browse, purchase, and download apps to iPhone.



View a category.

View updates and previous purchases.

Browse buttons

Use the App Store to:

- Find new free or purchased apps by browsing or searching
- Download updates and previous purchases
- Redeem a gift card or download code
- Recommend an app to a friend
- Manage your App Store account

Note: You need an Internet connection and an Apple ID to use the App Store.

Purchase an app: Tap the app's price (or tap Free), then tap Buy Now. If you already purchased the app, "install" appears instead of the price. You won't be charged to download it again. While an app is being downloaded, its icon appears on the Home screen with a progress indicator.

Download a previous purchase: Tap Updates, then tap Purchased. To automatically download new purchases made on other devices, go to Settings > iTunes & App Stores.

Download updated apps: Tap Updates. Tap an app to read about the new version, then tap Update to download it. Or tap Update All to download all the apps in the list.

Redeem a gift card or download code: Tap Featured, scroll to the bottom, then tap Redeem.

Tell a friend about an app: Find the app, then tap ⤴ and select how you want to share it.

View and edit your account: Go to Settings > iTunes & App Stores, tap your Apple ID, then tap View Apple ID. You can turn subscribe to iTunes newsletters, and view Apple's privacy policy. To change your password, tap the Apple ID field.

Sign in using a different Apple ID: Go to Settings > iTunes & App Stores, tap your account name, then tap Sign Out. The next time you download an app, you can enter a different Apple ID.

Create a new Apple ID: Go to Settings > iTunes & App Stores, then tap Create New Apple ID and follow the onscreen instructions.

Download purchases using the cellular network: Go to Settings > iTunes & App Stores > Use Cellular Data. Downloading purchases over the cellular network may incur charges from your carrier. Newsstand apps update only over Wi-Fi.

Deleting apps

Delete an App Store app: Touch and hold its icon on the Home screen until the icon starts to jiggle, then tap ⊗. You can't delete built-in apps. When you finish, press the Home button ▢.

Deleting an app also deletes all of its data. You can re-download any app you've purchased from the App Store, free of charge.

For information about erasing all of your apps, data, and settings, see Reset on page 138.

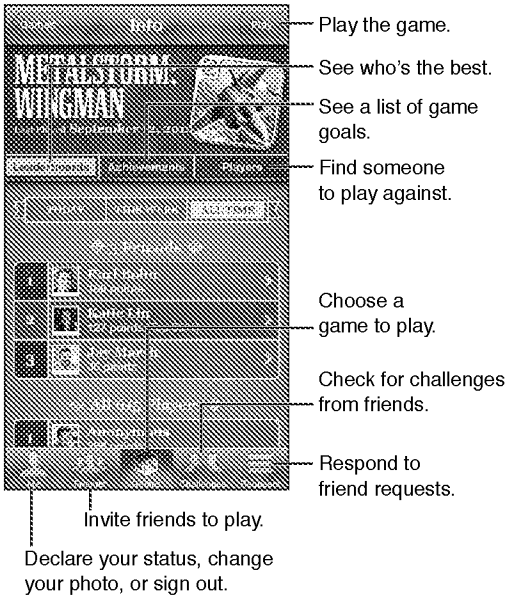
Game Center

24

At a glance

Game Center lets you play your favorite games with friends who have an iPhone, iPad, iPod touch, or a Mac with OS X Mountain Lion.

> *WARNING:* For important information about avoiding repetitive motion injuries, see Important safety information on page 146.


Play the game.
See who's the best.
See a list of game goals.
Find someone to play against.
Choose a game to play.
Check for challenges from friends.
Respond to friend requests.
Invite friends to play.
Declare your status, change your photo, or sign out.

Sign in: Open Game Center. If you see your nickname and photo at the top of the screen, you're already signed in. If not, enter your Apple ID and password, then tap Sign In. You can use the same Apple ID you use for iCloud, App Store, or iTunes Store purchases, or tap Create New Account if you want a separate Apple ID for gaming.

Purchase a game: Tap Games, then tap a recommended game or tap Find Game Center Games.

Play a game: Tap Games, choose a game, then tap Play.

Return to Game Center after playing: Press the Home button □, then tap Game Center on the Home screen.

Sign out: Tap Me, tap the Account banner, then tap Sign Out. You don't need to sign out each time you quit Game Center.

Playing with friends

Invite friends to a multiplayer game: Tap Friends, choose a friend, choose a game, then tap Play. If the game allows or requires more players, choose additional players, then tap Next. Send your invitation, then wait for the others to accept. When everyone is ready, start the game. If a friend isn't available or doesn't respond to your invitation, you can tap Auto-Match to have Game Center find another player for you, or tap Invite Friend to invite someone else.

Send a friend request: Tap Friends or Requests, tap +, then enter your friend's email address or Game Center nickname. To browse your contacts, tap ⊕. To add several friends in one request, type Return after each address.

Challenge someone to outdo you: Tap one of your scores or achievements, then tap Challenge Friends.

See the games a friend plays and check your friend's scores: Tap Friends, tap your friend's name, then tap Games or Points.

Purchase a game your friend has: Tap Friends, then tap the name of your friend. Tap the game in your friend's list of games, then tap the price at the top of the screen.

See a list of a friend's friends: Tap Friends, tap the friend's name, then tap Friends just below their picture.

Remove a friend: Tap Friends, tap a name, then tap Unfriend.

Keep your email address private: Turn off Public Profile in your Game Center account settings. See "Game Center settings" below.

Disable multiplayer activity or friend requests: Go to Settings > General > Restrictions and turn off Multiplayer Games or Adding Friends. If the switches are disabled, tap Enable Restrictions (at the top) first.

Report offensive or inappropriate behavior: Tap Friends, tap the person's name, then tap "Report a Problem."

Game Center settings

Some Game Center settings are associated with the Apple ID you use to sign in. Others are in the Settings app on iPhone.

Change Game Center settings for your Apple ID: Sign in with your Apple ID, tap Me, tap the Account banner, then choose View Account.

Specify which notifications you want for Game Center: Go to Settings > Notifications > Game Center. If Game Center doesn't appear, turn on Notifications.

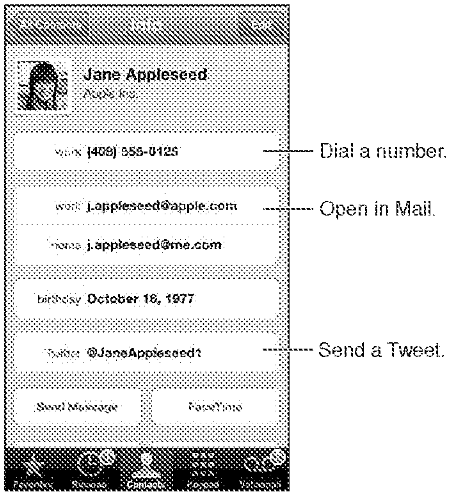
Change restrictions for Game Center: Go to Settings > General > Restrictions.

Contacts

25

At a glance

iPhone lets you easily access and edit your contact lists from personal, business, and organizational accounts.



Jane Appleseed

Apple Inc.

work (408) 555-0125

work j.appleseed@apple.com

home j.appleseed@me.com

birthday October 16, 1977

twitter @JaneAppleseed1

Dial a number.

Open in Mail.

Send a Tweet.

Set your My Info card: Go to Settings > Mail, Contacts, Calendars, then tap My Info and select the contact card with your name and information. The My Info card is used by Siri and other apps. Use the related persons fields to define relationships you want Siri to know about, so you can say things like "call my sister."

Search contacts: Tap the search field at the top of the contact list and enter your search. You can also search your contacts from the Home screen. See Searching on page 27.

Share a contact: Tap a contact, then tap Share Contact. You can send the contact info by email or message.

Add a contact: Tap +. You can't add contacts to a directory you're only viewing, such as a Microsoft Exchange Global Address List.

Add a contact to your Favorites list: Choose a contact, then scroll down and tap the Add to Favorites button. The Favorites list is used by Do Not Disturb. See Do Not Disturb and Notifications on page 132.

Add a phone number to Contacts when dialing: In Phone, tap Keypad, enter a number, then tap +👤. Tap Create New Contact or tap "Add to Existing Contact" and choose a contact.

Add a recent caller to Contacts: In Phone, tap Recents and tap ⊙ next to the number. Then tap Create New Contact, or tap "Add to Existing Contact" and choose a contact.

Delete a contact: Choose a contact, than tap Edit. Scroll down and tap Delete Contact.

Edit a contact: Choose a contact, then tap Edit. You can:

- *Add a new field:* Tap ⊕, then choose or enter a label for the field.
- *Change a field label:* Tap the label and choose a different one. To add a new field, tap Add Custom Label.
- *Change the ringtone or text tone for the contact:* Tap the ringtone or text tone field, then choose a new sound. To change the default tone for contacts, go to Settings > Sounds.
- *Change how iPhone vibrates for call or messages from the contact:* Tap the ringtone or text tone vibration field, then select a vibration pattern. If you don't see the vibration field, tap Edit and add it. For information about creating custom vibration patterns, see Sounds on page 139.
- *Assign a photo to the contact:* Tap Add Photo. You can take a photo with the camera or use an existing photo.
- *Update contact info using Twitter:* Go to Settings > Twitter > Update Contacts. Contacts are matched using email addresses. For friends that you're following, their contact card is updated with their Twitter user name and photo.
- *Update contact info using Facebook:* Go to Settings > Facebook > Update Contacts. Contacts are matched using email addresses. For each match in your friend list, their contact card is updated with their Facebook user name and photo.
- *Enter a pause in a telephone number:* Tap +*#, then tap Pause or Wait. Each pause lasts two seconds. Each wait stops dialing until you tap Dial again. Use these to automate dialing of an extension or passcode, for example.

Adding contacts

In addition to entering contacts, you can:

- *Use your iCloud contacts:* Go to Settings > iCloud, then turn on Contacts.
- *Import your Facebook Friends:* Go to Settings > Facebook, then turn on Contacts in the "Allow These Apps to Use Your Accounts" list. This creates a Facebook group in Contacts.
- *Access a Microsoft Exchange Global Address List:* Go to Settings > Mail, Contacts, Calendars, then tap your Exchange account and turn on Contacts.
- *Set up an LDAP or CardDAV account to access business or school directories:* Go to Settings > Mail, Contacts, Calendars > Add Account > Other. Then tap "Add LDAP Account" or "Add CardDAV Account" and enter the account information.
- *Sync contacts from your computer, Yahoo!, or Google:* In iTunes on your computer, turn on contact syncing in the device info pane. For information, see iTunes Help.
- *Import contacts from a SIM card (GSM):* Go to Settings > Mail, Contacts, Calendars > Import SIM Contacts.
- *Import contacts from a vCard:* Tap a .vcf attachment in an email or message, or on a webpage.

Search a GAL, CardDAV, or LDAP server: Tap Groups, tap the directory you want to search, then enter your search.

Save contact information from a GAL, LDAP, or CardDAV server: Search for the contact you want to add, then tap Add Contact.

Show or hide a group: Tap Groups then select the groups you want to see. This button only appears if you have more than one source of contacts.

When you have contacts from multiple sources, you might have multiple entries for the same person. To keep redundant contacts from appearing in the All Contacts list, contacts from different sources that have the same name are linked and displayed as a single *unified contact*. When you view a unified contact, the title Unified Info appears at the top of the screen.

Link a contact: Edit a contact, tap Edit, then tap Link Contact and choose the contact entry to link to.

Linked contacts aren't merged. If you change or add information in a unified contact, the changes are copied to each source account where that information already exists.

If you link contacts with different first or last names, the names on the individual cards won't change, but only one name appears on the unified card. To choose which name appears when you view the unified card, tap Edit, tap the linked card with the name you prefer, then tap Use This Name For Unified Card.

View contact information from a source account: Tap one of the source accounts.

Unlink a contact: Tap Edit, tap ⊖, then tap Unlink.

Contacts settings

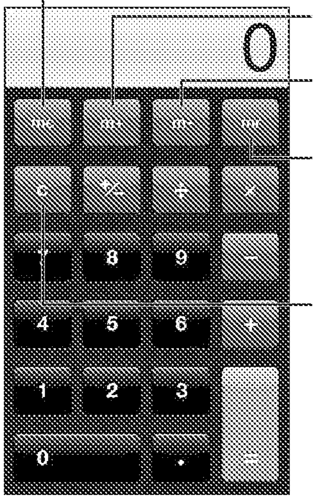
To change Contacts settings, go to Settings > Mail, Contacts, Calendars. Available options let you:

- Change how contacts are sorted
- Display contacts by first or last name
- Set a default account for new contacts
- Set your My Info card

Calculator

26

Tap numbers and functions in Calculator just as you would with a standard calculator.



Clear memory.

Add a number to memory.

Subtract a number from memory.

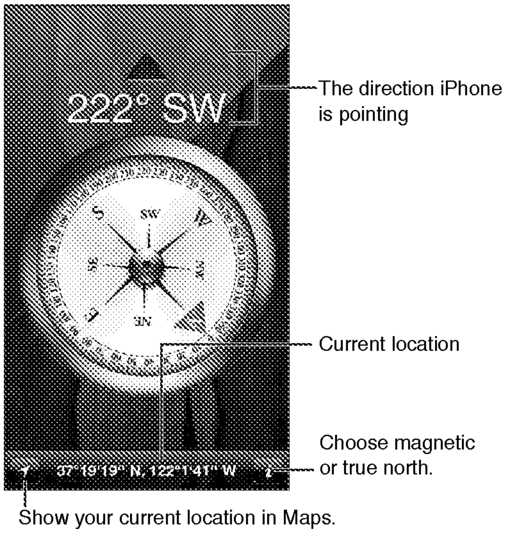
Get a number from memory (a white ring indicates a number is stored in memory).

Clear the display.

Use the scientific calculator: Rotate iPhone to landscape orientation.

Compass 27

Find a direction or your current heading, see your latitude and longitude, or show your location and heading in Maps.


222° SW
37°19'19" N, 122°1'41" W
The direction iPhone is pointing
Current location
Choose magnetic or true north.
Show your current location in Maps.

Find the direction your iPhone is pointing: Hold iPhone flat in your hand, level with the ground.

If Location Services is turned off when you open Compass, you may be asked to turn it on. You can use Compass without turning on Location Services. See Privacy on page 140.

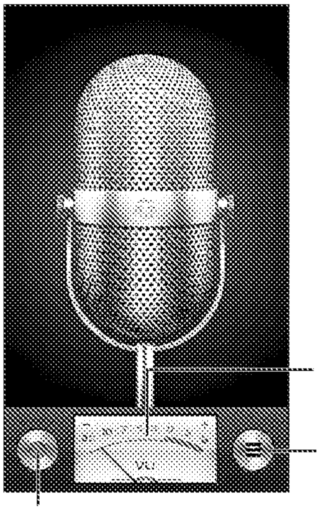
Important: The accuracy of the compass can be affected by magnetic or environmental interference; even the magnets in the iPhone earbuds can cause a deviation. Use the digital compass only for basic navigation assistance and don't rely on it to determine precise location, proximity, distance, or direction.

Voice Memos

28

At a glance

Voice Memos lets you use iPhone as a portable recording device using the built-in microphone, iPhone or Bluetooth headset mic, or supported external microphone.


Recording level
See your list of recordings.
Start, pause, or stop recording.
VU

Make a recording: Tap ● or press the center button on your headset. Tap ‖ to pause or ■ to stop recording, or press the center button on your headset.

Recordings using the built-in microphone are mono, but you can record stereo using an external stereo microphone that works with the iPhone headset jack, or with the Lightning connector (iPhone 5) or 30-pin dock connector (earlier iPhone models). Look for accessories marked with the Apple "Made for iPhone" or "Works with iPhone" logo.

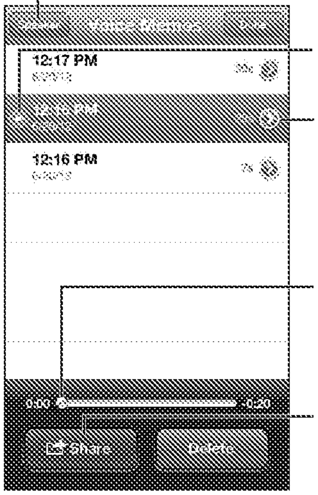
Adjust the recording level: Move the microphone closer to or further away from what you're recording. For better recording quality, the loudest level on the level meter should be between –3 dB and 0 dB.

Play or mute the start/stop tone: Use the iPhone volume buttons to turn the volume all the way down.

Use another app while recording: Press the Home button ○ and open an app. To return to Voice Memos, tap the red bar at the top of the screen.

Play a recording: Tap ☰, tap a recording, then tap ▶. Tap ❚❚ to pause.

Switch between the speaker and the receiver.



Listen to the recording.

Trim or relabel the recording.

Drag to skip to any point.

Attach the recording to an email or text message.

Trim a recording: Tap ⊙ next to the recording, then tap Trim Memo. Drag the edges of the audio region, then tap ▶ to preview. Adjust if necessary, then tap Trim Voice Memo to save. The portions you trim can't be recovered.

Sharing voice memos with your computer

You can sync voice memos with the primary iTunes library on your computer, then listen to memos on your computer or sync them with another iPhone or iPod touch.

When you delete a synced memo from iTunes, it stays on the device where it was recorded, but is deleted from any other iPhone or iPod touch you synced. If you delete a synced memo on iPhone, it's copied back to iPhone the next time you sync with iTunes, but you can't sync that copy back to iTunes a second time.

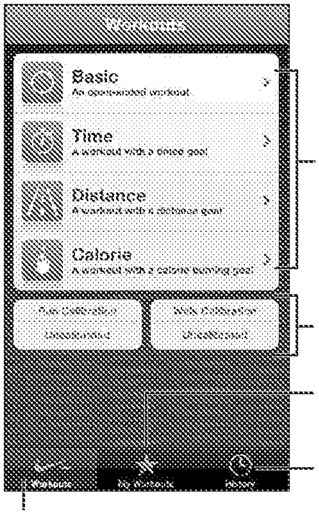
Sync voice memos with iTunes: Connect iPhone to your computer, then in iTunes select iPhone. Select Music at the top of the screen (between Apps and Movies), select Sync Music, select "Include voice memos," and click Apply.

Voice memos synced from iPhone to your computer appear in the Music list and in the Voice Memos playlist in iTunes. Memos synced from your computer appear in the Voice Memos app on iPhone, but not in the Music app.

Nike + iPod

29

With a Nike + iPod Sensor (sold separately), the Nike + iPod app provides audible feedback on your speed, distance, time elapsed, and calories burned during a run or walk.



Choose a workout type.

Calibrate based on your last workout.

Choose or create a custom workout.

Review your workout history.

Choose a standard workout.

The Nike + iPod app doesn't appear on the Home screen until you turn it on.

Turn on Nike + iPod: Go to Settings > Nike + iPod.

Nike + iPod collects workout data from a wireless sensor (sold separately) that you attach to your shoe. Before you use it the first time, you need to link your sensor to iPhone.

Link your sensor to iPhone: Attach the sensor to your shoe, then go to Settings > Nike + iPod > Sensor.

Start a workout: Tap Workouts, and choose a workout.

- *Pause a workout:* Wake iPhone and tap ⏸ on the lock screen. Tap ▶ when you're ready to continue.
- *End a workout:* Wake iPhone, tap ⏸, then tap End Workout.

Change workout settings: Go to Settings > Nike + iPod.

Calibrate Nike + iPod: Record a workout over a known distance of at least a quarter mile (400 meters). Then, after you tap End Workout, tap Calibrate on the workout summary screen and enter the actual distance you covered.

Reset to the default calibration: Go to Settings > Nike + iPod.

Send workout data to nikeplus.com: With iPhone connected to the Internet, open Nike + iPod, tap History, then tap "Send to Nike+."

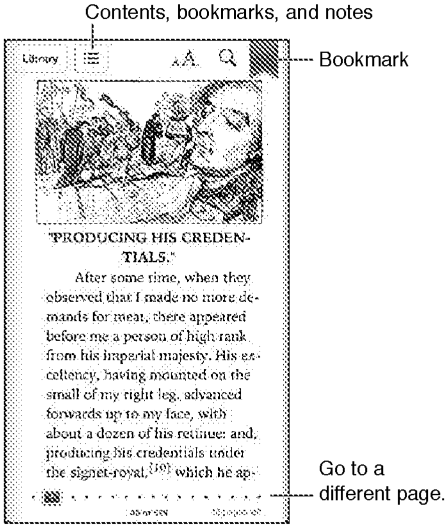
See your workouts on nikeplus.com: In Safari, go to nikeplus.com, log in to your account, and follow the onscreen instructions.

iBooks

30

At a glance

iBooks is a great way to read and buy books. Download the free iBooks app from the App Store, and then enjoy everything from classics to bestsellers.



"PRODUCING HIS CREDENTIALS."

After some time, when they observed that I made no more demands for meat, there appeared before me a person of high rank from his imperial majesty. His excellency, having mounted on the small of my right leg, advanced forwards up to my face, with about a dozen of his retinue: and, producing his credentials under the signet-royal,[10] which he ap-

To download the iBooks app and use the iBookstore, you need an Internet connection and an Apple ID.

Visit the iBookstore: In iBooks, tap Store to:

- Find books by browsing or searching
- Get a sample of a book to see if you like it
- Read and write reviews, and see current bestsellers
- Tell a friend about a book via Facebook, Twitter, iMessage, or email

Purchase a book: Find one you want, tap the price, then tap again to get it.

Get information about a book: You can read a summary of the book, read reviews, and try a sample of the book before buying it. After buying a book, you can write a review of your own.

Download a previous purchase: If you download a book you've previously purchased, you won't be charged again. To automatically download items purchased on other devices, go to Settings > iTunes & App Stores. For information about purchased books and iCloud, see Organizing the bookshelf on page 111.

Update a book: If there's an update to a book you've downloaded, a badge notifies you of the new version. To see and download the updated book, tap Purchased, then tap Updates.

Reading books

Each book has a particular set of features, based on its contents and format. Some of the features described below might not be available in the book you're reading.

Open a book: Tap the book you want to read. If you don't see it, swipe the shelf left or right to see other collections.

- *Show the controls:* Tap near the center of the page.
- *Enlarge an image:* Double-tap the image. In some books, touch and hold to display a magnifying glass you can use to view an image.
- *Go to a specific page:* Use the page navigation controls at the bottom of the screen. Or, tap 🔍 and enter a page number, then tap the page number in the search results.
- *Look up a word:* Double-tap a word, then tap Define in the menu that appears. Definitions aren't available for all languages.
- *View the table of contents:* Tap ☰. With some books, you can also pinch to see the the table of contents.
- *Add or remove a bookmark:* Tap 🔖. Tap again to remove the bookmark. You don't need to add a bookmark when you close the book, because iBooks remembers where you left off. You can have multiple bookmarks—to see them all, tap ☰, then tap Bookmarks.

Annotate a book: You can add notes and highlights to a book.

- *Add a highlight:* Double-tap a word, use the grab points to adjust the selection, then tap Highlight and choose a style.
- *Share highlighted text:* Tap the highlighted text, then tap ↗. If the book you're reading is from the iBookstore, a link to the book is included.
- *Remove a highlight:* Tap the highlighted text, then tap ⊘.
- *Add a note:* Double-tap a word, then tap Note.
- *Remove a note:* Delete its text. To remove the note and its highlight, tap the highlighted text, then tap ⊘.
- *See all your notes:* Tap ☰, then tap Notes. Tap ↗ to print or email your notes.
- *Delete notes:* Tap the center of the screen to display the controls, tap ☰, then tap Notes. Tap ↗, then tap Edit Notes. Select the notes you want to delete, then tap Delete.
- *Share your notes:* Tap the center of the screen to display the controls, tap ☰, then tap Notes. Tap ↗, then tap Edit Notes. Select the notes you want to share, then tap Share.
- *Share a link to a book:* Tap the center of the screen to display the controls, then tap ☰. Tap ↗, then tap Share Book.

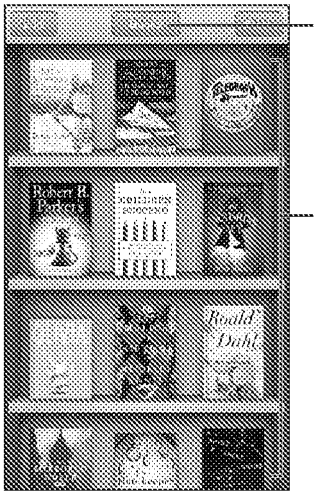
Change a book's appearance: Many books let you change the font, font size, and page color.

- *Change the font or font size:* Tap the center of the screen to display the controls, then tap AA. Some books allow you to change the font size only when iPhone is in portrait orientation.
- *Change the color of the page and text:* Tap the center of the screen to display the controls, tap AA, then tap Themes and choose White, Sepia, or Night. This setting applies to all books that support it.
- *Change the brightness:* Tap the center of the screen to display the controls, then tap ☼. If you don't see ☼, tap AA first.

- *Change how pages are displayed:* Tap the center of the screen to display the controls, tap aA, then tap Themes and choose Book, Full Screen, or Scroll.
- *Turn justification and hyphenation on or off:* Go to Settings > iBooks. PDFs and some books can't be justified or hyphenated.

Organizing the bookshelf

Use the bookshelf to browse your books and PDFs. You can also organize items into collections.



Move a book or PDF to a collection: Tap Edit. Select the items you want to move, then tap Move and select a collection.

View and manage collections: Tap the name of the current collection at the top of the screen. You can't edit or remove the built-in collections.

Sort the bookshelf: Tap the status bar to scroll to the top of the screen, then tap ☰ and select a sort method at the bottom of the screen.

Delete items from the bookshelf: Tap Edit, then tap each item that you want to delete, so that a checkmark appears. Tap Delete, then tap Done.

- *Delete this copy:* Removes the item from iPhone, but it still appears on the bookshelf and can be downloaded again.
- *Delete from all devices:* Removes the item from all of your iOS devices and from the bookshelf. You can download it again from Purchases in the iBookstore. See At a glance on page 109.

Search for a book: Go to the bookshelf. Tap the status bar to scroll to the top of the screen, then tap 🔍. Searching looks for the title and the author's name.

Download a book from iCloud: Books you've purchased that aren't on iPhone appear with an iCloud badge. To download the book, tap its cover. To see all of your purchases, go to the Purchased Books collection.

- *Hide purchases on the bookshelf:* To show or hide purchased books that aren't on iPhone, go to Settings > iBooks > Show All Purchases. You can download purchases from the iBookstore. See At a glance on page 109.

Syncing books and PDFs

Use iTunes to sync your books and PDFs between iPhone and your computer, and to buy books from the iTunes Store. When iPhone is connected to your computer, the Books pane lets you select which items to sync. You can also find DRM-free ePub books and PDFs on the web and add them to your iTunes library.

Sync a book or PDF to iPhone: In iTunes on your computer, choose File > Add to Library and select the file. Then sync.

Add a book or PDF to iBooks without syncing: If the book or PDF isn't too large, email it to yourself from your computer. Open the email message on iPhone, then touch and hold the attachment and choose "Open in iBooks."

Printing or emailing a PDF

You can use iBooks to email a copy of a PDF, or to print all or a portion of the PDF to an AirPrint printer.

Email a PDF: Open the PDF, tap ↗ then choose Email Document.

Print a PDF: Open the PDF, tap ↗ then choose Print. For more information, see Printing with AirPrint on page 30.

iBooks settings

iBooks stores your purchases, collections, bookmarks, notes, and current page information in iCloud, so you can read books seamlessly across all your iOS devices. iBooks saves information about all of your books when you open or quit the app. Information about individual books is also saved when you open or close the book.

Turn syncing on or off: Go to Settings > iBooks. You can sync collections and bookmarks.

Some books might access video or audio that's stored on the web. If iPhone has a cellular data connection, playing these files may incur carrier charges.

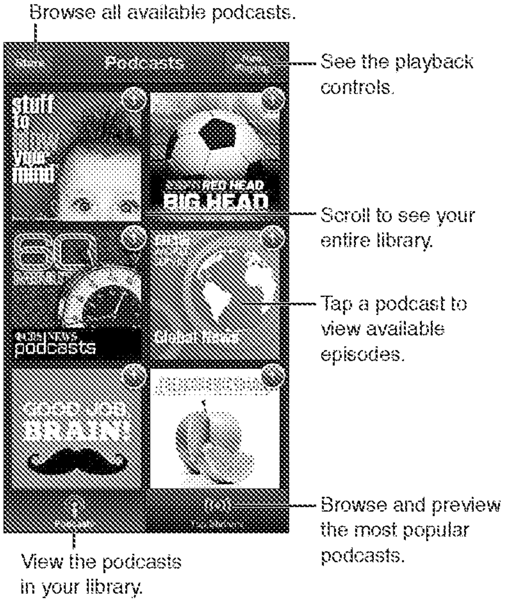
Turn online content access on or off: Go to Settings > iBooks > Online Content.

Change the direction the page turns when you tap the left margin: Go to Settings > iBooks > Both Margins Advance.

Podcasts

31

Download the free Podcasts app from the App Store, then browse, subscribe to, and play your favorite audio and video podcasts.



Browse all available podcasts.

See the playback controls.

Scroll to see your entire library.

Tap a podcast to view available episodes.

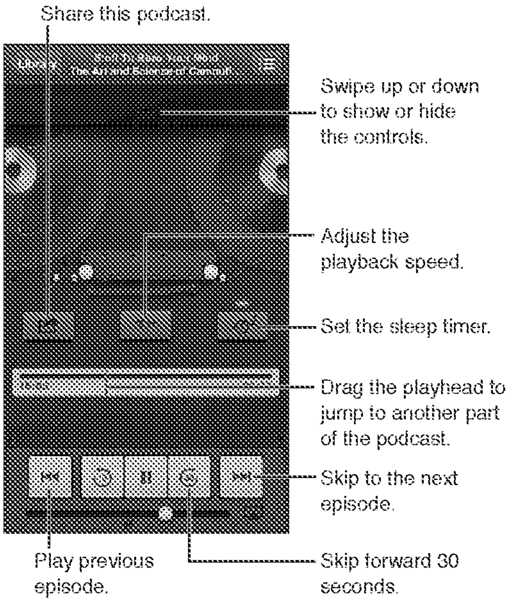
Browse and preview the most popular podcasts.

View the podcasts in your library.

Get podcasts:

- *Browse the full catalog:* Tap Catalog, then tap any podcast that interests you.
- *Browse the most popular podcasts:* Tap Top Stations (if you don't see it, tap Library first). Swipe left or right to change the category, or swipe up or down to browse the current category. Tap a podcast to preview the latest episode, or tap ⓘ to see a list of episodes.
- *Stream an episode:* Tap any episode.
- *Download an episode so you can listen to it when you're not connected to Wi-Fi:* Tap ⬇ next to any episode.
- *Subscribe to a podcast to always get the latest episode:* If you're browsing the catalog, tap a podcast to see the list of episodes, then tap Subscribe. If you've already downloaded an episode, tap the podcast in your library, then tap it again at the top of the list of episodes, and turn on Subscription.
- *Automatically get the latest episode of a subscribed podcast:* Tap the podcast in your library, tap it again at the top of the episode list, then turn on Auto-Download.

Control audio playback: To see all of the playback controls, swipe the artwork upward.


Share this podcast.
Library
Swipe up or down to show or hide the controls.
Adjust the playback speed.
Set the sleep timer.
Drag the playhead to jump to another part of the podcast.
Skip to the next episode.
Play previous episode.
Skip forward 30 seconds.

Control video playback: Tap the screen while you're watching a video podcast.

Accessibility 32

Accessibility features

iPhone incorporates these accessibility features:

- VoiceOver
- Call audio routing
- Siri voice assistant
- Zoom magnification
- Large Text
- Invert Colors
- Speak Selection
- Speak Auto-text
- Mono Audio and balance
- Hearing aids and Hearing Aid Mode
- Assignable ringtones and vibrations
- LED Flash for Alerts
- Guided Access
- AssistiveTouch
- Support for braille displays
- Playback of closed-captioned content

Turn on accessibility features using iPhone: Go to Settings > General > Accessibility.

Turn on accessibility features using iTunes: Connect iPhone to your computer and select iPhone in the iTunes device list. Click Summary, then click Configure Universal Access at the bottom of the Summary screen.

For more information about iPhone accessibility features, go to www.apple.com/accessibility.

Large Text can only be turned on or off in iPhone settings. See Large Text on page 125.

VoiceOver

VoiceOver describes aloud what appears onscreen, so you can use iPhone without seeing it.

VoiceOver tells you about each item on the screen as you select it. When you select an item, the VoiceOver cursor (a black rectangle) encloses it and VoiceOver speaks the name or describes the item.

Touch the screen or drag your fingers to hear different items on the screen. When you select text, VoiceOver reads the text. If you turn on Speak Hints, VoiceOver may tell you the name of the item and provide instructions—for example,"double-tap to open." To interact with items on the screen, such as buttons and links, use the gestures described in Learning VoiceOver gestures on page 118.

When you go to a new screen, VoiceOver plays a sound, then selects and speaks the first item on the screen (typically in the upper-left corner). VoiceOver also lets you know when the display changes to landscape or portrait orientation, and when the screen becomes locked or unlocked.

Note: VoiceOver speaks in the language specified in International settings, which may be influenced by the Region Format setting in Settings > General > International. VoiceOver is available in many languages, but not all.

VoiceOver basics

Important: VoiceOver changes the gestures you use to control iPhone. Once VoiceOver is turned on, you must use VoiceOver gestures to operate iPhone—even to turn VoiceOver off again and resume standard operation.

Turn VoiceOver on or off: Go to Settings > General > Accessibility > VoiceOver. You can also set Triple-click Home to turn VoiceOver on or off. See Triple-click Home on page 124.

Explore the screen: Drag your finger over the screen. VoiceOver speaks each item you touch. Lift your finger to leave an item selected.

- *Select an item:* Tap it, or lift your finger while dragging over it.
- *Select the next or previous item:* Swipe right or left with one finger. Item order is left-to-right, top-to-bottom.
- *Select the item above or below:* Use the rotor to turn on Vertical Navigation, then swipe up or down with one finger.
- *Select the first or last item on the screen:* Swipe up or down with four fingers.
- *Select an item by name:* Triple-tap with two fingers anywhere on the screen to open the Item Chooser. Then type a name in the search field, or swipe right or left to move through the list alphabetically, or tap the table index to the right of the list and swipe up or down to move quickly through the list of items.
- *Change the name of the selected item so it's easier to find:* Tap and hold with two fingers anywhere on the screen.
- *Speak the text of the selected item:* Set the rotor control to characters or words, then swipe down or up with one finger.
- *Turn spoken hints on or off:* Go to Settings > General > Accessibility > VoiceOver.
- *Include phonetic spelling:* Go to Settings > General > Accessibility > VoiceOver > Use Phonetics.
- *Speak the entire screen from the top:* Swipe up with two fingers.
- *Speak from the current item to the bottom of the screen:* Swipe down with two fingers.
- *Stop speaking:* Tap once with two fingers. Tap again with two fingers to resume speaking. Speaking resumes when you select another item.
- *Mute VoiceOver:* Triple-tap with three fingers. Triple-tap again with three fingers to turn speaking back on. To turn off only VoiceOver sounds, set the Ring/Silent switch to Silent. If an external keyboard is connected, you can also press the Control key on the keyboard to mute or unmute VoiceOver.

Adjust the speaking voice: You can adjust the characteristics of the VoiceOver speaking voice to make it easier for you to understand:

- *Change the speaking volume:* Use the volume buttons on iPhone. You can also add volume to the rotor and swipe up and down to adjust; see Using the VoiceOver rotor control on page 119.
- *Change the speaking rate:* Go to Settings > General > Accessibility > VoiceOver and drag the Speaking Rate slider. You can also add Speech Rate to the rotor, then swipe up or down to adjust.
- *Use pitch change:* VoiceOver uses a higher pitch when speaking the first item of a group (such as a list or table) and a lower pitch when speaking the last item of a group. Go to Settings > General > Accessibility > VoiceOver > Use Pitch Change.
- *Change the language for iPhone:* Go to Settings > General > International > Language. VoiceOver pronunciation of some languages is affected by Settings > General > International > Region Format.
- *Change pronunciation:* Set the rotor to Language, then swipe up or down. Language is available in the rotor only if you select more than one pronunciation.
- *Select the pronunciations available in the language rotor:* Go to Settings > General > Accessibility > VoiceOver > Language Rotor. To change the position of a language in the list, drag ≡ up or down.
- *Change the basic reading voice:* Go to Settings > General > Accessibility > VoiceOver > Use Compact Voice.

Using iPhone with VoiceOver

Unlock iPhone: Select the Unlock slide, then double-tap the screen.

"Tap" to activate the selected item: Double-tap anywhere on the screen.

"Double-tap" the selected item: Triple-tap anywhere on the screen.

Adjust a slider: Select the slider, then swipe up or down with one finger.

Use a standard gesture when VoiceOver is turned on: Double-tap and hold your finger on the screen. A series of tones indicates that normal gestures are in force. They remain in effect until you lift your finger, when VoiceOver gestures resume.

Scroll a list or area of the screen: Swipe up or down with three fingers. When paging through a list, VoiceOver speaks the range of items displayed (for example, "showing rows 5 through 10").

- *Scroll continuously through a list:* Double-tap and hold. When you hear a series of tones, move your finger up or down to scroll the list. Continuous scrolling stops when you lift your finger.
- *Use a list index:* Some lists have an alphabetical index along the right side. The index can't be selected by swiping between items; you must touch the index directly to select it. With the index selected, swipe up or down to move along the index. You can also double-tap, then slide your finger up or down.
- *Reorder a list:* You can change the order of items in some lists, such as the Rotor and Language Rotor items in Accessibility settings. Select ≡ on the right side of an item, double-tap and hold until you hear a sound, then drag up or down. VoiceOver speaks the item you've moved above or below, depending on the direction you're dragging.

Rearrange your Home screen: On the Home screen, select the icon you want to move. Double-tap and hold the icon, then drag it. VoiceOver speaks the row and column position as you drag the icon. Release the icon when it's in the location you want. You can drag additional icons. Drag an item to the left or right edge of the screen to move it to a different page of the Home screen. When you finish, press the Home button ○.

Speak the iPhone status information: Tap the top of the screen to hear information about the time, battery life, Wi-Fi signal strength, and more.

Speak notifications: Go to Settings > General > Accessibility > VoiceOver and turn on Speak Notifications. Notifications, including the text of incoming text messages, are spoken as they occur, even if iPhone is locked. Unacknowledged notifications are repeated when you unlock iPhone.

Turn the screen curtain on or off: Tap four times with three fingers. When the screen curtain is on, the screen contents are active even though the display is turned off.

Learning VoiceOver gestures

When VoiceOver is turned on, the standard touchscreen gestures have different effects. These and some additional gestures let you move around the screen and control individual items when they're selected. VoiceOver gestures include two- and three-finger gestures to tap or swipe. For best results when using two- and three-finger gestures, relax and let your fingers touch the screen with some space between them.

You can use different techniques to enter VoiceOver gestures. For example, you can enter a two-finger tap using two fingers from one hand, or one finger from each hand. You can also use your thumbs. Many find the "split-tap" gesture especially effective: instead of selecting an item and double-tapping, you can touch and hold an item with one finger, then tap the screen with another finger. Try different techniques to discover which works best for you.

If your gestures don't work, try quicker movements, especially for double-tapping and swiping gestures. To swipe, try quickly brushing the screen with your finger or fingers. When VoiceOver is turned on, the VoiceOver Practice button appears, which gives you a chance to practice VoiceOver gestures before proceeding.

Practice VoiceOver gestures: Go to Settings > General > Accessibility > VoiceOver, then tap VoiceOver Practice. When you finish practicing, tap Done. If you don't see the VoiceOver Practice button, make sure VoiceOver is turned on.

Here's a summary of key VoiceOver gestures:

Navigate and read

- *Tap:* Speak the item.
- *Swipe right or left:* Select the next or previous item.
- *Swipe up or down:* Depends on the Rotor Control setting. See Using the VoiceOver rotor control on page 119.
- *Two-finger tap:* Stop speaking the current item.
- *Two-finger flick up:* Read all from the top of the screen.
- *Two-finger flick down:* Read all from the current position.
- *Two-finger "scrub":* Move two fingers back and forth three times quickly (making a "z") to dismiss an alert or go back to the previous screen.
- *Three-finger swipe up or down:* Scroll one page at a time.
- *Three-finger swipe right or left:* Go to the next or previous page (such as the Home screen, Stocks, or Safari).
- *Three-finger tap:* Speak additional information, such as position within a list or whether text is selected.
- *Four-finger tap at top of screen:* Select the first item on the page.
- *Four-finger tap at bottom of screen:* Select the last item on the page.

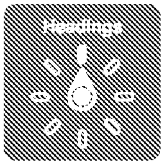
Activate

- *Double-tap:* Activate the selected item.
- *Triple-tap:* Double-tap an item.
- *Split-tap:* As an alternative to selecting an item and double-tapping to activate it, touch an item with one finger, and then tap the screen with another.
- *Double-tap and hold (1 second) + standard gesture:* Use a standard gesture. The double-tap and hold gesture tells iPhone to interpret the next gesture as standard. For example, you can double-tap and hold, and then without lifting your finger, drag your finger to slide a switch.
- *Two-finger double-tap:* Answer or end a call. Play or pause in Music, Videos, Voice Memos, or Photos. Take a photo in Camera. Start or pause recording in Camera or Voice Memos. Start or stop the stopwatch.
- *Two-finger double-tap and hold:* Change an item's label to make it easier to find.
- *Two-finger triple-tap:* Open the Item Chooser.
- *Three-finger triple-tap:* Mute or unmute VoiceOver.
- *Three-finger quadruple-tap:* Turn the screen curtain on or off.

Using the VoiceOver rotor control

Use the rotor to choose what happens when you swipe up or down with VoiceOver turned on.

Operate the rotor: Rotate two fingers on the iPhone screen around a point between them.



Change the options included in the rotor: Go to Settings > General > Accessibility > VoiceOver > Rotor and select the options you want to be available using the rotor.

The effect of the rotor setting depends on what you're doing. For example, if you're reading an email, you can use the rotor to switch between hearing text spoken word-by-word or character-by-character when you swipe up or down. If you're browsing a webpage, you can set the rotor to speak all the text (either word-by-word or character-by-character), or to jump from one item to another of a certain type, such as headers or links.

When you use an Apple Wireless Keyboard to control VoiceOver, a speech rotor lets you adjust settings such as volume, speech rate, use of pitch or phonetics, typing echo, and reading of punctuation. See Controlling VoiceOver using an Apple Wireless Keyboard on page 122.

Entering and editing text with VoiceOver

When you enter an editable text field, you can use the onscreen keyboard or an external keyboard connected to iPhone to enter text.

Enter text: Select an editable text field, double-tap to display the insertion point and the onscreen keyboard, then type characters.

- *Standard typing:* Select a key on the keyboard by swiping left or right, then double-tap to enter the character. Or move your finger around the keyboard to select a key and, while continuing to touch the key with one finger, tap the screen with another finger. VoiceOver speaks the key when it's selected, and again when the character is entered.

- *Touch typing:* Touch a key on the keyboard to select it, then lift your finger to enter the character. If you touch the wrong key, slide your finger to the key you want. VoiceOver speaks the character for each key as you touch it, but doesn't enter a character until you lift your finger.
- *Choose standard or touch typing:* With VoiceOver turned on and a key selected on the keyboard, use the rotor to select Typing Mode, then swipe up or down.

Move the insertion point: Swipe up or down to move the insertion point forward or backward in the text. Use the rotor to choose whether you want to move the insertion point by character, by word, or by line.

VoiceOver makes a sound when the insertion point moves, and speaks the character, word, or line that the insertion point moves across. When moving forward by words, the insertion point is placed at the end of each word, before the space or punctuation that follows. When moving backward, the insertion point is placed at the end of the preceding word, before the space or punctuation that follows it.

Move the insertion point past the punctuation at the end of a word or sentence: Use the rotor to switch back to character mode.

When moving the insertion point by line, VoiceOver speaks each line as you move across it. When moving forward, the insertion point is placed at the beginning of the next line (except when you reach the last line of a paragraph, when the insertion point is moved to the end of the line just spoken). When moving backward, the insertion point is placed at the beginning of the line that's spoken.

Change typing feedback: Go to Settings > General > Accessibility > VoiceOver > Typing Feedback.

Use phonetics in typing feedback: Go to Settings > General > Accessibility > VoiceOver > Use Phonetics. Text is read character by character. VoiceOver first speaks the character, then its phonetic equivalent—for example, "f" and then "foxtrot."

Delete a character: Select ⌫, then double-tap or split-tap. You must do this even when touch typing. To delete multiple characters, touch and hold the Delete key, then tap the screen with another finger once for each character you want to delete. VoiceOver speaks the character as it's deleted. If Use Pitch Change is turned on, VoiceOver speaks deleted characters in a lower pitch.

Select text: Set the rotor to Edit, swipe up or down to choose Select or Select All, then double tap. If you chose Select, the word closest to the insertion point is selected when you double-tap. If you chose Select All, all text is selected. Pinch to increase or decrease the selection.

Cut, copy, or paste: Make sure the rotor is set to Edit. With text selected, swipe up or down to choose Cut, Copy, or Paste, then double-tap.

Undo: Shake iPhone, swipe left or right to choose the action to undo, then double-tap.

Enter an accented character: In standard typing mode, select the plain character, then double-tap and hold until you hear a sound indicating alternate characters have appeared. Drag left or right to select and hear the choices. Release your finger to enter the current selection.

Change the keyboard language: Set the rotor to Language, then swipe up or down. Choose "default language" to use the language specified in International settings. The Language rotor appears only if you select more than one language in Settings > General > Accessibility > VoiceOver > Language Rotor.

Making phone calls with VoiceOver

Answer or end a call: Double-tap the screen with two fingers.

When a phone call is established with VoiceOver on, the screen displays the numeric keypad by default, instead of showing call options.

Display call options: Select the Hide Keypad button in the lower-right corner and double-tap.

Display the numeric keypad again: Select the Keypad button near the center of the screen and double-tap.

Using VoiceOver with Safari

When you search the web in Safari with VoiceOver on, the Search Results rotor items lets you hear the list of suggested search phrases.

Search the web: Select the search field, enter your search, then swipe right or left to move down or up the list of suggested search phrases. Then double-tap the screen to search the web using the selected phrase.

Set the rotor options for web browsing: Go to Settings > General > Accessibility > VoiceOver > Rotor. Tap to select or deselect options, or drag ≡ up to reposition an item.

Skip images while navigating: Go to Settings > General > Accessibility > VoiceOver > Navigate Images. You can choose to skip all images or only those without descriptions.

Reduce page clutter for easier reading and navigation: Select the Reader item in the Safari address field (not available for all pages).

Using VoiceOver with Maps

You can use VoiceOver to explore a region, browse points of interest, follow roads, zoom in or out, select a pin, or get information about a location.

Explore the map: Drag your finger around the screen, or swipe left or right to move to another item.

Zoom in or out: Select the map, set the rotor to Zoom, then swipe up or down with one finger.

Pan the map: Swipe with three fingers.

Browse visible points of interest: Set the rotor to Points of Interest, then swipe up or down with one finger.

Follow a road: Hold your finger down on the road, wait until you hear "pause to follow," then move your finger along the road while listening to the guide tone. The pitch increases when you stray from the road.

Select a pin: Touch a pin, or swipe left or right to select the pin.

Get information about a location: With a pin selected, double-tap to display the information flag. Swipe left or right to select the More Info button, then double-tap to display the information page.

Hear location cues as you move about: Turn on tracking with heading to hear street names and points of interest as you approach them.

Editing videos and voice memos with VoiceOver

You can use VoiceOver gestures to trim Camera videos and Voice Memo recordings.

Trim a voice memo: On the Voice Memos screen, select the button to the right of the memo you want to trim, then double-tap. Then select Trim Memo and double-tap. Select the beginning or end of the trim tool. Swipe up to drag right, or swipe down to drag left. VoiceOver announces the amount of time the current position will trim from the recording. To complete the trim, select Trim Voice Memo and double-tap.

Trim a video: While viewing a video in Photos, double-tap the screen to display the video controls, then select the beginning or end of the trim tool. Then swipe up to drag to the right, or swipe down to drag to the left. VoiceOver announces the amount of time the current position will trim from the recording. To complete the trim, select Trim and double-tap.

Controlling VoiceOver using an Apple Wireless Keyboard

You can control VoiceOver using an Apple Wireless Keyboard paired with iPhone. See Apple Wireless Keyboard on page 24.

VoiceOver Help speaks keys or keyboard commands as you type them. You can use VoiceOver Help to learn the keyboard layout and the actions associated with key combinations.

Use VoiceOver keyboard commands to navigate the screen, select items, read screen contents, adjust the rotor, and perform other VoiceOver actions. Most keyboard commands use the Control-Option key combination, abbreviated in the table below as "VO."

VoiceOver keyboard commands

VO = Control-Option

- *Read all, starting from the current position:* VO–A
- *Read from the top:* VO–B
- *Move to the status bar:* VO–M
- *Press the Home button:* VO–H
- *Select the next or previous item:* VO–Right Arrow or VO–Left Arrow
- *Tap an item:* VO–Space bar
- *Open the Item Chooser:* VO–I
- *Double-tap with two fingers:* VO–"-"
- *Select the next or previous item specified by the rotor:* VO–Up Arrow or VO–Down Arrow
- *Adjust the speech rotor:* VO–Command–Left Arrow or VO–Command–Right Arrow
- *Adjust the setting specified by the speech rotor:* VO–Command–Up Arrow or VO–Command–Down Arrow
- *Mute or unmute VoiceOver:* VO–S
- *Switch apps:* Command–Tab or Command–Shift–Tab
- *Turn the screen curtain on or off:* VO–Shift–S
- *Turn on VoiceOver help:* VO–K
- *Return to the previous screen, or turn off VoiceOver help:* Escape

Quick Nav

Turn on Quick Nav to control VoiceOver using the arrow keys.

- *Turn Quick Nav on or off:* Left Arrow–Right Arrow
- *Select the next or previous item:* Right Arrow or Left Arrow

- *Select the next or previous item specified by the rotor:* Up Arrow or Down Arrow
- *Select the first or last item:* Control–Up Arrow or Control–Down Arrow
- *"Tap" an item:* Up Arrow–Down Arrow
- *Scroll up, down, left, or right:* Option–Up Arrow, Option–Down Arrow, Option–Left Arrow, or Option–Right Arrow
- *Adjust the rotor:* Up Arrow–Left Arrow or Up Arrow–Right Arrow

You can also use the number keys on an Apple Wireless Keyboard to dial a phone number in Phone or enter numbers in Calculator.

Single-letter Quick Nav for the web
When you view a webpage with Quick Nav enabled, you can use the following keys on the keyboard to navigate the page quickly. Typing the key moves to the next item of the indicated type. To move to the previous item, hold the Shift key as you type the letter.

- *Heading:* H
- *Link:* L
- *Text field:* R
- *Button:* B
- *Form control:* C
- *Image:* I
- *Table:* T
- *Static text:* S
- *ARIA landmark:* W
- *List:* X
- *Item of the same type:* M
- *Level 1 heading:* 1
- *Level 2 heading:* 2
- *Level 3 heading:* 3
- *Level 4 heading:* 4
- *Level 5 heading:* 5
- *Level 6 heading:* 6

Using a braille display with VoiceOver

You can use a refreshable Bluetooth braille display to read VoiceOver output in braille, and you can use a braille display with input keys and other controls to control iPhone when VoiceOver is turned on. iPhone works with many wireless braille displays. For a list of supported displays, go to www.apple.com/accessibility/iphone/braille-display.html.

Set up a braille display: Turn on the display, then go to Settings > Bluetooth and turn on Bluetooth. Then, go to Settings > General > Accessibility > VoiceOver > Braille and choose the display.

Turn contracted or eight-dot braille on or off: Go to Settings > General > Accessibility > VoiceOver > Braille.

For information about common braille commands for VoiceOver navigation, and for information specific to certain displays, go to support.apple.com/kb/HT4400.

The braille display uses the language that's set for Voice Control. This is normally the language set for iPhone in Settings > International > Language. You can use the VoiceOver language setting to set a different language for VoiceOver and braille displays.

Set the language for VoiceOver: Go to Settings > General > International > Voice Control, then choose the language.

If you change the language for iPhone, you may need to reset the language for VoiceOver and your braille display.

You can set the leftmost or rightmost cell of your braille display to provide system status and other information:

- Announcement History contains an unread message
- The current Announcement History message hasn't been read
- VoiceOver speech is muted
- The iPhone battery is low (less than 20% charge)
- iPhone is in landscape orientation
- The screen display is turned off
- The current line contains additional text to the left
- The current line contains additional text to the right

Set the leftmost or rightmost cell to display status information: Go to Settings > General > Accessibility > VoiceOver > Braille > Status Cell, and tap Left or Right.

See an expanded description of the status cell: On your braille display, press the status cell's router button.

Routing the audio of incoming calls

You can have the audio of incoming calls automatically routed to a headset or speaker phone instead of the iPhone receiver.

Reroute audio for incoming calls: Go to Settings > General > Accessibility > Incoming Calls and choose where you want to hear your calls.

Siri

With Siri, you can do things with your iPhone, such as opening apps, just by asking, and VoiceOver can read Siri responses to you. For information, see Chapter 4, Siri, on page 36.

Triple-click Home

Triple-click Home lets you turn some Accessibility features on or off by pressing the Home button □ quickly three times. You can use Triple-click Home for:

- VoiceOver
- Invert Colors
- Zoom
- AssistiveTouch
- Hearing Aid Control
- Guided Access (Triple-click Home starts Guided Access if it's already turned on. See Guided Access on page 127.)

Set the Triple-click Home function: Go to Settings > General > Accessibility > Triple-click Home. If you select more than one, you're asked which one you want to control whenever you triple-click the Home button.

Slow down the click speed: Go to Settings > General > Accessibility > Home-click Speed.

Zoom

Many apps let you zoom in or out on specific items. For example, you can double-tap or pinch to expand webpage columns in Safari. But, there's also a Zoom accessibility feature that lets you magnify the entire screen of any app you're using. And, you can use Zoom together with VoiceOver.

Turn Zoom on or off: Go to Settings > General > Accessibility > Zoom. Or, use Triple-click Home. See Triple-click Home on page 124.

Zoom in or out: Double-tap the screen with three fingers.

Vary the magnification: With three fingers, tap and drag up or down. The tap-and-drag gesture is similar to a double-tap, except you don't lift your fingers on the second tap—instead, drag your fingers on the screen. Once you start dragging, you can drag with a single finger. iPhone returns to the adjusted magnification when you zoom out and in again using the three-finger double-tap.

Pan around the screen: While zoomed in, drag the screen with three fingers. Once you start dragging, you can drag with a single finger so that you can see more of the screen. Or, hold a single finger near the edge of the display to pan to that side. Move your finger closer to the edge to pan more quickly. When you open a new screen, Zoom goes to the top-middle of the screen.

While using Zoom with an Apple Wireless Keyboard (see Apple Wireless Keyboard on page 24), the screen image follows the insertion point, keeping it in the center of the display.

Large Text

Large Text lets you increase the text size in alerts, and in Calendar, Contacts, Mail, Messages, and Notes.

Set the text size: Go to Settings > General > Accessibility > Large Text.

Invert Colors

Sometimes, inverting the colors on the iPhone screen may make it easier to read. When Invert Colors is turned on, the screen looks like a photographic negative.

Invert the screen's colors: Go to Settings > General > Accessibility > Invert Colors.

Speak Selection

Even with VoiceOver turned off, you can have iPhone read aloud any text you select. iPhone analyzes the text to determine the language, then reads it to you using the appropriate pronunciation.

Turn on Speak Selection: Go to Settings > General > Accessibility > Speak Selection. There you can also:

- Adjust the speaking rate
- Choose to have individual words highlighted as they're read

Have text read to you: Select the text, then tap Speak.

Speak Auto-text

Speak Auto-text speaks the text corrections and suggestions iPhone makes when you type.

Turn Speak Auto-text on or off: Go to Settings > General > Accessibility > Speak Auto-text.

Speak Auto-text also works with VoiceOver and Zoom.

Mono Audio

Mono Audio combines the left and right stereo channels into a mono signal played through both channels. You can adjust the balance of the mono signal for greater volume on the right or left.

Turn Mono Audio on or off and adjust the balance: Go to Settings > General > Accessibility > Mono Audio.

Hearing aids

Made for iPhone hearing aids

If you have a Made for iPhone hearing aid (available for iPhone 4S and later), you can adjust its settings on iPhone to suit your listening needs.

Adjust your hearing aid settings: Go to Settings > General > Accessibility > Hearing Aids, or set Triple-Click Home to open Hearing Aid Control. See Triple-click Home on page 124.

Hearing aid compatibility

The FCC has adopted hearing aid compatibility (HAC) rules for digital wireless phones. These rules require certain phones to be tested and rated under the American National Standard Institute (ANSI) C63.19-2007 hearing aid compatibility standards.

The ANSI standard for hearing aid compatibility contains two types of ratings:

- An "M" rating for reduced radio frequency interference to enable acoustic coupling with hearing aids that are not operating in telecoil mode
- A "T" rating for inductive coupling with hearing aids operating in telecoil mode

These ratings are given on a scale from one to four, where four is the most compatible. A phone is considered hearing aid compatible under FCC rules if it is rated M3 or M4 for acoustic coupling and T3 or T4 for inductive coupling.

For iPhone hearing aid compatibility ratings, go to www.apple.com/support/hac.

Hearing aid compatibility ratings don't guarantee that a particular hearing aid works with a particular phone. Some hearing aids may work well with phones that don't meet particular ratings. To ensure interoperability between a hearing aid and a phone, try using them together before purchase.

This phone has been tested and rated for use with hearing aids for some of the wireless technologies it uses. However, there may be some newer wireless technologies used in this phone that have not been tested yet for use with hearing aids. It is important to try the different features of this phone thoroughly and in different locations, using your hearing aid or cochlear implant, to determine if you hear any interfering noise. Consult your service provider or Apple for information on hearing aid compatibility. If you have questions about return or exchange policies, consult your service provider or phone retailer.

Hearing Aid Mode

iPhone has a Hearing Aid Mode that, when activated, may reduce interference with some hearing aid models. Hearing Aid Mode reduces the transmission power of the cellular radio in the GSM 1900 MHz band and may result in decreased 2G cellular coverage.

Activate Hearing Aid Mode: Go to Settings > General > Accessibility > Hearing Aids.

Assignable ringtones and vibrations

You can assign distinctive ringtones to people in your contacts list for audible caller ID. You can also assign vibration patterns for notifications from specific apps, for phone calls, for FaceTime calls or messages from special contacts, and to alert you of a variety of other events, including new voicemail, new mail, sent mail, Tweet, Facebook Post, and reminders. Choose from existing patterns, or create new ones. See Sounds on page 139.

You can purchase ringtones from the iTunes Store on iPhone. See Chapter 22, iTunes Store, on page 94.

LED Flash for Alerts

If you can't hear the sounds that announce incoming calls and other alerts, you can have iPhone flash its LED (next to the camera lens on the back of the iPhone). This works only when iPhone is locked or asleep. Available for iPhone 4 or later.

Turn on LED Flash for Alerts: Go to Settings > General > Accessibility > LED Flash for Alerts.

Guided Access

Guided Access helps someone using iPhone to stay focused on a particular task. Guided Access limits iPhone to a single app, and lets you control which app features are available. Use Guided Access to:

- Temporarily restrict iPhone to a particular app
- Disable areas of the screen that aren't relevant to a task, or areas where an accidental gesture might cause a distraction
- Disable the iPhone hardware buttons

Use Guided Access: Go to Settings > General > Accessibility > Guided Access, where you can:

- Turn Guided Access on or off
- Set a passcode that controls the use of Guided Access and prevents someone from leaving an active session
- Set whether iPhone can go to sleep during a session

Start a Guided Access session: Open the app you want to run, then triple-click the Home button. Adjust settings for the session, then click Start.

- *Disable app controls and areas of the app screen:* Circle any part of the screen you want to disable. You can use the handles to adjust the area.
- *Ignore all screen touches:* Turn off Touch.
- *Keep iPhone from switching from portrait to landscape or from responding to any other motions:* Turn off Motion.

End a Guided Access session: Triple-click the Home button and enter the Guided Access passcode.

AssistiveTouch

AssistiveTouch helps you use iPhone if you have difficulty touching the screen or pressing the buttons. You can use a compatible adaptive accessory (such as a joystick) together with AssistiveTouch to control iPhone. You can also use AssistiveTouch without an accessory to perform gestures that are difficult for you.

Turn on AssistiveTouch: Go to Settings > General > Accessibility > AssistiveTouch. To set Triple-click Home to turn AssistiveTouch on or off, go to Settings > General > Accessibility > Triple-click Home.

Adjust the tracking speed (with accessory attached): Go to Settings > General > Accessibility > AssistiveTouch > Touch speed.

Show or hide the AssistiveTouch menu: Click the secondary button on your accessory.

Move the menu button: Drag it to any edge of the screen.

Hide the menu button (with accessory attached): Go to Settings > General > Accessibility > AssistiveTouch > Always Show Menu.

Perform a swipe or drag that uses 2, 3, 4, or 5 fingers: Tap the menu button, tap Gestures, and then tap the number of digits needed for the gesture. When the corresponding circles appear on the screen, swipe or drag in the direction required by the gesture. When you finish, tap the menu button.

Perform a pinch gesture: Tap the menu button, tap Favorites, and then tap Pinch. When the pinch circles appear, touch anywhere on the screen to move the pinch circles, then drag the pinch circles in or out to perform a pinch gesture. When you finish, tap the menu button.

Create your own gesture: Tap the menu button, tap Favorites, and then tap an empty gesture placeholder. Or, go to Settings > General > Accessibility > AssistiveTouch > Create New Gesture.

Lock or rotate the screen, adjust iPhone volume, or simulate shaking iPhone: Tap the menu button, then tap Device.

Simulate pressing the Home button: Tap the menu button, then tap Home.

Exit a menu without performing a gesture: Tap anywhere outside the menu.

TTY support

You can use the iPhone TTY Adapter cable (sold separately in many areas) to connect iPhone to a TTY machine. Go to www.apple.com/store (may not be available in all areas) or check with your local Apple retailer.

Connect iPhone to a TTY machine: Go to Settings > Phone and turn TTY on, and then connect iPhone to your TTY machine using the iPhone TTY Adapter.

When TTY on iPhone is turned on, the TTY icon ☎ appears in the status bar at the top of the screen. For information about using a particular TTY machine, see the documentation that came with the machine.

Assignable ringtones

You can assign distinctive ringtones to people in your contacts list for audible caller ID. You can purchase ringtones from the iTunes Store on iPhone. See Chapter 22, iTunes Store, on page 94.

Visual voicemail

The play and pause controls in visual voicemail let you control the playback of messages. Drag the playhead on the scrubber bar to repeat a portion of the message that's hard to understand. See Visual voicemail on page 47.

Widescreen keyboards

Many apps, including Mail, Safari, Messages, Notes, and Contacts, let you rotate iPhone when you're typing, so you can use a larger keyboard.

Large phone keypad

Make phone calls simply by tapping entries in your contacts and favorites lists. When you need to dial a number, iPhone's large numeric keypad makes it easy. See Phone calls on page 43.

Voice Control

Voice Control lets you make phone calls and control Music playback using voice commands. See Making calls on page 43, and Siri and Voice Control on page 62.

Closed captioning

Turn on closed captioning for videos: Go to Settings > Videos > Closed Captioning.

Not all video content includes closed captions.

Accessibility in OS X

Take advantage of the accessibility features in OS X when you use iTunes to sync information and content from your iTunes library to iPhone. In the Finder, choose Help > Help Center, then search for "accessibility."

For more information about iPhone and OS X accessibility features, go to www.apple.com/accessibility.

Settings 33

Settings lets you configure iPhone, set app options, add accounts, and set other preferences. See other chapters for information about settings for the built-in apps. For example, for Safari settings, see Chapter 7, Safari, on page 55.

Airplane mode

Airplane mode disables the wireless features in order to reduce potential interference with aircraft operation and other electrical equipment.

Turn on airplane mode: Go to Settings and turn on airplane mode.

When airplane mode is on, ✈ appears in the status bar at the top of the screen. No phone, Wi-Fi, or Bluetooth signals are emitted from iPhone, and GPS reception is turned off. You won't be able to use apps or features that depend on these signals, such as connecting to the Internet, placing or receiving phone calls or messages, getting visual voicemail, and so on. If allowed by the aircraft operator and applicable laws and regulations, you can use iPhone and apps that don't require these signals.

If Wi-Fi is available and allowed by the aircraft operator and applicable laws and regulations, go to Settings > Wi-Fi to turn it on. You can also turn on Bluetooth in Settings > Bluetooth.

Wi-Fi

Joining Wi-Fi networks

Wi-Fi settings determine whether iPhone uses local Wi-Fi networks to connect to the Internet. When iPhone is joined to a Wi-Fi network, the Wi-Fi icon 📶 in the status bar at the top of the screen shows signal strength. The more bars you see, the stronger the signal. If no Wi-Fi networks are available, or if you've turned Wi-Fi off, then iPhone connects to the Internet via your cellular data network when available.

Once you join a Wi-Fi network, iPhone connects to it whenever the network is in range. If more than one previously used network is in range, iPhone joins the one last used.

You can also use iPhone to set up a new AirPort base station that provides Wi-Fi services to your home or office. See Setting up an AirPort base station on page 131.

Turn Wi-Fi on or off: Go to Settings > Wi-Fi. You can:

- *Set iPhone to ask if you want to join a new network:* Turn "Ask to Join Networks" on or off. If "Ask to Join Networks" is off, you must manually join a network to connect to the Internet when a previously used network isn't available.

- *Forget a network, so iPhone doesn't join it:* Tap ⊙ next to a network you've joined before. Then tap "Forget this Network."
- *Join a closed Wi-Fi network:* In the list of network names, tap Other, then enter the name of the closed network. You must already know the network name, password, and security type to connect to a closed network.
- *Adjust the settings for connecting to a Wi-Fi network:* Tap ⊙ next to a network. You can set an HTTP proxy, define static network settings, turn on BootP, or renew the settings provided by a DHCP server.

Setting up an AirPort base station

An AirPort base station provides a Wi-Fi connection to your home, school, or small business network. You can use iPhone to set up a new AirPort Express, AirPort Extreme, or Time Capsule base station.

Use the AirPort Setup Assistant: Go to Settings > Wi-Fi. Under "Set up an AirPort base station," tap the name of the base station you want to set up. Then follow the onscreen instructions.

If the base station you want to set up isn't listed, make sure that it has power, that you're within range, and that it hasn't already been configured. You can only set up base stations that are new or have been reset. Some older AirPort base stations cannot be set up using an iOS device. For setup instructions, see the documentation that came with the base station.

Manage an AirPort network: If iPhone is connected to an AirPort base station, tap ⊙ next to the network name. If you haven't already downloaded AirPort Utility, the App Store opens so you can get it.

Bluetooth

iPhone can connect wirelessly to Bluetooth devices such as headsets, headphones, and car kits for music listening and hands-free talking. You can also connect the Apple Wireless Keyboard with Bluetooth. See Apple Wireless Keyboard on page 24.

Turn Bluetooth on or off: Go to Settings > Bluetooth.

Connect to a Bluetooth device: Tap the device in the Devices list, then follow the onscreen instructions to connect to it. See the documentation that came with the device for information about Bluetooth pairing.

VPN

Your organization may use a VPN to communicate private information securely over a non-private network. You may need to configure VPN, for example, to access your work email. This setting appears when you have VPN configured on iPhone, allowing you to turn VPN on or off. See Cellular on page 135.

Personal Hotspot

You can use Personal Hotspot (iPhone 4 or later) to share an Internet connection with a computer or other device—such as an iPod touch, iPad, or other iPhone—connected to your iPhone via Wi-Fi. You can also use Personal Hotspot to share an Internet connection with a computer connected to iPhone via Bluetooth or USB. Personal Hotspot works only if iPhone is connected to the Internet over the cellular data network.

Note: This feature may not be available in all areas. Additional fees may apply. Contact your carrier for more information.

Share an Internet connection: Go to Settings > General > Cellular and tap Set Up Personal Hotspot—if it appears—to set up the service with your carrier.

After you turn on Personal Hotspot, other devices can connect in the following ways:

- *Wi-Fi:* On the device, choose your iPhone from the list of available Wi-Fi networks.
- *USB:* Connect your iPhone to your computer using the cable that came with it. In your computer's Network preferences, choose iPhone and configure the network settings.
- *Bluetooth:* On iPhone, go to Settings > Bluetooth and turn on Bluetooth. To pair and connect iPhone with your device, refer to the documentation that came with your computer.

Note: When a device is connected, a blue band appears at the top of the iPhone screen. The Personal Hotspot icon 🔗 appears in the status bar of iOS devices using Personal Hotspot.

Change the Wi-Fi password for iPhone: Go to Settings > Personal Hotspot > Wi-Fi Password, then enter a password of at least 8 characters.

Monitor your cellular data network usage: Go to Settings > General > Usage > Cellular Usage.

Do Not Disturb and Notifications

Push notifications appear in Notification Center and alert you to new information, even when the associated app isn't running. Notifications vary by app, but may include text or sound alerts, and a numbered badge on the app icon on the Home screen.

Turn off all notifications: Go to Settings and turn on Do Not Disturb. If it's on and iPhone is locked, all notifications and calls are silenced, but alarms will still sound. You can set the following options in Settings > Notifications > Do Not Disturb:

- *Automatically turn on Do Not Disturb:* Turn on Scheduled, then set the time when you don't want to be disturbed. iPhone automatically turns on Do Not Disturb during this period each day.
- *Allow some phone calls during Do Not Disturb:* When Do Not Disturb is on, ringing is silenced. To allow calls from some callers to ring, tap Allow Calls From. You can allow calls from your Favorites list, or from other Contacts groups you define. For information about Favorites, see Chapter 25, Contacts, on page 100.
- *Allow persistent callers to ring through:* Turn on Repeated Calls. If the same caller (based on their Caller ID) calls you again within three minutes, iPhone will ring.

Turn an app's notifications on or off: Go to Settings > Notifications. Tap an item in the list, then turn notifications on or off for that item. Apps that have notifications turned off appear in the Not In Notification Center list.

Change how notifications appear: Go to Settings > Notifications. You can:

- *Change the number of notifications:* Choose an item in the In Notification Center list. To set how many notifications of this type appear in Notification Center, tap Show.
- *Change the alert styles:* Choose an item in the In Notification Center list. Choose an alert style, or select None to turn off alerts and banners. Notifications will still appear in Notification Center.
- *Change the order of notifications:* Tap Edit. Drag the notifications into the order you want. To turn off a notification, drag it to the Not In Notification Center list.
- *Display numbered badges on apps with notifications:* Choose an item in the In Notification Center list and turn on Badge App Icon.
- *Hide alerts from an app when iPhone is locked:* Choose the app in the In Notification Center list, then turn off "View in Lock Screen."

Some apps have additional options. For example, Messages lets you specify whether to include message previews in the notification.

Remove Post and Tweet from Notification Center: These sharing options appear only if you have Facebook or Twitter accounts configured. To remove these buttons, go to Settings > Notifications and turn off the Share Widget.

Show government alerts in Notification Center: Choose the alerts you want to see from the Government Alerts list. Government alerts are not available in all areas, vary by carrier and iPhone model, and may not work under all conditions. For example, in the United States, iPhone 4S or later can receive presidential alerts and you can turn AMBER and Emergency Alerts (which includes both Severe and Extreme Imminent Threat alerts) on or off. In Japan, iPhone 4 or later can receive Emergency Earthquake Alerts from the Japan Meteorological Agency.

Carrier

This setting appears on GSM networks when you're outside your carrier's network and other local carrier data networks are available to use for your phone calls, visual voicemail, and cellular network Internet connections. You can make calls only on carriers that have a roaming agreement with your carrier. Additional fees may apply. Roaming charges may be billed to you by the other carrier, through your carrier.

Select a carrier: Go to Settings > Carrier and select the network you want to use.

Once you select a network, iPhone uses only that network. If the network is unavailable, "No service" appears on the iPhone.

General

General settings include network, sharing, security, and other settings. You can also find information about your iPhone, and reset various iPhone settings.

About

Display information about iPhone: Go to Settings > General > About. The items you can view include:

- Available storage space
- Serial number
- iOS version
- Network addresses
- IMEI (International Mobile Equipment Identity)
- ICCID (Integrated Circuit Card Identifier, or Smart Card) for GSM networks
- MEID (Mobile Equipment Identifier) for CDMA networks
- Legal notices, license, and regulatory marks

To copy the serial number and other identifiers, touch and hold the identifier until Copy appears.

Change the device name: Go to Settings > General > About, then tap Name. The device name is used by both iTunes and iCloud.

To help Apple improve products and services, iPhone sends diagnostic and usage data. This data does not personally identify you but may include location information.

View or turn off diagnostic information: Go to Settings > General > About > Diagnostics & Usage.

Restrict or reset Ad Tracking: Go to Settings > General > About > Advertising. Turn on Limit Ad Tracking to prevent apps from accessing your iPhone's advertising identifier. For more information, tap Learn More.

Software Update

Software Update lets you download and install iOS updates from Apple.

Update to the latest iOS version: Go to Settings > General > Software Update.

If a newer version of iOS is available, follow the onscreen instructions to download and install it.

Usage

View usage information: Go to Settings > General > Usage. You can:

- See your cellular usage and reset statistics
- View and delete iCloud backups, turn off backing up the Camera Roll, and buy additional storage
- View each app's storage
- Display battery level as a percentage
- See the elapsed time since iPhone has been charged

Siri

Enable Siri: Go to Settings > General > Siri.

For information about using Siri and changing Siri settings, see Setting options for Siri on page 39.

Cellular

Use Cellular settings to turn cellular data and roaming on or off, to set up Personal Hotspot, and to set cellular data options.

When an app needs to use the Internet, iPhone does the following, in order, until connected:

- Connects over the most recently used available Wi-Fi network.
- Shows a list of Wi-Fi networks in range, and connects using the one you choose.
- Connects over the cellular data network, if available.

If iPhone is connected to the Internet via the cellular data network, the **LTE**, 4G, 3G, E, or ○ icon appears in the status bar.

LTE, 4G and 3G service on GSM cellular networks support simultaneous voice and data communications. For all other cellular connections, you can't use Internet services while you're talking on the phone unless iPhone also has a Wi-Fi connection to the Internet. Depending on your network connection, you may not be able to receive calls while iPhone transfers data over the cellular network—when downloading a webpage, for example.

GSM networks: On an EDGE or GPRS connection, incoming calls may go directly to voicemail during data transfers. For incoming calls that you answer, data transfers are paused.

CDMA networks: On EV-DO connections, data transfers are paused when you answer incoming calls. On 1xRTT connections, incoming calls may go directly to voicemail during data transfers. For incoming calls that you answer, data transfers are paused.

Data transfer resumes when you end the call.

If Cellular Data is off, all data services use only Wi-Fi—including email, web browsing, push notifications, and other services. If Cellular Data is on, carrier charges may apply. For example, using certain features and services that transfer data, such as Siri and Messages, could result in charges to your data plan.

Turn Cellular Data on or off: Go to Settings > General > Cellular. The following options may also be available:

- *Turn Voice Roaming on or off (CDMA):* Turn Voice Roaming off to avoid charges from using other carrier's networks. When your carrier's network isn't available, iPhone won't have cellular (data or voice) service.
- *Turn Data Roaming on or off:* Data Roaming permits Internet access over a cellular data network when you're in an area not covered by your carrier's network. When you're traveling, you can turn off Data Roaming to avoid roaming charges. See Carrier on page 133.
- *Enable or disable 3G:* Using 3G loads Internet data faster in some cases, but may decrease battery performance. If you're making a lot of phone calls, you may want to turn 3G off to extend battery life. This option is not available in all areas.

Set up Personal Hotspot: Go to Settings > General > Cellular > Set Up Personal Hotspot. Personal Hotspot shares iPhone's Internet connection with your computer and other iOS devices. See Personal Hotspot on page 132.

Set when cellular data is used: Go to Settings > General > Cellular, then turn cellular data on or off for iCloud Documents, iTunes, FaceTime, Passbook Updates, or Reading List. If a setting is off, iPhone will use only Wi-Fi for that service. The iTunes settings includes both iTunes Match and automatic downloads from the iTunes Store and the App Store.

VPN

VPNs used within organizations allow you to communicate private information securely over a non-private network. You may need to configure VPN, for example, to access your work email. Ask the network's administrator for the settings necessary to configure VPN for your network. After one or more VPN settings are defined you can:

- *Turn VPN on or off:* Go to Settings > VPN.
- *Switch between VPNs:* Go to Settings > General > VPN, then choose a configuration.

See also Appendix A, iPhone in Business, on page 141.

iTunes Wi-Fi Sync

You can sync iPhone with iTunes on a computer that's connected to the same Wi-Fi network.

Enable iTunes Wi-Fi Sync: To set up Wi-Fi syncing for the first time, connect iPhone to the computer that you want to sync with. For instructions see Syncing with iTunes on page 16.

After you configure Wi-Fi Sync, iPhone automatically syncs with iTunes once a day, when:

- iPhone is connected to a power source,
- iPhone and your computer are both connected to the same Wi-Fi network, and
- iTunes on your computer is running.

Spotlight Search

The Spotlight Search setting lets you specify the content areas searched by Search, and rearrange the order of the results.

Set which content areas are searched by Search: Go to Settings > General > Spotlight Search, then select the items to search. You can also change the order of the result categories.

Auto-Lock

Locking iPhone turns off the display in order to save the battery and prevent unintended operation of iPhone. You can still receive calls and text messages, and you can adjust the volume and use the mic button on your headset while listening to music or on a call.

Set the amount of time before iPhone locks: Go to Settings > General > Auto-Lock, then choose a time.

Passcode Lock

By default, iPhone doesn't require you to enter a passcode to unlock it.

Set a passcode: Go to Settings > General > Passcode Lock and set a 4-digit passcode. To increase security, turn off Simple Passcode and use a longer passcode.

If you forget your passcode, you must restore the iPhone software. See Updating and restoring iPhone software on page 152.

Allow access when iPhone is locked: Go to Settings > General > Passcode Lock. You can use the following without unlocking iPhone:

- Siri (See Setting options for Siri on page 39.)
- Voice Dial (This setting is available only when Siri is turned off.)
- Reply with Message (See Receiving calls on page 44.)
- Passbook (See Chapter 16, Passbook, on page 84.)

Erase data after ten failed passcode attempts: Go to Settings > General > Passcode Lock and tap Erase Data. After ten failed passcode attempts, all settings are reset, and all your information and media are erased by removing the encryption key to the data (which is encrypted using 256-bit AES encryption).

Restrictions

You can set restrictions for some apps and for purchased content. For example, parents can restrict explicit music from being seen on playlists, or prevent the installation of apps.

Turn on restrictions: Go to Settings > General > Restrictions, then tap Enable Restrictions. You'll be asked to define a restrictions passcode that's necessary in order to change the settings you make. This is distinct from the passcode for unlocking iPhone.

Important: If you forget your restrictions passcode, you must restore the iPhone software. See Updating and restoring iPhone software on page 152.

You can set restrictions for the following apps:

- Safari
- Camera (and apps that use the camera)
- FaceTime
- iTunes Store
- iBookstore
- Siri (including voice command and dictation)

You can also restrict the following:

- *Installing Apps:* The App Store is disabled and its icon is removed from the Home screen. You cannot install apps on iPhone.
- *Deleting Apps:* You cannot delete apps from iPhone. ⊗ doesn't appear on app icons when you're customizing the Home screen.
- *Explicit Language:* Siri attempts to replace explicit words you speak by replacing them with asterisks and beep sounds
- *Privacy:* The privacy settings for Location Services, Contacts, Calendars, Reminders, Photos, Bluetooth Sharing, Twitter, and Facebook can each be locked.
- *Accounts:* The current Mail, Contacts, Calendar settings are locked. You cannot add, modify, or delete accounts. You also cannot modify iCloud settings.
- *Find My Friends:* The current Find My Friends settings are locked. This option is available when the Find My Friends app is installed.
- *Volume Limit:* The current sound volume limit setting is locked.
- *In-App Purchases:* When In-App Purchases is turned off, you can't purchase additional content or functionality for apps you download from the App Store.
- *Require Passwords:* Requires you to enter your Apple ID for in-app purchases after the time period you specify.
- *Content Restrictions:* Tap Ratings For, then select a country from the list. Then set restrictions for music, podcasts, movies, TV shows, and apps. Content that doesn't meet the rating you select won't appear on iPhone.
- *Multiplayer Games:* When Multiplayer Games is off, you can't request a match, send or receive invitations to play games, or add friends in Game Center.
- *Adding Friends:* When Adding Friends is off, you can't make or receive friend requests in Game Center. If Multiplayer Games is turned on, you can continue to play with existing friends.

Date & Time

These settings affect the time shown in the status bar at the top of the screen, and in world clocks and calendars.

Set whether iPhone shows 24-hour time or 12-hour time: Go to Settings > General > Date & Time, then turn 24-Hour Time on or off. (24-Hour Time may not be available in all areas.)

Set whether iPhone updates the date and time automatically: Go to Settings > General > Date & Time, then turn Set Automatically on or off. If you set iPhone to update the time automatically, it gets the correct time over the cellular network and updates it for the time zone you're in. Some carriers don't support network time, so in some areas iPhone may not be able to automatically determine the local time.

Set the date and time manually: Go to Settings > General > Date & Time, then turn Set Automatically off. Tap Time Zone to set your time zone. Tap the Date & Time button, then tap Set Date & Time.

Keyboard

You can turn on keyboards for writing in different languages, and you can turn typing features, such as spell-checking, on or off. For information about the keyboard, see Typing on page 22.

For information about international keyboards, see Appendix B, International Keyboards, on page 143.

International

Go to Settings > General > International to set the following:

- The language for iPhone
- The calendar format
- The language for Voice Control
- The keyboards you use
- The date, time, and telephone number formats

Accessibility

Go to Settings > General > Accessibility and turn on the features you want. See Chapter 32, Accessibility, on page 115.

Profiles

This setting appears if you install one or more profiles on iPhone. Tap Profiles to see information about the profiles you've installed. For more information see Using configuration profiles on page 141.

Reset

You can reset the word dictionary, network settings, home screen layout, and location warnings. You can also erase all of your content and settings.

Reset iPhone: Go to Settings > General > Reset, then choose an option:

- *Reset all settings:* All your preferences and settings are reset.
- *Erase all content and settings:* Your information, and settings are removed. iPhone cannot be used until it's set up again.

- *Reset network settings:* When you reset network settings, your list of previously used networks and VPN settings not installed by a configuration profile are removed. Wi-Fi is turned off and then back on, disconnecting you from any network you're on. The Wi-Fi and "Ask to Join Networks" settings remain turned on. To remove VPN settings installed by a configuration profile, go to Settings > General > Profile, then select the profile and tap Remove. This also removes other settings or accounts provided by the profile.
- *Reset the keyboard dictionary:* You add words to the keyboard dictionary by rejecting words iPhone suggests as you type. Resetting the keyboard dictionary erases all words you've added.
- *Reset the Home screen layout:* Returns the built-in apps to their original layout on the Home screen.
- *Reset location and privacy:* Resets the location services and privacy settings to their factory defaults.

Sounds

You can set iPhone to play a sound whenever you get a new message, email, call, Tweet, Facebook post, voicemail, or reminder. You can also set sounds for appointments, sending an email, pressing keys, and locking iPhone.

For information about silencing iPhone see Ring/Silent switch on page 10.

Change sound settings: Go to Settings > Sounds. Available options include:

- Set whether iPhone vibrates when get a call.
- Set whether iPhone vibrates when you turn on silent mode.
- Adjust the ringer and alerts volume.
- Prevent the side buttons from changing the ringer volume.
- Set the ringtone. To set a ringtone for a person, go to their card in Contacts.
- Set alert and other tones.
- Turn on keyboard clicks and a sound for when when iPhone locks.

Set vibration patterns: Go to Settings > Sounds and choose an item from the Sounds and Vibration Patterns list. Tap Vibration to select a pattern.

- *Define a custom vibration pattern:* Tap an item in the Sounds and Vibrations list, then tap Vibration. Tap Create New Vibration then define the pattern by touching and tapping the screen.

Brightness & Wallpaper

Screen brightness affects battery life. Dim the screen to extend the time before you need to recharge iPhone, or use Auto-Brightness.

Adjust the screen brightness: Go to Settings > Brightness & Wallpaper and drag the slider. If Auto-Brightness is on, iPhone adjusts the screen brightness for current light conditions using the built-in ambient light sensor.

Wallpaper settings let you set an image or photo as wallpaper for the Lock screen or Home screen. See Changing the wallpaper on page 21.

Privacy

Privacy settings let you see and control which apps and system services have access to Location Services, and to contacts, calendars, reminders, and photos.

Location Services lets location-based apps such as Reminders, Maps, and Camera gather and use data indicating your location. Your approximate location is determined using available information from cellular network data, local Wi-Fi networks (if you have Wi-Fi turned on), and GPS (may not be available in all areas). The location data collected by Apple isn't collected in a form that personally identifies you. When an app is using Location Services, ↗ appears in the menu bar.

Turn Location Services on or off: Go to Settings > Privacy > Location Services. You can turn it off for some or for all apps and services. If you turn off Location Services, you're prompted to turn it on again the next time an app or service tries to use it.

Turn Location Services off for system services: Several system services, such as compass calibration and location-based iAds, use Location Services. To see their status, turn them on or off, or show ↗ in the menu bar when these services use your location, go to Settings > Privacy > Location Services > System Services.

Turn off access to private information: Go to Settings > Privacy. You can see which apps have requested and been granted access to the following information:

- Contacts
- Calendar
- Reminders
- Photos
- Bluetooth Sharing
- Twitter
- Facebook

You can turn off each app's access to each category of information. Review the terms and privacy policy for each third-party app to understand how it uses the data it's requesting.

iPhone in Business

A Appendix

With support for secure access to corporate networks, directories, and Microsoft Exchange, iPhone is ready to go to work. For detailed information about using iPhone in business, go to www.apple.com/iphone/business.

Using configuration profiles

If you're in an enterprise environment, you may be able to set up accounts and other items on iPhone by installing a configuration profile. Configuration profiles let your administrator set up your iPhone to use the information systems at your company, school, or organization. For example, a configuration profile might set up your iPhone to access the Microsoft Exchange servers at work, so iPhone can access your Exchange email, calendars, and contacts, and it may turn on Passcode Lock to help keep the information secure.

Your administrator may distribute configuration profiles by email, by putting them on a secure webpage, or by installing them directly on iPhone for you. Your administrator may have you install a profile that ties your iPhone to a mobile device management server, which allows your administrator to configure your settings remotely.

Install configuration profiles: On iPhone, open the email message or download the configuration profiles from the website your administrator provides. When you open a configuration profile, installation begins.

Important: You may be asked whether a configuration profile is trusted. If in doubt, ask your administrator before installing the configuration profile.

You can't change the settings defined by a configuration profile. If you want to change settings, you must first remove the configuration profile, or install a new configuration profile with the new settings.

Remove a configuration profile: Go to Settings > General > Profile, then select the configuration profile and tap Remove.

Removing a configuration profile deletes the settings and all other information installed by the profile.

Setting up Microsoft Exchange accounts

Microsoft Exchange provides email, contact, tasks, and calendar information that you can automatically sync wirelessly to iPhone. You can set up an Exchange account directly on iPhone.

Set up an Exchange account on iPhone: Go to Settings > Mail, Contacts, Calendars. Tap Add Account, then tap Microsoft Exchange. Ask your service provider or administrator what settings you should use.

VPN access

VPN (virtual private network) provides secure access over the Internet to private networks, such as the network at your company or school. Use Network settings on iPhone to configure and turn on VPN. Ask your administrator what settings you should use.

VPN can also be set up automatically by a configuration profile. When VPN is set up by a configuration profile, iPhone may turn VPN on automatically whenever it's needed. For more information, contact your administrator.

LDAP and CardDAV accounts

When you set up an LDAP account, you can view and search for contacts on your organization's LDAP server. The server appears as a new group in Contacts. Because LDAP contacts aren't downloaded to iPhone, you must have an Internet connection to view them. Check with your administrator for account settings and other requirements (such as VPN).

When you set up a CardDAV account, your account contacts are synced with iPhone over the air. You may also be able to search for contacts on your organization's CardDAV server.

Set up an LDAP or CardDAV account: Go to Settings > Mail, Contacts, Calendars, then tap Add Account. Tap Other. Ask your service provider or administrator what settings you should use.

International Keyboards

B Appendix

Using international keyboards

International keyboards let you type text in many different languages, including Asian languages and languages written from right to left. For a list of supported keyboards, go to www.apple.com/iphone/specs.html.

Manage keyboards: Go to Settings > General > International > Keyboards.

- *Add a keyboard:* Tap Add New Keyboard, then choose a keyboard from the list. Repeat to add more keyboards.
- *Remove a keyboard:* Tap Edit, tap ⊖ next to the keyboard you want to remove, then tap Delete.
- *Edit your keyboard list:* Tap Edit, then drag ≡ next to a keyboard to a new place in the list.

To enter text in a different language, switch keyboards.

Switch keyboards while typing: Touch and hold the Globe key 🌐 to show all your enabled keyboards. To choose a keyboard, slide your finger to the name of the keyboard, then release. The Globe key 🌐 appears only if you enable more than one keyboard.

You can also just tap 🌐. When you tap 🌐, the name of the newly activated keyboard appears briefly. Continue tapping to access other enabled keyboards.

Many keyboards provide letters, numbers, and symbols that aren't visible on the keyboard.

Enter accented letters or other characters: Touch and hold the related letter, number, or symbol, then slide to choose a variant. For example:

- *On a Thai keyboard:* Choose native numbers by touching and holding the related Arabic number.
- *On a Chinese, Japanese, or Arabic keyboard:* Suggested characters or candidates appear at the top of the keyboard. Tap a candidate to enter it, or flick left to see more candidates.

Use the extended candidate list: Tap the up arrow at the right to view the full candidate list.

- *Scroll the list:* Flick up or down.
- *Return to the short list:* Tap the down arrow.

When using certain Chinese or Japanese keyboards, you can create a shortcut for word and input pairs. The shortcut is added to your personal dictionary. When you type a shortcut while using a supported keyboard, the paired word or input is substituted for the shortcut.

Turn shortcuts on or off: Go to Settings > General > Keyboard > Shortcuts. Shortcuts are available for:

- Simplified Chinese: Pinyin
- Traditional Chinese: Pinyin and Zhuyin
- Japanese: Romaji and 50 Key

Special input methods

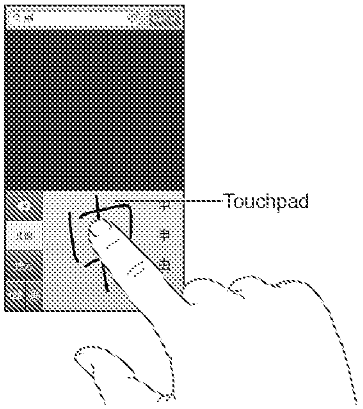
You can use keyboards to enter some languages in different ways. A few examples are Chinese Cangjie and Wubihua, Japanese Kana, and Facemarks. You can also use your finger or a stylus to write Chinese characters on the screen.

Build Chinese characters from the component Cangjie keys: As you type, suggested characters appear. Tap a character to choose it, or continue typing up to five components to see more options.

Build Chinese Wubihua (stroke) characters: Use the keypad to build Chinese characters using up to five strokes, in the correct writing sequence: horizontal, vertical, left falling, right falling, and hook. For example, the Chinese character 圈 (circle) should begin with the vertical stroke 丨.

- As you type, suggested Chinese characters appear (the most commonly used characters appear first). Tap a character to choose it.
- If you're not sure of the correct stroke, enter an asterisk (*). To see more character options, type another stroke, or scroll through the character list.
- Tap the match key (匹配) to show only characters that match exactly what you typed.

Write Chinese characters: Write Chinese characters directly on the screen with your finger when Simplified or Traditional Chinese handwriting formats are turned on. As you write character strokes, iPhone recognizes them and shows matching characters in a list, with the closest match at the top. When you choose a character, its likely follow-on characters appear in the list as additional choices



Some complex characters, such as 鱲 (part of the name for the Hong Kong International Airport), 𨋢 (elevator), and 㗎 (particle used in Cantonese), can be typed by writing two or more component characters in sequence. Tap the character to replace the characters you typed. Roman characters are also recognized.

Type Japanese kana: Use the Kana keypad to select syllables. For more syllable options, tap the arrow key and select another syllable or word from the window.

Type Japanese romaji: Use the Romaji keyboard to type syllables. Alternative choices appear along the top of the keyboard, tap one to type it. For more syllable options, tap the arrow key and select another syllable or word from the window.

Type facemarks or emoticons: Use the Japanese Kana keyboard and tap the ^_^ key. Or you can:

- *Use the Japanese Romaji keyboard (QWERTY-Japanese layout):* Tap the Number key 123, then tap the ^_^ key.

- *Use the Chinese (Simplified or Traditional) Pinyin or (Traditional) Zhuyin keyboard:* Tap the Symbols key ▦, then tap the ^_^ key.

Safety, Handling, & Support

C Appendix

Important safety information

WARNING: Failure to follow these safety instructions could result in fire, electric shock, or other injuries, or damage to iPhone or other property. Read all the safety information below before using iPhone.

Handling Handle iPhone with care. It is made of metal, glass, and plastic and has sensitive electronic components inside. iPhone can be damaged if dropped, burned, punctured, or crushed, or if it comes in contact with liquid. Don't use a damaged iPhone, such as one with a cracked screen, as it may cause injury. If you're concerned about scratching, consider using a case.

Repairing Don't open iPhone and don't attempt to repair iPhone by yourself. Disassembling iPhone may cause injury to you or damage to iPhone. If iPhone is damaged, malfunctions, or comes in contact with liquid, contact Apple or an Apple Authorized Service Provider. You can find more information about getting service at www.apple.com/support/iphone/service/faq.

Battery Don't attempt to replace the iPhone battery yourself—you may damage the battery, which could cause overheating and injury. The lithium-ion battery in iPhone should be replaced only by Apple or an Apple Authorized Service Provider, and must be recycled or disposed of separately from household waste. Don't incinerate the battery. For information about battery recycling and replacement, go to www.apple.com/batteries.

Distraction Using iPhone in some circumstances can distract you and may cause a dangerous situation. Observe rules that prohibit or restrict the use of mobile phones or headphones (for example, avoid texting while driving a car or using headphones while riding a bicycle).

Navigation Maps, directions, Flyover, and location-based apps depend on data services. These data services are subject to change and may not be available in all areas, resulting in maps, directions, Flyover, or location-based information that may be unavailable, inaccurate, or incomplete. Compare the information provided on iPhone to your surroundings, and defer to posted signs to resolve any discrepancies. Some Maps features require Location Services. See Privacy on page 140. Use common sense when navigating.

Charging Charge iPhone with the included USB cable and power adapter or other third-party "Made for iPhone" cables and power adapters that are compatible with USB 2.0 or power adapters compliant with one or more of the following standards EN 301489-34, IEC 62684, YD/T 1591-2009, CNS 15285, ITU L.1000, or another applicable mobile phone power adapter interoperability standard. An iPhone Micro USB Adapter (available separately in some areas) or other adapter may be needed to connect iPhone to some compatible power adapters. Using damaged cables or chargers, or charging when moisture is present, can cause electric shock. When you use the Apple USB Power Adapter to charge iPhone, make sure that the AC plug or AC power cord is fully inserted into the adapter before you plug it into a power outlet. Power adapters may become warm during normal use, and prolonged contact may cause injury. Always allow adequate ventilation around power adapters when using them.

Note: Only micro USB power adapters in certain regions that comply with applicable mobile phone power adapter interoperability standards are compatible. Please contact the power adapter manufacturer to find out if your micro USB power adapter complies with these standards.

Hearing loss Listening to sound at high volumes may damage your hearing. Background noise, as well as continued exposure to high volume levels, can make sounds seem quieter than they actually are. Turn on the audio and check the volume before inserting anything in your ear. For more information about hearing loss, see www.apple.com/sound. For information about how to set a maximum volume limit on iPhone, see Music settings on page 63.

WARNING: To prevent possible hearing damage, do not listen at high volume levels for long periods.

Apple headsets The headsets sold with iPhone 4S or later in China (identifiable by dark insulating rings on the plug) are designed to comply with Chinese standards and are compatible with iPhone 4S or later, iPad 2 or later, and iPod touch 5th generation. Use only compatible headsets with your device.

Radio signals iPhone uses radio signals to connect to wireless networks. For information about the amount of power used to transmit these signals, and about steps you can take to minimize exposure, see Settings > General > About > Legal > RF Exposure.

Radio frequency interference Observe signs and notices that prohibit or restrict the use of mobile phones (for example, in healthcare facilities or blasting areas). Although iPhone is designed, tested, and manufactured to comply with regulations governing radio frequency emissions, such emissions from iPhone can negatively affect the operation of other electronic equipment, causing them to malfunction. Turn off iPhone or use Airplane Mode to turn off the iPhone wireless transmitters when use is prohibited, such as while traveling in aircraft, or when asked to do so by authorities.

Medical devices iPhone contains radios that emit electromagnetic fields. These electromagnetic fields may interfere with pacemakers or other medical devices. If you wear a pacemaker, maintain at least 6 inches (approximately 15 cm) of separation between your pacemaker and iPhone. If you suspect iPhone is interfering with your pacemaker or any other medical device, stop using iPhone and consult your physician for information specific to your medical device. iPhone has magnets near the bottom, and the included headphones also have magnets in the earbuds, which may interfere with pacemakers, defibrillators or other medical devices. Maintain at least 6 inches (approximately 15 cm) of separation between your pacemaker or defibrillator and iPhone or the earbuds.

Medical conditions If you have any other medical condition that you believe could be affected by iPhone (for example, seizures, blackouts, eyestrain, or headaches), consult with your physician prior to using iPhone.

Explosive atmospheres Do not charge or use iPhone in any area with a potentially explosive atmosphere, such as at a fueling area, or in areas where the air contains chemicals or particles (such as grain, dust, or metal powders). Obey all signs and instructions.

Repetitive motion When you perform repetitive activities such as typing or playing games on iPhone, you may experience occasional discomfort in your hands, arms, wrists, shoulders, neck, or other parts of your body. If you experience discomfort, stop using iPhone and consult a physician.

High-consequence activities This device is not intended for use where the failure of the device could lead to death, personal injury, or severe environmental damage.

Choking hazard Some iPhone accessories may present a choking hazard to small children. Keep these accessories away from small children.

Important handling information

Cleaning Clean iPhone immediately if it comes in contact with anything that may cause stains—such as dirt, ink, makeup, or lotions. To clean:

- Disconnect all cables and turn iPhone off (press and hold the Sleep/Wake button, then slide the onscreen slider).
- Use a soft, lint-free cloth.
- Avoid getting moisture in openings.
- Don't use cleaning products or compressed air.

The front or back cover of iPhone may be made of glass with a fingerprint-resistant oleophobic (oil repellant) coating. This coating wears over time with normal usage. Cleaning products and abrasive materials will further diminish the coating, and may scratch the glass. Abrasive media may also scratch iPhone.

Using connectors, ports, and buttons Never force a connector into a port or apply excessive pressure to a button, because this may cause damage that is not covered under the warranty. If the connector and port don't join with reasonable ease, they probably don't match. Check for obstructions and make sure that the connector matches the port and that you have positioned the connector correctly in relation to the port.

Lightning Discoloration of the Lightning plug after regular use is normal. Dirt, debris, and exposure to liquids may cause discoloration. To remove the discoloration or if the cable becomes warm during use or won't charge or sync your iPhone, disconnect the Lightning cable from your computer or power adapter and clean it with a soft, dry, lint-free cloth. Do not use liquids or cleaning products when cleaning the Lightning connector.

Operating temperature iPhone is designed to work in ambient temperatures between 32° and 95° F (0° and 35° C) and stored in temperatures between -4° and 113° F (-20° and 45° C). iPhone can be damaged and battery life shortened if stored or operated outside of these temperature ranges. Avoid exposing iPhone to dramatic changes in temperature or humidity. When you're using iPhone or charging the battery, it is normal for iPhone to get warm.

If the interior temperature of iPhone exceeds normal operating temperatures (for example, in a hot car or in direct sunlight for extended periods of time), you may experience the following as it attempts to regulate its temperature:

- iPhone stops charging.
- The screen dims.
- A temperature warning screen appears.
- Some apps may close.

Important: You may not be able to use iPhone while the temperature warning screen is displayed. If iPhone can't regulate its internal temperature, it goes into deep sleep mode until it cools. Move iPhone to a cooler location out of direct sunlight and wait a few minutes before trying to use iPhone again.

For more information, go to support.apple.com/kb/HT2101.

iPhone Support site

Comprehensive support information is available online at www.apple.com/support/iphone. To contact Apple for personalized support (not available in all areas), see www.apple.com/support/contact.

Restarting or resetting iPhone

If something isn't working right, try restarting iPhone, forcing an app to close, or resetting iPhone.

Restart iPhone: Hold down the Sleep/Wake button until the red slider appears. Slide your finger across the slider to turn off iPhone. To turn iPhone back on, hold down the Sleep/Wake button until the Apple logo appears.

Force an app to close: Hold down the Sleep/Wake button for a few seconds until a red slider appears, then hold down the Home button □ until the app closes.

You can also remove an app from the recents list to force it to close. See Opening and switching between apps on page 17.

If you can't turn off iPhone or if the problem continues, you may need to reset iPhone. A reset should be done only if turning iPhone off and on doesn't resolve the problem.

Reset iPhone: Hold down the Sleep/Wake button and the Home button □ at the same time for at least ten seconds, until the Apple logo appears.

"Wrong Passcode" or "iPhone is disabled" appears

If you forget your passcode or iPhone displays an alert that it is disabled, see "iOS: Wrong passcode results in red disabled screen" at support.apple.com/kb/HT1212.

"This accessory is not supported by iPhone" appears

The accessory you attached may not work with iPhone. Make sure the USB cable and connectors are free of debris, and refer to the documentation that came with the accessory.

Can't view email attachments

If iPhone can't view email attachments, try the following:

- *View an attached file:* Tap the attachment to open it in Quick Look. You may need to wait while it downloads before viewing.
- *Save an attached photo or video:* Tap the attachment to open it in Quick Look. You may need to wait while it downloads before viewing.

Quick Look supports the following document types:

- *.doc, .docx*—Microsoft Word
- *.htm, .html*—webpage
- *.key*—Keynote
- *.numbers*—Numbers
- *.pages*—Pages
- *.pdf*—Preview, Adobe Acrobat
- *.ppt, .pptx*—Microsoft PowerPoint
- *.rtf*—Rich Text Format
- *.txt*—text
- *.vcf*—contact information
- *.xls, .xlsx*—Microsoft Excel

For additional troubleshooting information, go to www.apple.com/support/iphone.

Backing up iPhone

You can use iCloud or iTunes to automatically back up iPhone. If you choose to back up using iCloud, you can't also use iTunes to automatically back up to your computer, but you can use iTunes to manually back up to your computer.

Backing up with iCloud

iCloud backs up to iPhone daily over Wi-Fi, when it's connected to a power source and is locked. The date and time of the last backup is listed at the bottom of the Storage & Backup screen. iCloud backs up your:

- Purchased music, TV shows, apps, and books
- Photos and videos in your Camera Roll
- iPhone settings
- App data
- Home screen and app organization
- Messages (iMessage, SMS, and MMS)
- Ringtones

Note: Purchased music is not backed up in all areas and TV shows are not available in all areas.

If you didn't enable iCloud backup when you first set up iPhone, you can turn it on in iCloud settings.

Turn on iCloud backups: Go to Settings > iCloud, then log in with your Apple ID and password, if required. Go to Storage & Backup, then turn on iCloud Backup.

Back up immediately: Go to Settings > iCloud > Storage & Backup, then tap Back Up Now.

Manage your backups: Go to Settings > iCloud > Storage & Backup, then tap Manage Storage. Tap the name of your iPhone.

Turn Camera Roll backup on or off: Go to Settings > iCloud > Storage & Backup, then tap Manage Storage. Tap the name of your iPhone, then turn Camera Roll backup on or off.

View the devices being backed up: Go to Settings > iCloud > Storage & Backup > Manage Storage.

Stop iCloud backups: Go to Settings > iCloud > Storage & Backup > Backup, then turn off iCloud Backup.

Music that isn't purchased in iTunes isn't backed up in iCloud. You have to use iTunes to back up and restore that content. See Syncing with iTunes on page 16.

Important: Backups for music or TV show purchases are not available in all areas. Previous purchases may be unavailable if they are no longer in the iTunes Store, App Store, or iBookstore.

Purchased content, as well as Photo Stream content, doesn't count against your 5 GB of free iCloud storage.

Backing up with iTunes

iTunes creates a backup of photos in your Camera Roll or Saved Photos album, and backups of text messages, notes, call history, your Favorites list, sound settings, and more. Media files, such as songs, and some photos, aren't backed up, but can be restored by syncing with iTunes.

When you connect iPhone to the computer you normally sync with, iTunes creates a backup each time you:

- *Sync with iTunes:* iTunes syncs iPhone each time you connect iPhone to your computer. iTunes won't automatically back up an iPhone that isn't configured to sync with that computer. See Syncing with iTunes on page 16.
- *Update or restore iPhone:* iTunes always backs up iPhone before updating and restoring.

iTunes can also encrypt iPhone backups to secure your data.

Encrypt iPhone backups: Select "Encrypt iPhone backup" in the iTunes Summary pane.

Restore iPhone files and settings: Connect iPhone to the computer you normally sync with, select iPhone in the iTunes window, and click Restore in the Summary pane.

For more information about backups, go to support.apple.com/kb/HT1766.

Removing an iTunes backup

You can remove an iPhone backup from the list of backups in iTunes. You may want to do this, for example, if a backup was created on someone else's computer.

Remove a backup:

1 In iTunes, open iTunes Preferences.

- *Mac:* Choose iTunes > Preferences.
- *Windows:* Choose Edit > Preferences.

2 Click Devices (iPhone doesn't need to be connected).

3 Select the backup you want to remove, then click Delete Backup.

4 Click Delete, to confirm you wish to remove the selected backup, then click OK.

Updating and restoring iPhone software

You can update iPhone software in Settings, or by using iTunes. You can also erase or restore iPhone, and then use iCloud or iTunes to restore from a backup.

Deleted data is no longer accessible through the iPhone user interface, but it isn't erased from iPhone. For information about erasing all content and settings, see Reset on page 138.

Updating iPhone

You can update software in iPhone Settings or by using iTunes.

Update wirelessly on iPhone: Go to Settings > General > Software Update. iPhone checks for available software updates.

Update software in iTunes: iTunes checks for available software updates each time you sync iPhone using iTunes. See Syncing with iTunes on page 16.

For more information about updating iPhone software, go to support.apple.com/kb/HT4623.

Restoring iPhone

You can use iCloud or iTunes to restore iPhone from a backup.

Restore from an iCloud backup: Reset iPhone to erase all settings and information. Sign in to iCloud and choose Restore from a Backup in the Setup Assistant. See Reset on page 138.

Restore from an iTunes backup: Connect iPhone to the computer you normally sync with, select iPhone in the iTunes window, and click Restore in the Summary pane.

When the iPhone software is restored, you can either set it up as a new iPhone, or restore your music, videos, app data, and other content from a backup.

For more information about restoring iPhone software, go to support.apple.com/kb/HT1414.

Learning more, service, and support

This table describes where to get more iPhone-related safety, software, and service information.

To learn about	Do this
Using iPhone safely	See Important safety information on page 146.
iPhone service and support, tips, forums, and Apple software downloads	Go to www.apple.com/support/iphone.
Service and support from your carrier	Contact your carrier or go to your carrier's website.
The latest information about iPhone	Go to www.apple.com/iphone.
Managing your Apple ID account	Go to appleid.apple.com.
Using iCloud	Go to www.apple.com/support/icloud.
Using iTunes	Open iTunes and choose Help > iTunes Help. For an online iTunes tutorial (may not be available in all areas), go to www.apple.com/support/itunes.
Using other Apple iOS apps	Go to www.apple.com/support/ios.
Finding your iPhone serial number, IMEI, ICCID, or MEID	You can find your iPhone serial number, International Mobile Equipment Identity (IMEI), ICCD, or Mobile Equipment Identifier (MEID) on the iPhone packaging. Or, on iPhone, choose Settings > General > About. For more information, go to support.apple.com/kb/ht4061.

To learn about	Do this
Obtaining warranty service	First follow the advice in this guide. Then go to www.apple.com/support/iphone.
Viewing iPhone regulatory information	On iPhone, go to Settings > General > About > Legal > Regulatory.
Battery replacement service	Go to www.apple.com/batteries/replacements.html.
Using iPhone in an enterprise environment	Go to www.apple.com/iphone/business to learn more about the enterprise features of iPhone, including Microsoft Exchange, IMAP, CalDAV, CardDAV, VPN, and more.

Using iPhone in an enterprise environment

Go to www.apple.com/iphone/business to learn more about the enterprise features of iPhone, including Microsoft Exchange, IMAP, CalDAV, CardDAV, VPN, and more.

Using iPhone with other carriers

Some carriers let you unlock iPhone for use with their network. To see if your carrier offers this option, go to support.apple.com/kb/HT1937.

Contact your carrier for authorization and setup information. You need to connect iPhone to iTunes to complete the process. Additional fees may apply.

For more information, go to support.apple.com/kb/HT5014.

Disposal and recycling information

Apple Recycling Program (available in some areas): For free recycling of your old mobile phone, a prepaid shipping label, and instructions, see www.apple.com/recycling.

iPhone disposal and recycling: You must dispose of iPhone properly according to local laws and regulations. Because iPhone contains electronic components and a battery, iPhone must be disposed of separately from household waste. When iPhone reaches its end of life, contact local authorities to learn about disposal and recycling options, or simply drop it off at your local Apple retail store or return it to Apple. The battery will be removed and recycled in an environmentally friendly manner. For more information, see www.apple.com/recycling.

Battery replacement: The lithium-ion battery in iPhone should be replaced only by Apple or an Apple Authorized Service Provider, and must be recycled or disposed of separately from household waste. For more information about battery replacement services, go to www.apple.com/batteries/replacements.html.

Battery Charger Efficiency

BC

Türkiye

Türkiye Cumhuriyeti: EEE Yönetmeliğine Uygundur.

台灣

廢電池請回收

European Union—Disposal Information

The symbol above means that according to local laws and regulations your product and/or its battery shall be disposed of separately from household waste. When this product reaches its end of life, take it to a collection point designated by local authorities. The separate collection and recycling of your product and/or its battery at the time of disposal will help conserve natural resources and ensure that it is recycled in a manner that protects human health and the environment.

Union Européenne—informations sur l'élimination: Le symbole ci-dessus signifie que, conformément aux lois et réglementations locales, vous devez jeter votre produit et/ou sa batterie séparément des ordures ménagères. Lorsque ce produit arrive en fin de vie, apportez-le à un point de collecte désigné par les autorités locales. La collecte séparée et le recyclage de votre produit et/ou de sa batterie lors de sa mise au rebut aideront à préserver les ressources naturelles et à s'assurer qu'il est recyclé de manière à protéger la santé humaine et l'environnement.

Europäische Union—Informationen zur Entsorgung: Das oben aufgeführte Symbol weist darauf hin, dass dieses Produkt und/oder die damit verwendete Batterie den geltenden gesetzlichen Vorschriften entsprechend und vom Hausmüll getrennt entsorgt werden muss. Geben Sie dieses Produkt zur Entsorgung bei einer offiziellen Sammelstelle ab. Durch getrenntes Sammeln und Recycling werden die Rohstoffreserven geschont und es ist sichergestellt, dass beim Recycling des Produkts und/oder der Batterie alle Bestimmungen zum Schutz von Gesundheit und Umwelt eingehalten werden.

Unione Europea—informazioni per lo smaltimento: Il simbolo qui sopra significa che, in base alle leggi e alle normative locali, il prodotto e/o la sua batteria dovrebbero essere riciclati separatamente dai rifiuti domestici. Quando il prodotto diventa inutilizzabile, portalo nel punto di raccolta stabilito dalle autorità locali. La raccolta separata e il riciclaggio del prodotto e/o della sua batteria al momento dello smaltimento aiutano a conservare le risorse naturali e assicurano che il riciclaggio avvenga nel rispetto della salute umana e dell'ambiente.

Europeiska unionen—information om kassering: Symbolen ovan betyder att produkten och/eller dess batteri enligt lokala lagar och bestämmelser inte får kastas tillsammans med hushållsavfallet. När produkten har tjänat ut måste den tas till en återvinningsstation som utsetts av lokala myndigheter. Genom att låta den uttjänta produkten och/eller dess batteri tas om hand för återvinning hjälper du till att spara naturresurser och skydda hälsa och miljö.

Brasil—Informações sobre descarte e reciclagem

O símbolo acima indica que este produto e/ou sua bateria não devem ser descartadas no lixo doméstico. Quando decidir descartar este produto e/ou sua bateria, faça-o de acordo com as leis e diretrizes ambientais locais. Para informações sobre o programa de reciclagem da Apple, pontos de coleta e telefone de informações, visite www.apple.com/br/environment.

Apple and the environment

At Apple, we recognize our responsibility to minimize the environmental impacts of our operations and products. For more information, go to www.apple.com/environment.

Mention of third-party products is for informational purposes only and constitutes neither an endorsement nor a recommendation. Apple assumes no responsibility with regard to the performance or use of these products. All understandings, agreements, or warranties, if any, take place directly between the vendors and the prospective users. Every effort has been made to ensure that the information in this manual is accurate. Apple is not responsible for printing or clerical errors.

019-2520/2013-01

Espacenet

Bibliographic data: CN108352094 (A) — 2018-07-31

Vending machine and method for distance selling regulated goods

Inventor(s):	PISHCHIK KIRILL EDUARDOVICH ± (K·E·皮希克)
Applicant(s):	PISHCHIK KIRILL EDUARDOVICH ± (K·E·皮希克)
Classification:	**- international:** *G07F11/00* **- cooperative:** G06Q10/0836 (KR); G06Q20/4014 (EP, KR, US); G07F11/00 (EP, KR, US); G07F17/0092 (EP, KR); G07F9/002 (EP, KR, US); G16H20/10 (EP, KR); G16H20/13 (EP, KR); G06Q10/0836 (EP)
Application number:	CN20168061694 20160816 Global Dossier
Priority number(s):	RU20150135449 20150821 ; WO2016RU00552 20160816
Also published as:	CN108352094 (B) AU2016312856 (A1) AU2016312856 (B2) EA038763 (B1) EA201890536 (A1) more

Abstract of CN108352094 (A)

The present invention relates to vending machines and methods for distance selling regulated goods. The claimed vending machine is equipped with modules for the return and reimbursement of goods, a communication unit for providing a communication link between the vending machine and a specialized operator station, and means which render the vending machine suitable for use by persons with poor vision, said means being graphical marks in the form of raised Braille. A method of distance selling regulated goods is realized using the proposed vending machine for distance selling and envisages monitoring and control, by a specialized operator, of the steps of order placement, verification of consent documents, payment, dispensing and, if necessary, return of the goods. The method provides for communication and the remote control of a vending machine and of a sales process by a specialized operator working at a computerized workstation within a specialized operator station.

Patent Translate
Powered by EPO and Google

Notice

This translation is machine-generated. It cannot be guaranteed that it is intelligible, accurate, complete, reliable or fit for specific purposes. Critical decisions, such as commercially relevant or financial decisions, should not be based on machine-translation output.

DESCRIPTION CN108352094A

Vending machine and method for remote sale of regulated goods

[0001]

technical field

[0002]

The present invention relates to an apparatus and method for automated remote sales, and more particularly to a vending machine for the sale of goods that require authorization or are age-restricted for purchase.

[0003]

Background technique

[0004]

Vending machines have been used extensively for the sale of a variety of single items involving everyday consumer-sized goods and offer advantages over outlets and venues in terms of accessibility, small footprint and low maintenance.

Vending machines are also used for the sale of age-restricted goods, such as tobacco and alcohol, and regulated goods, including pharmaceuticals.

Vending machines for selling prescription drugs have become more widespread, creating a need for vending machines with active sales control methods and sales specialist involvement.

[0005]

As we all know, there are commodity sales machines that require the identity of the buyer. Please refer to the utility model patent RU73106 published on October 5, 2008 by the Russian Federation, No. U1 [1], which includes a fuselage, a display box with a commodity sample (order unit), a device for storing and distributing goods (order distribution module), a data display device in the form of a display screen with an active screen (touch screen), a payment instrument in the form of a cash receiving device (cash receiving module), especially Is a cash acceptor with change and receipt printer (module to print receipt and change) function.

43 Vending machines can be equipped with devices for scanning ID scans, as well as video calls with sales specialists, if desired.

[0006]

48 Using this vending machine sales method, in order to complete all sales operations through the vending machine itself, the sale of one or several commodities is performed.

50 The method also includes the option of interrupting the sales process, or the sale of the goods requires identification of the buyer, sending the buyer's facial image through a webcam to contact a sales specialist in a call center, and scanning the device for identification, based on the buyer's age The result of the visual inspection of whether the age limit is met determines whether the next step can be performed by the vending machine.

[0007]

58 A similar solution was proposed in the application US2009/0276088[2] published on May 11, 2009.

59 The scheme relates to a method of selling age-restricted regulated goods and a vending machine used, and the scheme includes unlocking of the function of the vending machine after a visual inspection of a purchaser's ID by a sales officer.

[0008]

65 The above-mentioned automatic vending machines [1, 2] and methods for the sales of regulated goods basically only provide the result of identification based on the age of the purchaser and the further actions taken by the purchaser without the participation of the sales commissioner. Controlled unlocking of cargo aircraft functions,.

69 However, in order to ensure that regulated goods are authorized to be sold as intended, in the case where the sale of goods requires the submission of authorization documents through vending machines, such as medical prescriptions, the entire sales process will be participated through the duty of sales commissioners from the moment the shopping starts until the shopping is completed Rather than being controlled through the vending machine itself.

[0009]

77 There is a method of selling pharmaceuticals, including prescription drugs, through a vending machine where the purchaser inserts the prescription into a reading device and then pays for the purchase by means of a credit card using a cash slot (cash acceptor or coin acceptor) or using a card slot; Please refer to the electronic resources at this URL: http://www.1000ideas.ru/article/biznes/moda-krasota-zdorove/biznes-ideya-1886-avtomat-dlya-prodazhi-tabletok/[3] as an example.

82 After payment, the purchaser gets a receipt and a printed slip of the medication regimen.

83 During the drug purchase process, buyers can contact merchants, who will check medical prescriptions and provide consultation on medication regimens. The method is implemented in an automatic vending machine, and the automatic vending machine includes a body, a medical prescription interface, a payment module interface, a product delivery tray, a receipt and a medication plan printing module, and a device for audio/video communication with vendors. All operations from the beginning of the shopping process until the end of the shopping are performed by the vending machine without the participation of sales specialists.

[0010]

92 In order to solve the problem of sales of regulated goods, considering the lack of a device for checking and controlling the order of goods purchased during the period from product selection to product shipment, the automatic vending machine does not ensure sufficient security for the intended authorized sale of prescription drugs.

96 The function of this type of vending machine is limited to selling goods that do not need to be returned and refunded due to buyer's wrong operation.

[0011]

101 In addition, the vending machines in the related art have limited the use of certain categories of disabled people, especially those with poor eyesight, and the vending machines in the related art cannot ensure the identification of the buyer's identity guarantee, which increases the unplanned Risks of selling prescription drugs.

105 In addition, the medicine vending machines in the related art are inconvenient for purchasers to obtain a medicine certificate if necessary, because reading the certificate from a screen is inconvenient to operate, and it is basically impossible for people with poor eyesight to operate.

[0012]

111 Contents of the invention

[0013]

115 The purpose of the present invention is to develop a comprehensive facility for remote sales of regulated commodities and a sales method for regulated commodities with the participation of sales specialists, the sales of which are regulated by relevant laws and regulations, especially for use in accordance with basic laws and regulations Complexes, vending machines and methods for the sale of prescription drugs controlled by an operating pharmacist or salesperson.

[0014]

The proposed remote-controlled automatic vending machine and method for remote sales of regulated goods can achieve such a technical effect that the sales process can be safely and conveniently delivered to The process by which authorized persons sell regulated goods and the process by which vending machines can be facilitated for persons with disabilities, especially those with low vision.

The use of vending machines can not only reduce the price of medicines by reducing the rent and wages of business premises, but also eliminate geographical barriers through a large network of medicine vending machines, making medicines easier to obtain.

[0015]

The medicine vending machine has strong mobility and a small footprint. It can be installed on tourist routes and gas stations, making it more convenient for motor vehicle users to obtain medicines.

[0016]

Said technical effect is achieved by a remote-controlled vending machine comprising:

[0017]

The automatic vending machine for the remote sale of regulated commodities, which includes a fuselage and the following items installed in the fuselage:

[0018]

A communication unit, a software and hardware unit, a database and a data storage management server, which provide the possibility of establishing communication with a remote sales specialist station, where the sales specialist station includes at least one computerized workstation of a sales specialist for regulated goods, which Software and hardware unit for data collection and control of vending machines for the remote sale of regulated goods.

[0019]

A software and hardware unit for control directing: the acceptance and interpretation of the vending machine, the functional characteristics of the relevant vending machine, and the receipt, payment and dispatch of orders; the software and hardware unit contains audio and A video unit, the audio and video unit includes a display device, a video camera, a speaker and a microphone, and a sales specialist call button, wherein the data display unit is in the form of a touch screen with a buyer interface mounted on the front panel of the body , the video camera is located above the touch screen, the sales specialist call button is equipped with a plate with a graphic pointing point designated in Braille type, which is located on the front panel, and

[0020]

The following are controlled by the Sales Specialist,

[0021]

a commodity unit module which is acted as a container with a door on the front panel with a door surveillance camera,

[0022]

authorization document reading device in the form of a scanner, at least one payment module having a payment receiver located on the front panel of the fuselage,

[0023]

Commodity distribution module, which has a tray on the front panel of the fuselage, the commodity distribution module is equipped with a shipment monitoring camera, and commodity distribution is controlled by a rolling door,

[0024]

a change and receipt cashier module having a tray located on the front panel of the fuselage and equipped with a surveillance camera,

[0025]

A return module and a file storage module, the return module is on the side panel of the fuselage, the file storage module is on the side panel of the fuselage, and has a tray, and the return module and the file storage module are equipped with controlled rolling doors and and surveillance cameras for returns and document cashiers, here

[0026]

A plurality of panels with graphic indications of Braille type indications are provided adjacent to the payment acceptor, change and receipt cashier module, goods shipment module and returns module

[0027]

The job site of the sales specialist is the job site of a sales pharmacist or a business vendor.

[0028]

The commodity unit modules are preferably equipped with means to maintain the temperature and humidity conditions of the container.

[0029]

The method of commodity payment can be carried out in the form of a cash payment module and/or a credit card payment module.

[0030]

If desired, the authorization document reading device may be equipped with a container with a surveillance camera for the receipt of authorization documents (eg prescriptions) in the case of the sale of goods requiring authorization documents.

[0031]

A method of distance selling regulated goods according to claim 1, which may be carried out using a vending machine for distance selling according to claim 1, the method comprising

[0032]

Connecting, by wired or wireless link, one or more vending machines for the remote sale of regulated commodities to a sales specialist station comprising at least one computerized workstation, software and hardware unit of a sales specialist of regulated commodities, Database and data storage management server, software and hardware unit for data collection and control of vending machines for the remote sale of regulated goods.

[0033]

Activation of the vending machine is performed from standby mode by the purchaser touching the touch screen displaying the purchaser interface on the screen, or by pressing a call button on the front panel of the vending machine.

[0034]

The buyer establishes wired or wireless communication between the vending machine and the sales commissioner station by touching the icon "call" on the buyer interface or pressing the call button on the front panel of the vending machine,

[0035]

Determine the available sales specialist and upload the activated data of the vending machine to the sales specialist's computer work site, display the operation interface for controlling the functional devices and modules of the vending machine on the monitor of the sales specialist's work site, and the sales specialist starts the automatic Modes of operation of vending machine audio and video units,

[0036]

254 Display the list of goods requested by the buyer on the touch screen of the vending machine, use the scanner of the vending machine to submit the authorization documents for specific types of goods, and further display the list of selected goods on the screen of the sales specialist station,

[0037]

260 Display the total purchase amount on the touch screen of the vending machine, or inform the product quantity and price in the form of voice information of the sales specialist,

[0038]

265 Confirmation of selected items occurs by touching the selected item confirmation icon on the buyer interface, or by pressing a call button on the front panel of the vending machine.

[0039]

270 After specifying the customer payment option (cash/credit card), the vending machine payment module is unlocked.

[0040]

275 The payment for the selected commodity is completed through the payment module receiver, and the sales specialist uses the monitoring camera of the pallet of the commodity shipment module to further unlock the commodity shipment module and the change and receipt cashier module.

[0041]

281 Once the vending machine has not been used by the buyer for a period of time, the vending machine is placed into a standby mode by a sales specialist or automatically.

[0042]

286 During commodity selection, the sales specialist can provide commodity documents to the buyer through the tray of the document storage module according to the buyer's requirements, and the sales specialist uses the monitoring camera of the tray to further control the return of the documents.

[0043]

The method also offers the possibility of returns through a merchandise return pallet controlled by a sales specialist using a surveillance camera for the merchandise return pallet.

[0044]

The proposed vending machine for distance selling also provides a method of selling predetermined regulated goods, the method comprising:

[0045]

The purchaser receives information about the address and number of the vending machine, the time when the ordered goods were delivered to the vending machine and the unique password for the purchase,

[0046]

Activation of the vending machine is performed from standby mode by the purchaser touching the touch screen with the on-screen purchaser interface, or by pressing a call button on the front panel of the vending machine.

[0047]

The communication between the vending machine and the sales commissioner station is established by the buyer through a wired or wireless communication unit by touching the icon "call" on the buyer interface or pressing the call button on the front panel of the vending machine,

[0048]

The unique password is entered through the buyer interface on the vending machine screen, or by the sales representative based on the voice message sent by the buyer through the audio and video unit of the vending machine,

[0049]

Display the total purchase amount on the touch screen of the vending machine, or inform the product quantity and price in the form of voice information of the sales specialist,

[0050]

Confirmation of the order is made by touching the order confirmation icon on the buyer interface, or by pressing the call button on the front panel of the vending machine.

[0051]

The sales representative notifies the container number of the module with the commodity unit through the buyer interface or through the voice message.

[0052]

payment for the order by the buyer through the selected payment module receiver, further controlled by the sales specialist using the surveillance camera of the tray of the change and receipt distribution module to unlock the container door of the unit with the merchandise module and the change and receipt distribution module,

[0053]

The acquisition of the medicine in the correct container is controlled by the sales specialist using the surveillance camera 17 of the door 9 of the container,

[0054]

Once the vending machine has not been used by the buyer for a period of time, the vending machine is placed into a standby mode by a sales specialist or automatically.

[0055]

Description of drawings

[0056]

The proposed invention is explained in the accompanying drawings, in which:

[0057]

FIG. 1 shows a general view of a remote-controlled automatic vending machine for remote selling of regulated goods according to the present invention.

[0058]

FIG. 2 shows a flowchart of a vending machine using distance selling, which explains a method of remotely selling regulated goods.

[0059]

375 Detailed ways

[0060]

379 FIG. 1 is a schematic diagram of a general view of a remotely controlled vending machine for remotely selling regulated goods.

[0061]

384 The automatic vending machine includes a body 3, which is provided with equipment and functional modules for placing, storing and selling commodities.
386 The fuselage acts as a supporting frame, which is encapsulated with front and side panels and a rear panel, usually, the rear panel is in the form of a door to facilitate access to the devices and functional modules of the vending machine.
389 The airframe also includes a communication unit for communicating with a sales specialist at a remote station, a hardware and software unit that provides receipts and instructions for control instructions, for example, generated by a sales specialist for regulated commodities.

[0062]

395 The vending machine is equipped with an audio and video system comprising display means in the form of a touch screen 4 , a video camera 5 , speaker means in the form of powered speakers 6 a and 6 b , and a microphone 7 .
398 The camera 5 is arranged to capture images of the buyer's area as well as images of documents and other textual material submitted by persons with low vision.

[0063]

403 The powered loudspeaker is placed where it is convenient to listen to the voice information without noise, for example, above the data display unit.
405 The camera 5 is positioned so as to provide the widest possible capture of the buyer's area, for example it could be located in the middle above the touch screen 4 between the speakers as shown in Figure 2 .
407 Microphone 7 and sales commissioner call button 8 are positioned at the bottom of touch screen 4,

[0064]

411 A plate with a graphic indicator with a Braille type indication mounted on the surface of the call button 8 is used to inform people with poor eyesight of the type of button.
413 The touch screen is at eye level, and its height is determined according to the average height of a person.

[0065]

417 Audio and video systems are used for video conferencing between buyers and sales specialists, consultations during the merchandise purchase process, and consultations during the merchandise application process for shoppers.
420 In addition, a display device and a speaker device are used for familiarizing with commodities sold through the vending machine.
422 For example, in the brochure mode, buyers can browse leaflets for various products.
423 Specifically, during the sale of medicines, buyers can familiarize themselves with product descriptions and usage instructions, for example, with regard to medicines, when the vending machine is in standby mode, buyers can browse and listen to advertisements for various goods that may be demonstrated.

[0066]

429 In order to carry out the processes of ordering, payment and distribution, the automatic vending machine is equipped with a plurality of merchandising modules as containers of commodities (for example, medicines) in the size of daily consumer goods, including authorization document reading means, at least one payment receiving module, in particular Cash receiving module and/or credit card receiving module, change and receipt cashier module and payment goods shipping module.

[0067]

437 The unit modules for the consumer goods sized items to be sold are located inside the fuselage (not visible).
438 A modular unit acts as a container that can hold items within the standard range as well as items outside the standard range.
440 If goods outside the standard range are offered, they are accessed through the door 9 of the container, which is located to the left of the touch screen 4 on the front panel of the vending machine.
442 For the visually impaired, the door 9 is numbered and contains a tray 9a with a container number marking, a pictorial marking in Braille.

[0068]

447 The containers of the vendable merchandise modules are provided with means to maintain temperature and humidity conditions for merchandise storage.
449 For example, the container of the pre-ordered goods module of the medicine vending machine is filled with medicines outside the standard range of the medicine vending machine, which have been ordered earlier by the purchaser, for example through the company's website or through a similar medicine vending machine. Pre-orders for cargo planes, or pharmaceuticals manufactured through remote centralized prescription departments and via pre-orders.

[0069]

457 The work of the payment module is carried out by the receiver 10 of the cash (coin or banknote) receiving

module or the receiver 11 of the credit card module.

[0070]

462 Beneath the credit card acceptor is the scanner's window 12, which is used to enter and read the purchaser's authorization or identification document.
464 A container for receiving non-returnable authorization documents (eg, prescription blanks) is provided in a box near the scanner, and is equipped with a surveillance camera (not shown) for authorization documents (eg, medical prescriptions).
467 Below the scanner is located the tray 13 of the change and receipt cashier module, which is equipped with a camera (not shown) which is used by the salesperson to control the change receipt or refund.
469 Window 13 is used for refunds in case of wrong shipments (eg medication).

[0071]

473 The tray 14 of the commodity dispensing module is located in the middle area of the front panel, and is used for distributing goods according to orders.
475 Item assignment templates are used during purchase.
476 The pallet 14 of the module is provided with a rolling shutter 14b controlled by a sales specialist and a surveillance video camera (not shown) whose images are sent to the sales specialist's screen during distribution.
479 Thus, sales specialists control the distribution of those specific commodities, such as pharmaceuticals ordered and paid for by customers.

[0072]

484 The window 12 and trays 13, 14 are located at a height that facilitates manual handling during document scanning, pickup, receipt and change.
486 The tray 13 is also used for refunds in the event of misdispensing of goods, such as medicines.

[0073]

490 Plates of graphic indicators 10a, 11a, 13a and 14a with Braille type identification located near cash/credit card acceptors, merchandise dispensing trays and receipts, and change dispensing windows to facilitate separate payment and receipt of goods by the visually impaired using vending machines As well as getting back change and receipts.

[0074]

497 On the right side panel of the vending machine body is located the tray 15 of the returns module, which has a lockable roller door 15b, near which is placed the tray 16 of the board and document storage module, which has a graphic indicator in the form of a Braille type 15a, the tray 16 has a lockable roller shutter door 16b.