

**Ex. E-5 - Invalidity of U.S. Patent No. 8,327,426 against U.S. Patent Pub. No. 2007/0234408 to Burch et al. ("Burch")**

Fortinet, Inc. ("Fortinet") provides this chart subject to all reservations, objections, statements, and disclaimers set forth herein and in Fortinet's Preliminary Invalidity Contentions Cover Pleading, as well as any amendment, supplement, or modification thereof, which are incorporated herein by reference in their entirety.

On information and belief, and subject to further investigation and discovery, Burch was filed on December 8, 2005, and is thus available as prior art under at least §102(e).

As illustrated in the chart below, Burch anticipates the asserted claims of the '426 Patent. To the extent the Burch is found not to expressly disclose certain limitations in the asserted claims, such limitations are inherent. To the extent the Burch is found not to anticipate any asserted claims or claim elements of the '426 Patent, the reference nevertheless renders those claims or claim elements obvious under 35 U.S.C. § 103, either alone or in combination with other art identified in the cover pleading or herein. These Preliminary Invalidity Contentions are not an admission by Fortinet that the accused products, including any current or past versions of the accused products, are covered by, or infringe the asserted claims, but are based instead on the recognition that if the claims are interpreted to be broad enough to encompass the accused products, the claims must also be construed to have that same scope when considering whether they are invalid.

The following chart is partially based on, but is not limited by, the claim constructions implicit in Plaintiff's Infringement Contentions, to the extent that such constructions are apparent from the Infringement Contentions. Fortinet notes that in many instances, Plaintiff's Infringement Contentions fail to provide adequate notice of Plaintiff's construction of the asserted claims and fail to comply with the Court's scheduling order and other applicable rules. Fortinet does not accept the assumptions concerning the scope and meaning implicit in Plaintiff's Infringement Contentions, to the extent those assumptions are discernible, and reserves the right to challenge Plaintiff's proposed (or implied) constructions. Fortinet also reserves the right to revise and supplement these charts if and when Plaintiff is permitted to provide revised Infringement Contentions or otherwise make its positions known. To the extent that these Preliminary Invalidity Contentions rely on or otherwise embody particular constructions of terms or phrases in the asserted claims, Fortinet does not necessarily advocate any such construction as proper constructions of those terms or phrase. Fortinet also reserves the right to revise and supplement these charts after the Court construes the claims. Citations given in the chart below are merely representative of the respective elements and are not meant to be exhaustive.

Claim	Exemplary Citation from Burch
<p>[1PRE] A machine-implemented method to execute on a machine, comprising:</p>	<p>Burch discloses and/or renders obvious a machine-implemented method to execute on a machine, the following steps.</p> <p>Burch at Fig. 1 and corresponding text:</p> <pre> graph TD     100(( )) --&gt; 110[Intercept request directed to application]     110 --&gt; 115[Access Policy]     115 --&gt; 120[Forward request to application/ receive response]     120 --&gt; 125{Is the response an authentication request?}     125 -- No --&gt; 175([route response])     125 -- Yes --&gt; 130{Has the principal authenticated per policy?}     130 -- Yes --&gt; 160{Are the select credentials available?}     130 -- No --&gt; 135[Redirect request to Principal Service to obtain credentials]     160 -- Yes --&gt; 170[Fulfill the authentication request / cache select credentials]     160 -- No --&gt; 165[Obtain the select credentials]     170 --&gt; 175     165 --&gt; 170     135 --&gt; 140((User authenticates via an authentication service))     140 --&gt; 145[Receive returned request]     145 --&gt; 150[Obtain authentication credential]     150 --&gt; 155[Authenticate connection and cache credentials]     155 --&gt; 120   </pre> <p>Figure 1</p>

Claim

Exemplary Citation from Burch

Burch at Fig. 2 and corresponding text:

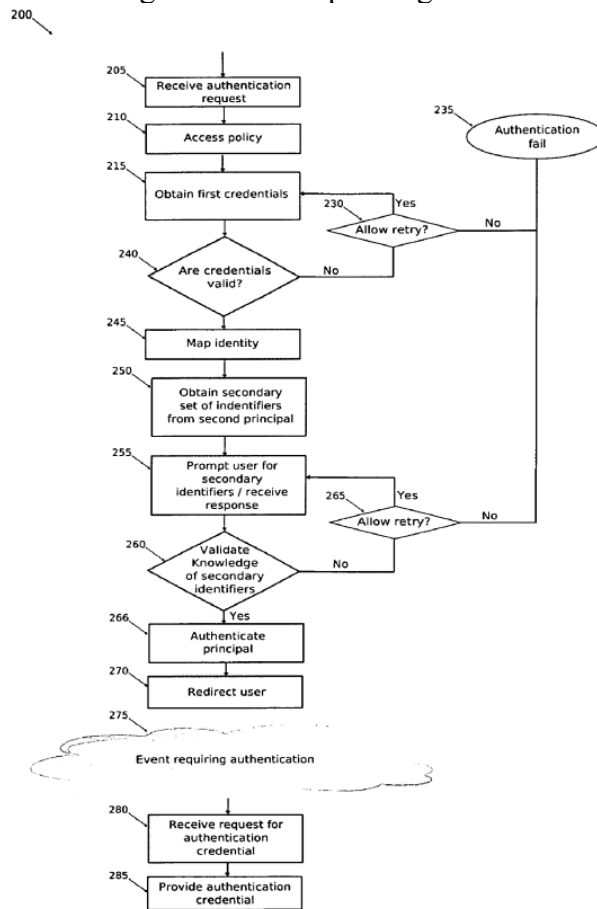
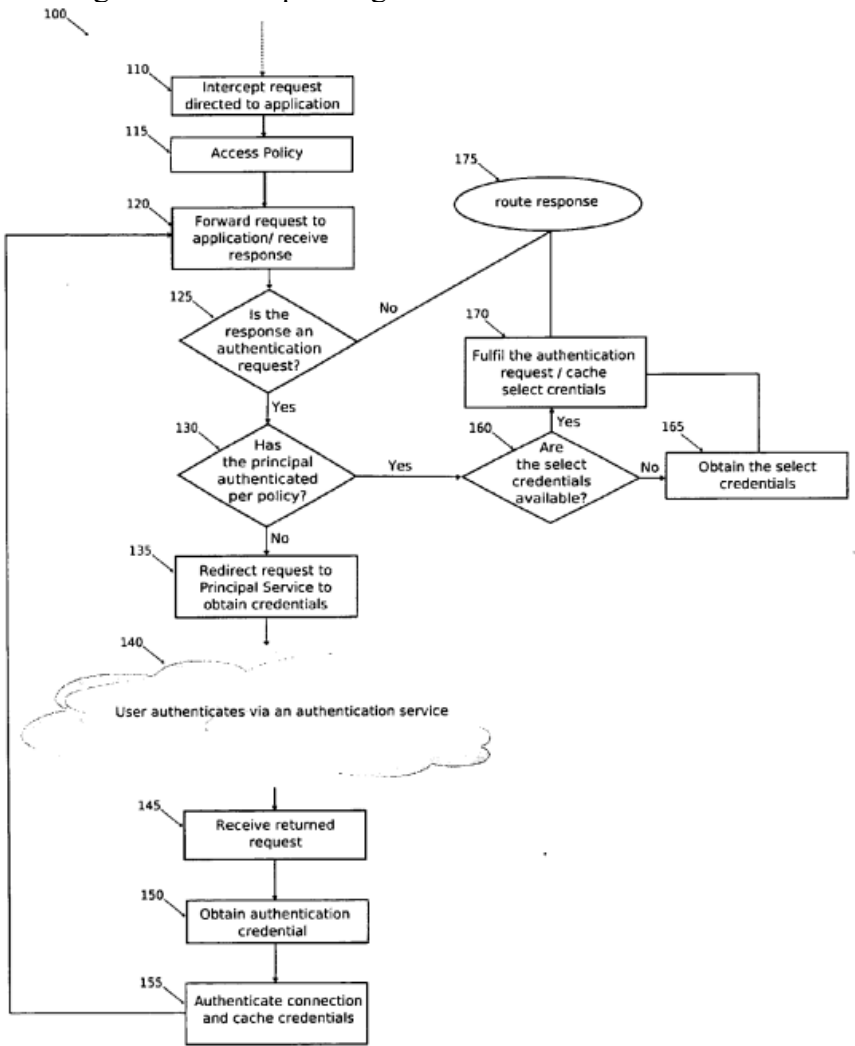


Figure 2

Claim	Exemplary Citation from Burch
	<p>Burch at Fig. 3 and corresponding text:</p> <p style="text-align: center;">Figure 3</p>
<p>[1A] receiving, by the machine, an authentication request from a principal, the request directed by the principal to an external service and intercepted by the method for receipt;</p>	<p>Burch discloses and/or renders obvious receiving, by the machine, an authentication request from a principal, the request directed by the principal to an external service and intercepted by the method for receipt.</p>

Claim	Exemplary Citation from Burch
	<p data-bbox="598 240 1102 267">Burch at Fig. 1 and corresponding text:</p>  <pre data-bbox="651 267 1501 1307"> graph TD     100(( )) --&gt; 110[Intercept request directed to application]     110 --&gt; 115[Access Policy]     115 --&gt; 120[Forward request to application / receive response]     120 --&gt; 125{Is the response an authentication request?}     125 -- No --&gt; 175([route response])     125 -- Yes --&gt; 130{Has the principal authenticated per policy?}     130 -- Yes --&gt; 160{Are the select credentials available?}     130 -- No --&gt; 135[Redirect request to Principal Service to obtain credentials]     135 --&gt; 140     140 --- 145[Receive returned request]     145 --&gt; 150[Obtain authentication credential]     150 --&gt; 155[Authenticate connection and cache credentials]     155 --&gt; 120     160 -- Yes --&gt; 170[Fulfill the authentication request / cache select credentials]     160 -- No --&gt; 165[Obtain the select credentials]     165 --&gt; 170     170 --&gt; 175 </pre> <p data-bbox="1039 1331 1144 1356">Figure 1</p>

**Claim**

**Exemplary Citation from Burch**

Burch at Fig. 2 and corresponding text:

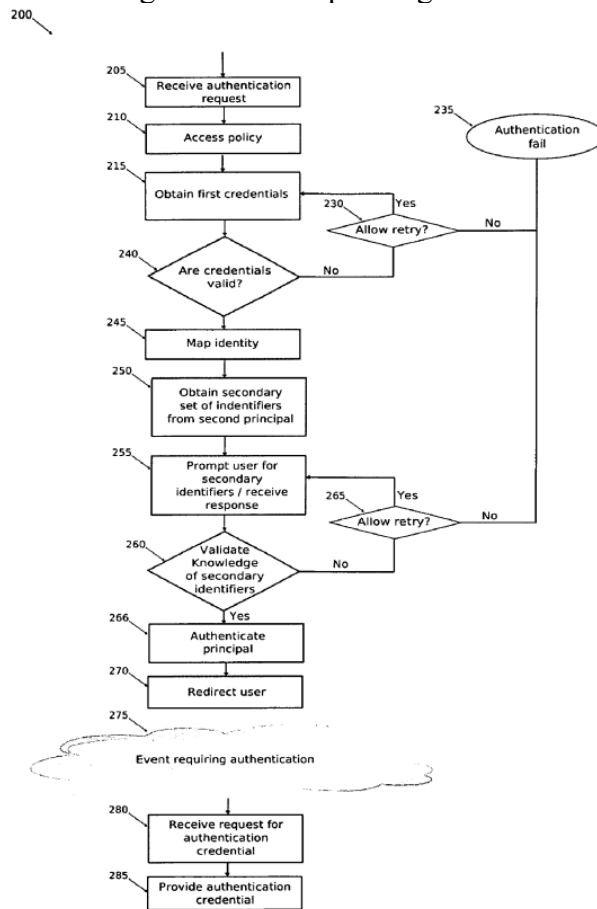


Figure 2

**Claim**

**Exemplary Citation from Burch**

Burch at Fig. 3 and corresponding text:

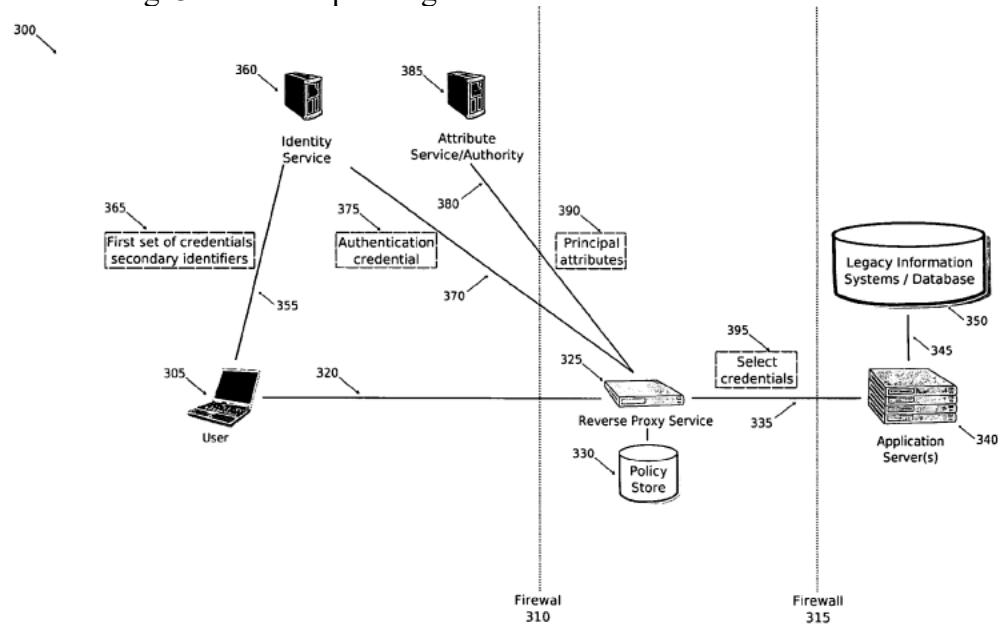


Figure 3

Burch at ¶ 31:

"The request is intercepted by the front-end service, at 110 . In an embodiment, at 115 , and upon receipt of the request, the front-end service accesses policy information. The policy information may indicate whether the first principal has to authenticate its identity before access to the second principal is granted. The policy information access, at 115 , may be performed in a number of different ways."

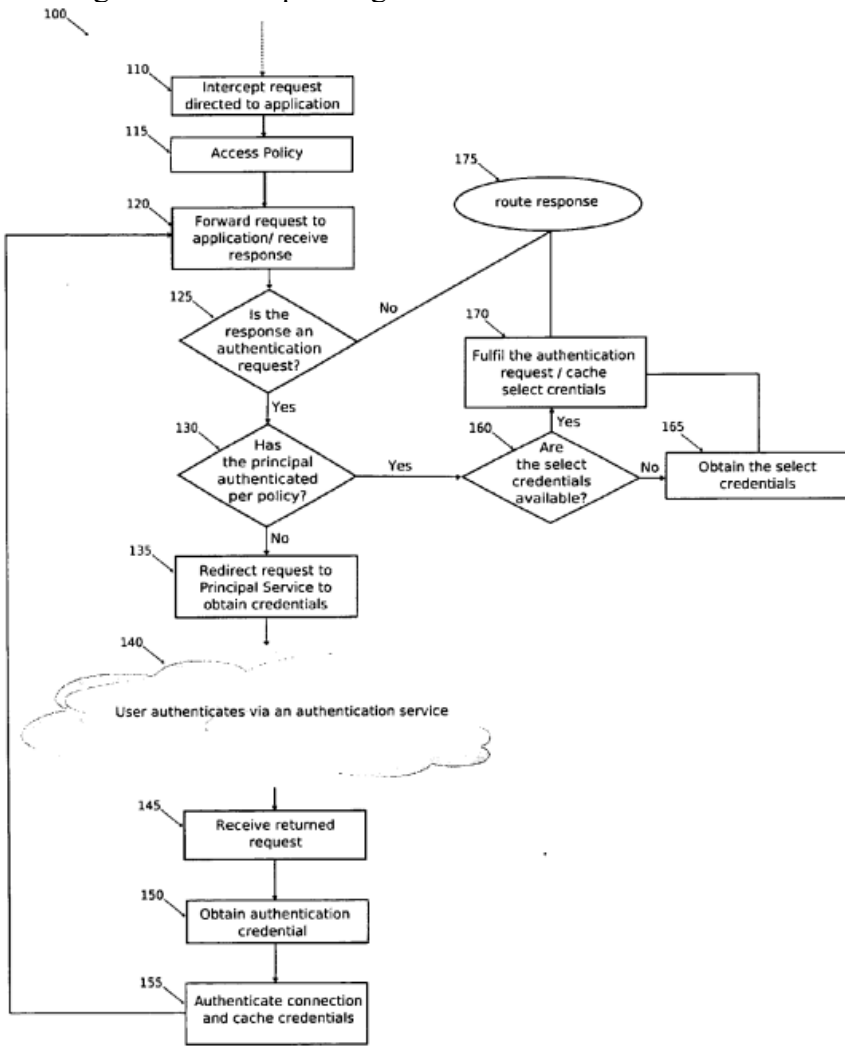
Burch at ¶43-44:

"Once the authentication credential for the first principal is been obtained, the front-end service may, at 155, verify the credential's authenticity and authenticate the first principal. If the authentication credential was obtained over a secure connection with the authentication service, the credential may

Claim	Exemplary Citation from Burch
	<p>be verified via the underlying communications protocol. For instance, if the communications link used to transfer the credential was secured via mutually authenticated secure socket layer (SSL), the identity of the authentication service providing the credential may have been validated as part of the handshake mechanism used to setup the SSL connection. Alternatively, the authentication credential itself may be cryptographically signed using techniques used with the Public Key Infrastructure (PKI) arts. Verifying such a signature may include: validating the signature on the credential against a public key certificate that is either included as part of the credential or otherwise available, verifying that the public key certificate used to validate the signature was itself signed by a trusted root certificate, and finally, checking the revocation status of the public key certificate by consulting a Certificate Revocation List (CRL) or making an Online Certificate Statute Protocol (OCSP) query to the issuer of the trusted root certificate. The trust relationships required for PKI authentication of the message may be specified by the policy. Verification of the message using PKI digital signatures is only one possible scenario of performing such verification; thus, the embodiments of the invention should not be read as being limited to just embodiments utilizing PKI verification techniques. After verifying the authenticity of the authentication credential obtained, the front-end service may authenticate the connection of the first principal, as depicted at 155. This may be performed in a number of different ways. In one embodiment, the use of web-browser cookies may be used to allow the front-end service to identify subsequent requests with the authenticated first principal. Alternatively, URL rewriting may be employed, which may encode session identifying information into the URL of subsequent requests routed through the front-end service. Again, it is noted that embodiments of the invention should not be read as being limited any particular method of connection authentication and/or session management as many other techniques may also be implemented without departing from the beneficial teachings presented herein."</p> <p>Burch at ¶ 50:  "Once the select credentials associated with the first principals are obtained, at 165, they may be cached for use in subsequent sessions or to authenticate the first principal to other principals associated with the front-end service. The select credentials may then be used to perform a proxy-authentication, as depicted at 170. Proxy-authentication includes authenticating the connection between the front-end service and the second principal with the identity of the first principal, thereby allowing the front-end service to make requests as a "proxy" of the first principal. This may be done in a variety of different manners. In an example, using the HTTP protocol, the "authorization header"</p>

Claim	Exemplary Citation from Burch
	<p>of subsequent HTTP requests made on behalf of the first principal may be processed to include the select credentials required by the second principal."</p> <p>Burch at ¶ 57:  "The request may be received as the result of an attempt by a first principal to access a second principal. So, a determination may have been made that the first principal must first authentication its identity before being granted access to the second principal. As such, the request may have been redirected to the authentication service in order authenticate its identity in the manner prescribed by a policy associated with the second principal as was discussed with respect to the front-end service represented by the method 100 of the FIG. 1."</p> <p>Burch at ¶67-68:  "Once the first principal has authenticated in accordance with the policy information, at 266, its identity may be authenticated. Authentication may include generating an authentication credential on behalf of the first principal for use in authenticating its identity to the second principal. In such a scenario, a token may be generated to allow the second principal, or entity acting on behalf of the second principal, to obtain the generated authentication credential. After successfully authenticating the identity of the first principal, at 270, the authentication service may redirect the request to the initially identified target resource or second principal. If an authentication credential was generated, at 266, the redirection URL may include the associated token. Or, in the event the Liberty or SAML POST profile is being used, the authentication credential generated, at 266, may be embedded into the redirected request. There are a number of different mechanisms and techniques that may be employed to include the credential or credential identifying information with the redirected request, the embodiments of this invention should be not read as limited to any particular technique or mechanism."</p> <p>Burch at ¶69:  "The redirection, at 270, may have directed the first principal to a resource provided by a second principal, as depicted at 275. In this case, the receiver of the request (a front-end service, such as the one presented with respect to the method 100 of the FIG. 1 that provides access control to the second principal) may identify the request as including authentication information. As such, the receiver may issue a request to obtain the authentication credential associated with the first principal, the request</p>

Claim	Exemplary Citation from Burch
	<p>being received at 280. The request may be issued over a communication link secured via mutually authenticated SSL. The request, at 280, may include identifying information included in the redirection URL, at 270. Alternatively, if no authentication credential for the first principal was generated, at 240, but instead a connection or session based authentication scheme was used, the authentication service may generate a credential for the first principal when the request, at 275, is received. At 285 the authentication credential may be provided to the requester who may then verify the credential and authenticate the first principal, providing the first principal with access to the application (second principal or resources of the second principal)."</p> <p>Burch at ¶ 92:          "In an embodiment, user 405 (first principal) may wish to access an application hosted by application server 440 (second principal). The user 405 may attempt to access the application via a WWW browser using an HTTP protocol via communication channel 420. The firewall 410 may be configured to allow the incoming connection to reach front-end service 425 configured to receive, process, and route traffic intended for application servers 440. Firewall 415 may be configured to allow the front-end service 425 to use communication link 435 to access application servers 440. Upon receipt of the request, the front-end service 425 may determine that access to the application server 440 is predicated upon the proper authentication of user 405. Alternatively, the applications running in conjunction with front-end service 425 may detect an authentication request from application servers 440 is responsive to a request from user 405. In order to authenticate the identity of user 405, the front-end service 425 may cause the request from user 405 to be redirected over communications channel 455 to an authentication service or an identity service 460."</p>
<p><b>[1B]</b> authenticating, by the machine, the principal; and</p>	<p>Burch discloses and/or renders obvious authenticating, by the machine, the principal.</p>

Claim	Exemplary Citation from Burch
	<p data-bbox="598 235 1102 267">Burch at Fig. 1 and corresponding text:</p>  <pre data-bbox="651 267 1491 1307"> graph TD     100(( )) --&gt; 110[Intercept request directed to application]     110 --&gt; 115[Access Policy]     115 --&gt; 120[Forward request to application / receive response]     120 --&gt; 125{Is the response an authentication request?}     125 -- No --&gt; 175([route response])     125 -- Yes --&gt; 130{Has the principal authenticated per policy?}     130 -- Yes --&gt; 160{Are the select credentials available?}     130 -- No --&gt; 135[Redirect request to Principal Service to obtain credentials]     135 --&gt; 140     140 --- 145[Receive returned request]     145 --&gt; 150[Obtain authentication credential]     150 --&gt; 155[Authenticate connection and cache credentials]     155 --&gt; 120     160 -- Yes --&gt; 170[Fulfill the authentication request / cache select credentials]     160 -- No --&gt; 165[Obtain the select credentials]     165 --&gt; 170     170 --&gt; 175   </pre> <p data-bbox="1039 1323 1144 1356">Figure 1</p>

**Claim**

**Exemplary Citation from Burch**

Burch at Fig. 2 and corresponding text:

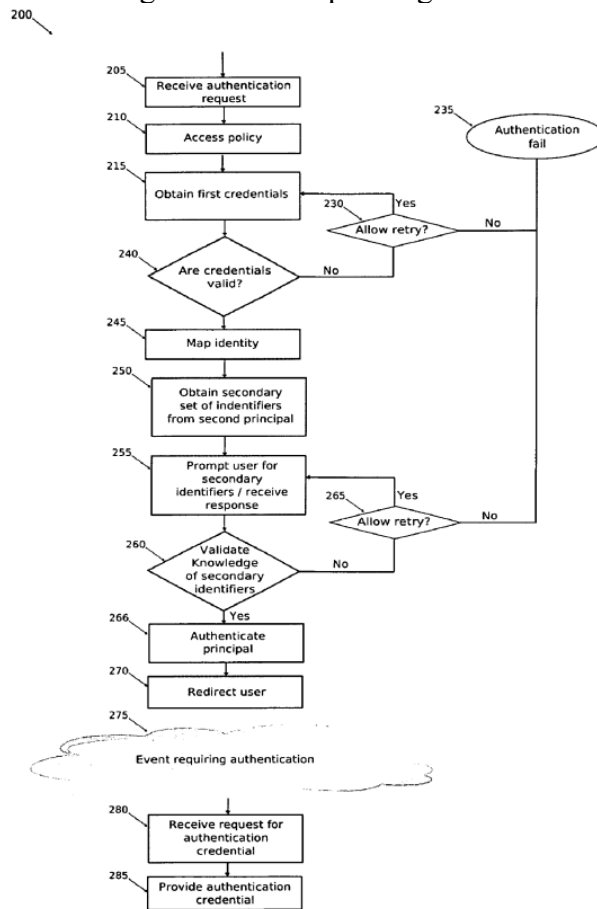


Figure 2

**Claim**

**Exemplary Citation from Burch**

Burch at Fig. 3 and corresponding text:

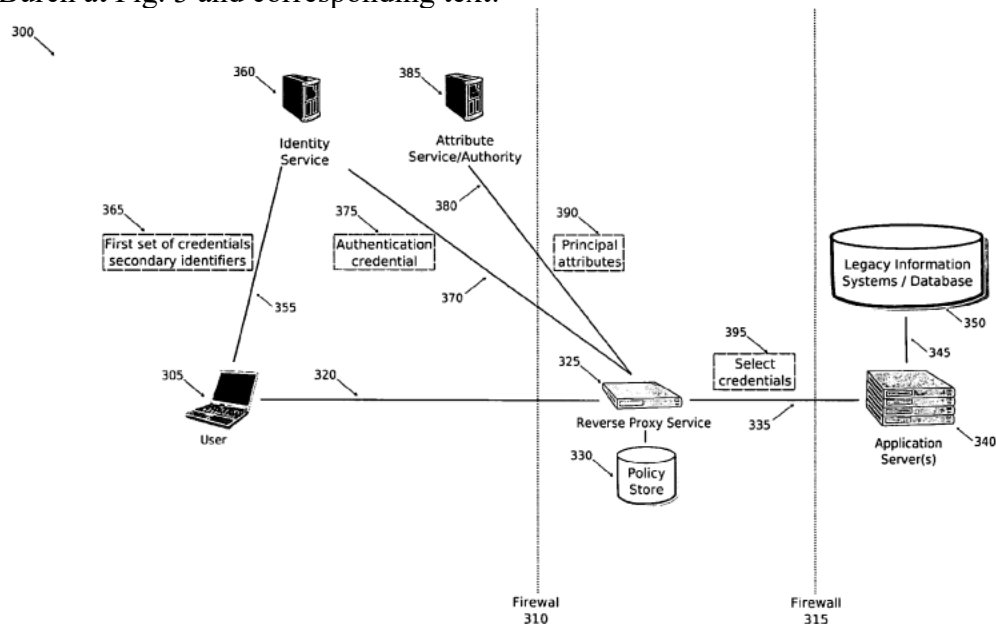


Figure 3

Burch at ¶ 31:

"The request is intercepted by the front-end service, at 110 . In an embodiment, at 115 , and upon receipt of the request, the front-end service accesses policy information. The policy information may indicate whether the first principal has to authenticate its identity before access to the second principal is granted. The policy information access, at 115 , may be performed in a number of different ways."

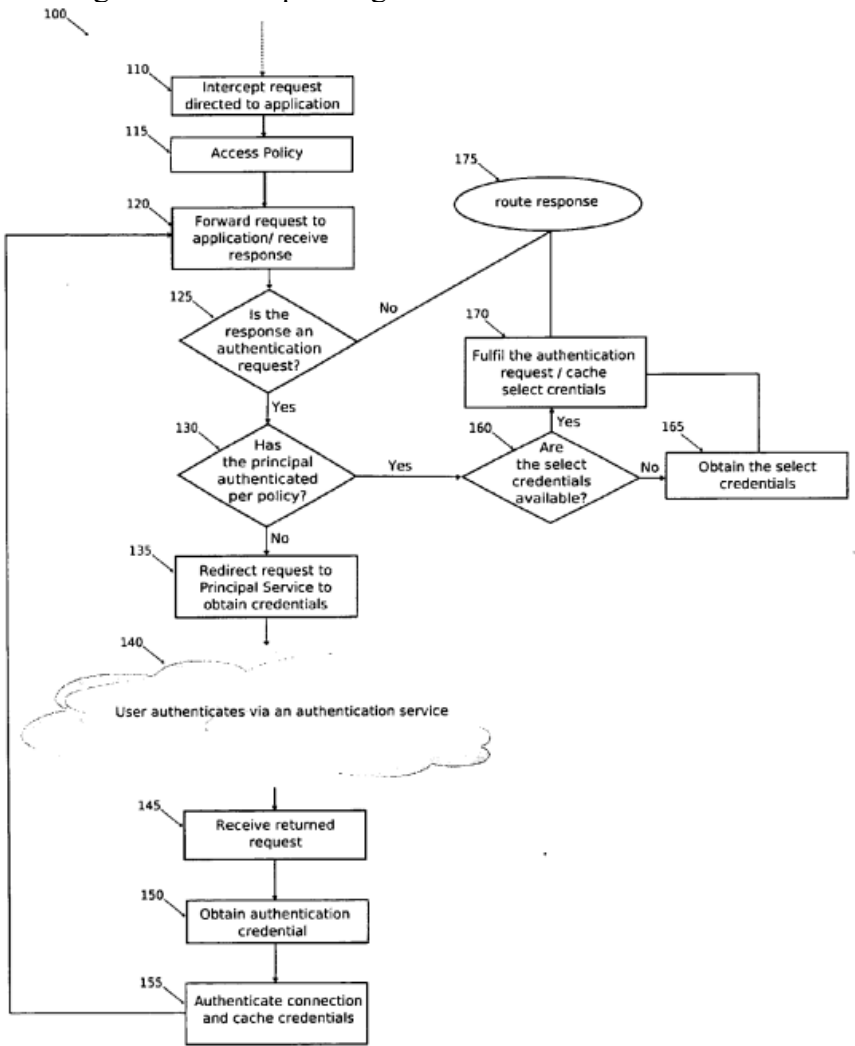
Burch at ¶43-44:

"Once the authentication credential for the first principal is been obtained, the front-end service may, at 155, verify the credential's authenticity and authenticate the first principal. If the authentication credential was obtained over a secure connection with the authentication service, the credential may

Claim	Exemplary Citation from Burch
	<p>be verified via the underlying communications protocol. For instance, if the communications link used to transfer the credential was secured via mutually authenticated secure socket layer (SSL), the identity of the authentication service providing the credential may have been validated as part of the handshake mechanism used to setup the SSL connection. Alternatively, the authentication credential itself may be cryptographically signed using techniques used with the Public Key Infrastructure (PKI) arts. Verifying such a signature may include: validating the signature on the credential against a public key certificate that is either included as part of the credential or otherwise available, verifying that the public key certificate used to validate the signature was itself signed by a trusted root certificate, and finally, checking the revocation status of the public key certificate by consulting a Certificate Revocation List (CRL) or making an Online Certificate Statute Protocol (OCSP) query to the issuer of the trusted root certificate. The trust relationships required for PKI authentication of the message may be specified by the policy. Verification of the message using PKI digital signatures is only one possible scenario of performing such verification; thus, the embodiments of the invention should not be read as being limited to just embodiments utilizing PKI verification techniques. After verifying the authenticity of the authentication credential obtained, the front-end service may authenticate the connection of the first principal, as depicted at 155. This may be performed in a number of different ways. In one embodiment, the use of web-browser cookies may be used to allow the front-end service to identify subsequent requests with the authenticated first principal. Alternatively, URL rewriting may be employed, which may encode session identifying information into the URL of subsequent requests routed through the front-end service. Again, it is noted that embodiments of the invention should not be read as being limited any particular method of connection authentication and/or session management as many other techniques may also be implemented without departing from the beneficial teachings presented herein."</p> <p>Burch at ¶ 50:  "Once the select credentials associated with the first principals are obtained, at 165, they may be cached for use in subsequent sessions or to authenticate the first principal to other principals associated with the front-end service. The select credentials may then be used to perform a proxy-authentication, as depicted at 170. Proxy-authentication includes authenticating the connection between the front-end service and the second principal with the identity of the first principal, thereby allowing the front-end service to make requests as a "proxy" of the first principal. This may be done in a variety of different manners. In an example, using the HTTP protocol, the "authorization header"</p>

Claim	Exemplary Citation from Burch
	<p>of subsequent HTTP requests made on behalf of the first principal may be processed to include the select credentials required by the second principal."</p> <p>Burch at ¶ 57:  "The request may be received as the result of an attempt by a first principal to access a second principal. So, a determination may have been made that the first principal must first authentication its identity before being granted access to the second principal. As such, the request may have been redirected to the authentication service in order authenticate its identity in the manner prescribed by a policy associated with the second principal as was discussed with respect to the front-end service represented by the method 100 of the FIG. 1."</p> <p>Burch at ¶67-68:  "Once the first principal has authenticated in accordance with the policy information, at 266, its identity may be authenticated. Authentication may include generating an authentication credential on behalf of the first principal for use in authenticating its identity to the second principal. In such a scenario, a token may be generated to allow the second principal, or entity acting on behalf of the second principal, to obtain the generated authentication credential. After successfully authenticating the identity of the first principal, at 270, the authentication service may redirect the request to the initially identified target resource or second principal. If an authentication credential was generated, at 266, the redirection URL may include the associated token. Or, in the event the Liberty or SAML POST profile is being used, the authentication credential generated, at 266, may be embedded into the redirected request. There are a number of different mechanisms and techniques that may be employed to include the credential or credential identifying information with the redirected request, the embodiments of this invention should be not read as limited to any particular technique or mechanism."</p> <p>Burch at ¶69:  "The redirection, at 270, may have directed the first principal to a resource provided by a second principal, as depicted at 275. In this case, the receiver of the request (a front-end service, such as the one presented with respect to the method 100 of the FIG. 1 that provides access control to the second principal) may identify the request as including authentication information. As such, the receiver may issue a request to obtain the authentication credential associated with the first principal, the request</p>

Claim	Exemplary Citation from Burch
	<p>being received at 280. The request may be issued over a communication link secured via mutually authenticated SSL. The request, at 280, may include identifying information included in the redirection URL, at 270. Alternatively, if no authentication credential for the first principal was generated, at 240, but instead a connection or session based authentication scheme was used, the authentication service may generate a credential for the first principal when the request, at 275, is received. At 285 the authentication credential may be provided to the requester who may then verify the credential and authenticate the first principal, providing the first principal with access to the application (second principal or resources of the second principal)."</p> <p>Burch at ¶ 92:          "In an embodiment, user 405 (first principal) may wish to access an application hosted by application server 440 (second principal). The user 405 may attempt to access the application via a WWW browser using an HTTP protocol via communication channel 420. The firewall 410 may be configured to allow the incoming connection to reach front-end service 425 configured to receive, process, and route traffic intended for application servers 440. Firewall 415 may be configured to allow the front-end service 425 to use communication link 435 to access application servers 440. Upon receipt of the request, the front-end service 425 may determine that access to the application server 440 is predicated upon the proper authentication of user 405. Alternatively, the applications running in conjunction with front-end service 425 may detect an authentication request from application servers 440 is responsive to a request from user 405. In order to authenticate the identity of user 405, the front-end service 425 may cause the request from user 405 to be redirected over communications channel 455 to an authentication service or an identity service 460."</p>
<p>[1C] supplying, by the machine, an authentication message for use by an identity service on behalf of the principal, the authentication message serves as a new authentication request and as a new authentication response for single sign-on</p>	<p>Burch discloses and/or renders obvious supplying, by the machine, an authentication message for use by an identity service on behalf of the principal, the authentication message serves as a new authentication request and as a new authentication response for single sign-on access of the principal to the identity service and other services external or internal to the identity service.</p>

Claim	Exemplary Citation from Burch
<p>access of the principal to the identity service and other services external or internal to the identity service,</p>	<p>Burch at Fig. 1 and corresponding text:</p>  <pre> graph TD     100(( )) --&gt; 110[Intercept request directed to application]     110 --&gt; 115[Access Policy]     115 --&gt; 120[Forward request to application / receive response]     120 --&gt; 125{Is the response an authentication request?}     125 -- No --&gt; 175([route response])     125 -- Yes --&gt; 130{Has the principal authenticated per policy?}     130 -- Yes --&gt; 160{Are the select credentials available?}     130 -- No --&gt; 135[Redirect request to Principal Service to obtain credentials]     135 --&gt; 140     140 --- 145[Receive returned request]     145 --&gt; 150[Obtain authentication credential]     150 --&gt; 155[Authenticate connection and cache credentials]     155 --&gt; 120     160 -- Yes --&gt; 170[Fulfill the authentication request / cache select credentials]     160 -- No --&gt; 165[Obtain the select credentials]     165 --&gt; 170     170 --&gt; 175   </pre> <p>Figure 1</p>

**Claim**

**Exemplary Citation from Burch**

Burch at Fig. 2 and corresponding text:

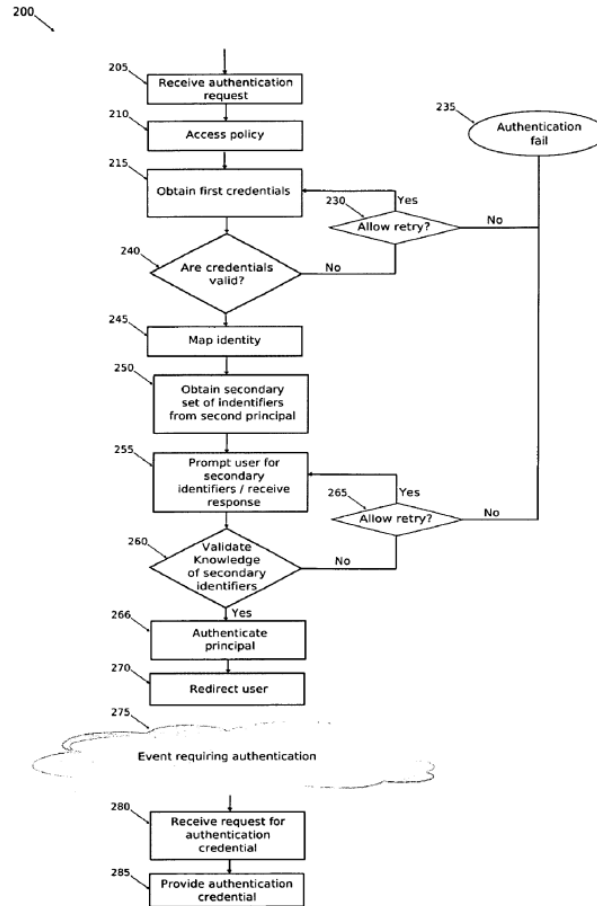


Figure 2

**Claim**

**Exemplary Citation from Burch**

Burch at Fig. 3 and corresponding text:

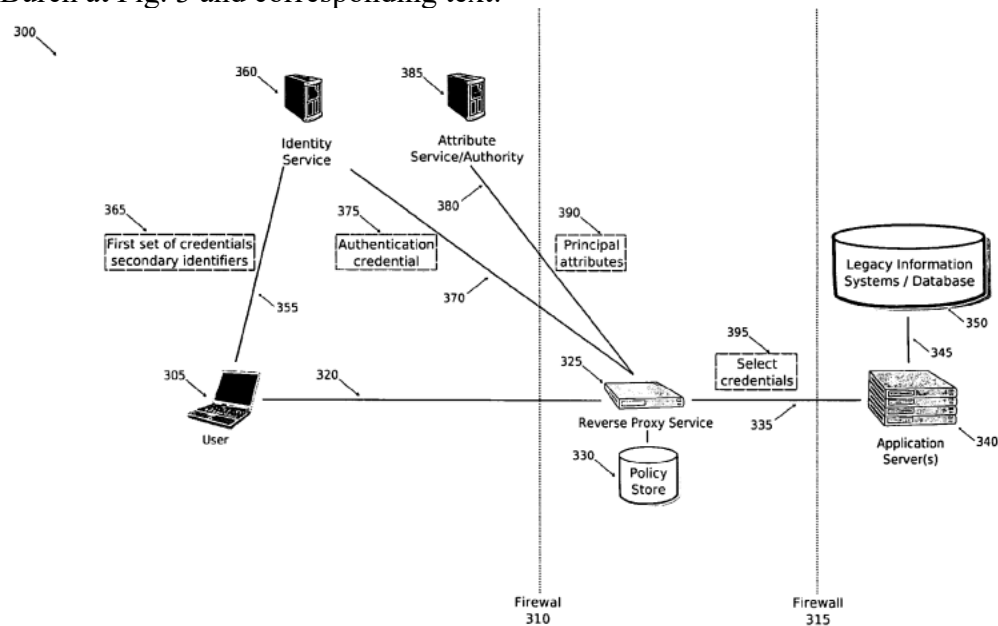


Figure 3

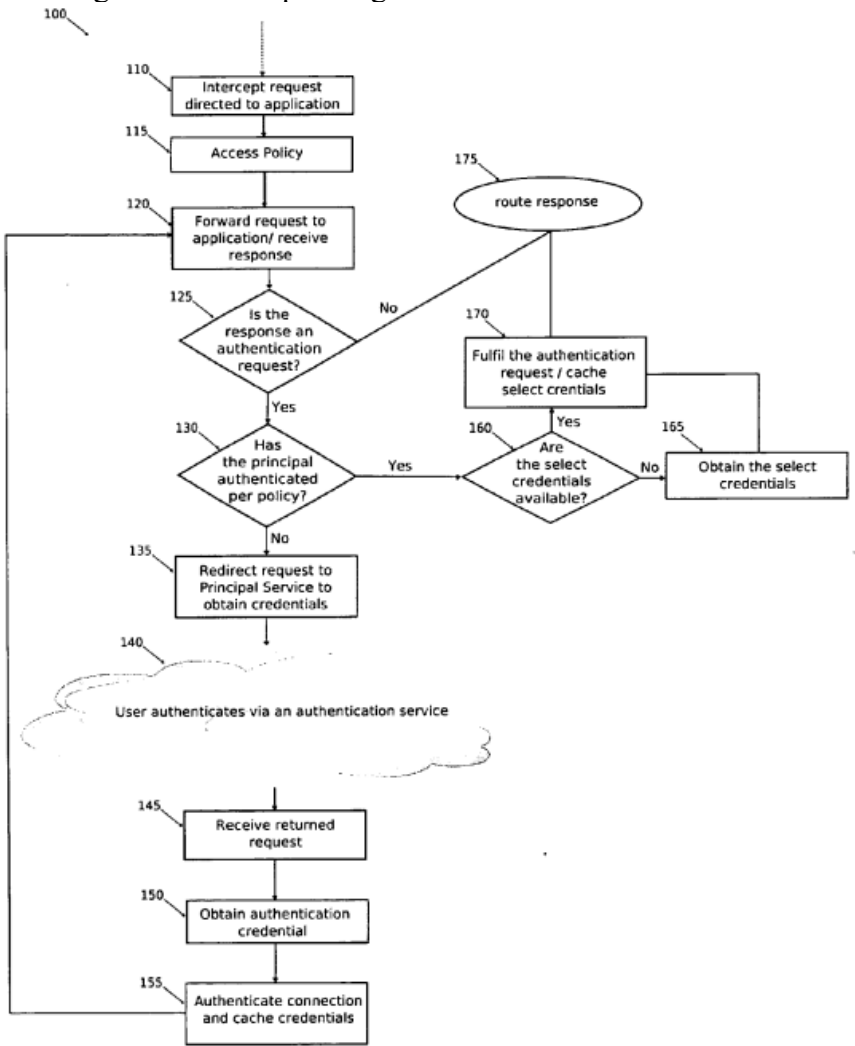
Burch at ¶45-46:

"After successfully authenticating the session of the first principal, at 155, the first principal's request may be reissued to the second principal back at the processing depicted in 120. Reissuing the original request may be done using a variety of different mechanisms. For example, in the Liberty and SAML context, the received request, at 145, may contain a URL tag indicating a resource that the first principal ultimately wishes to access after authentication; in the Liberty and SAML specifications this is referred to as the target URL. In such a scenario, the front-end service may parse the target URL from the request URL received, at 145, and reissue it to the second principal, at 120. Alternatively, the front-end service may cache the request URL before redirecting the first principal to the authentication service, at 135. The request is reissued to the second principal, at 120, and the response from the second principal is obtained. The front-end service may determine, at 125, whether

Claim	Exemplary Citation from Burch
	<p>the response from the second principal is an authentication request. Since the reissued request is the same that prompted an authentication request previously, the response, at 120, will likely be an authentication request. At 130, the front-end service determines whether the first principal has authenticated its identity, as required by the policy information. Since the first principal was redirected to the authentication service, at 140, and a credential authenticating the principal's identity was retrieved, at 150, this determining is positive."</p> <p>Burch at ¶ 51-52:  "Thus, as part of the message processing function associated with the front-end service, the front-end service may set the authorization header on each message received from the first principal as it is routed to the second principal. Under this scenario, the second principal validates the information provided in the HTTP authorization header and authenticates the connection of the front-end service as if it were the first principal. The front-end service may include the HTTP authorization header in subsequent requests from the first principal to the second principal for the duration of the first principal's session, as may also be stipulated by the policy information. From the perspective of the second principal, it appears that the first principal itself has directly authenticated using its own means of authentication, and the front-end service may route the reply messages from the second principal to the first principal, at 175. Thus, the first principal is granted seamless access to the second principal. And access to the second principal has been protected using a multifactor authentication mechanism, without altering the processing flow or processing aspects associated with the second principal. In other words, the multifactor authentication technique was implemented and enforced with no changes or modifications to the second principal. Using this technique, a high level of security is achievable with a minimal impact on existing or new application services that have no native multifactor authentication functionality."</p> <p>Burch at ¶69:  "The redirection, at 270, may have directed the first principal to a resource provided by a second principal, as depicted at 275. In this case, the receiver of the request (a front-end service, such as the one presented with respect to the method 100 of the FIG. 1 that provides access control to the second principal) may identify the request as including authentication information. As such, the receiver may issue a request to obtain the authentication credential associated with the first principal, the request being received at 280. The request may be issued over a communication link secured via mutually</p>

Claim	Exemplary Citation from Burch
	<p>authenticated SSL. The request, at 280, may include identifying information included in the redirection URL, at 270. Alternatively, if no authentication credential for the first principal was generated, at 240, but instead a connection or session based authentication scheme was used, the authentication service may generate a credential for the first principal when the request, at 275, is received. At 285 the authentication credential may be provided to the requester who may then verify the credential and authenticate the first principal, providing the first principal with access to the application (second principal or resources of the second principal)."</p> <p>Burch at ¶79:  "Upon receipt of the redirected request, the reverse proxy 325 may identify authentication information embedded within it. This information could include a token identifying an authentication credential 375 encoded within the URL as in the Liberty and SAML specifications. Alternatively, the request itself may include an authentication credential 375 as an HTTP POST parameter. There are a number of ways authentication information may be included in the redirected request. The embodiments of the invention should not be read as limited to any particular technique."</p> <p>Burch at ¶ 83:  "Before issuing the target URL request to the application servers 340 , the reverse proxy server 325 may access the policy store 330 . The policy information 330 may indicate that the application servers 340 themselves require select credentials 395 in order to authenticate the user 305 . This situation could arise if the application servers 340 originally included a proprietary authentication system. Thus, rather than modifying application server code, the reverse proxy 325 may be configured to provide the select credentials 395 that the application servers 340 expect via 335 . Similarly, the application servers 340 may themselves need to access a legacy information system (IS) or database 350 via link 345 in order to provide the services requested by user 305 . As such, the application servers 340 may need the select credentials 395 in order to access data specific to user 305 as required by the application. Given the disparate uses of the credentials 395 by the application servers 340 (some of which may not be strictly security related), the select credentials 395 may be completely distinct from those employed by user 305 to authenticate its identity to identity service 360 ."</p>

Claim	Exemplary Citation from Burch
	<p>Burch at ¶ 100-101:            "After obtaining the one or more secondary identifiers 497, user 405 may be prompted to verify its knowledge of the identifiers 497 via communications link 455. The communications link 455 may be secured via SSL to prevent disclosure of the credentials. The user 405 may respond to the request for the identifiers 497 via the link 455. Upon receipt of the identifiers 497 the authentication service 460 may verify the user's 405 knowledge of the identifiers 497. If successful, the user 405 may be expected, depending upon the policy information 465, to respond to additional authentication queries regarding the secondary identifiers 497. Further, upon verification of a first set of secondary identifiers 470, the authentication service 460 may be required by the policy 465, to obtain and verify the user's 405 knowledge of additional secondary identifiers 497. Once user 405 has authenticated in accordance with the policy 465, the authentication service 460 may generate an authentication credential 480 for the user 405. The credential 480 may be used to authenticate the identity of user 405 to other sites and/or principals 440. User 405 may then be redirected back to the application server 440. The URL used to redirect user 405 may include a token identifying the authentication service 460 and authenticated credential 480 generated for user 405. The token may be encoded into the URL string in accordance with the Liberty or SAML specifications. Alternatively, the authentication credential 480 may be encoded into the request itself. For instance, if the authentication service 460 were acting as a SAML Authentication Authority performing the SAML POST profile, the credential 480 transported via the as the payload of an HTTP POST to the application server 440."</p>
<p>[1D] the identity service acts as a proxy for access sessions to the other services on behalf of the principal,</p>	<p>Burch discloses and/or renders obvious that the identity service acts as a proxy for access sessions to the other services on behalf of the principal.</p>

Claim	Exemplary Citation from Burch
	<p data-bbox="598 240 1102 267">Burch at Fig. 1 and corresponding text:</p>  <pre data-bbox="651 267 1501 1307"> graph TD     100(( )) --&gt; 110[Intercept request directed to application]     110 --&gt; 115[Access Policy]     115 --&gt; 120[Forward request to application / receive response]     120 --&gt; 125{Is the response an authentication request?}     125 -- No --&gt; 175([route response])     125 -- Yes --&gt; 130{Has the principal authenticated per policy?}     130 -- Yes --&gt; 160{Are the select credentials available?}     130 -- No --&gt; 135[Redirect request to Principal Service to obtain credentials]     135 --&gt; 140     140 --- 145[Receive returned request]     145 --&gt; 150[Obtain authentication credential]     150 --&gt; 155[Authenticate connection and cache credentials]     155 --&gt; 120     160 -- Yes --&gt; 170[Fulfill the authentication request / cache select credentials]     160 -- No --&gt; 165[Obtain the select credentials]     165 --&gt; 170     170 --&gt; 175   </pre> <p data-bbox="1039 1331 1144 1356">Figure 1</p>

Claim

Exemplary Citation from Burch

Burch at Fig. 2 and corresponding text:

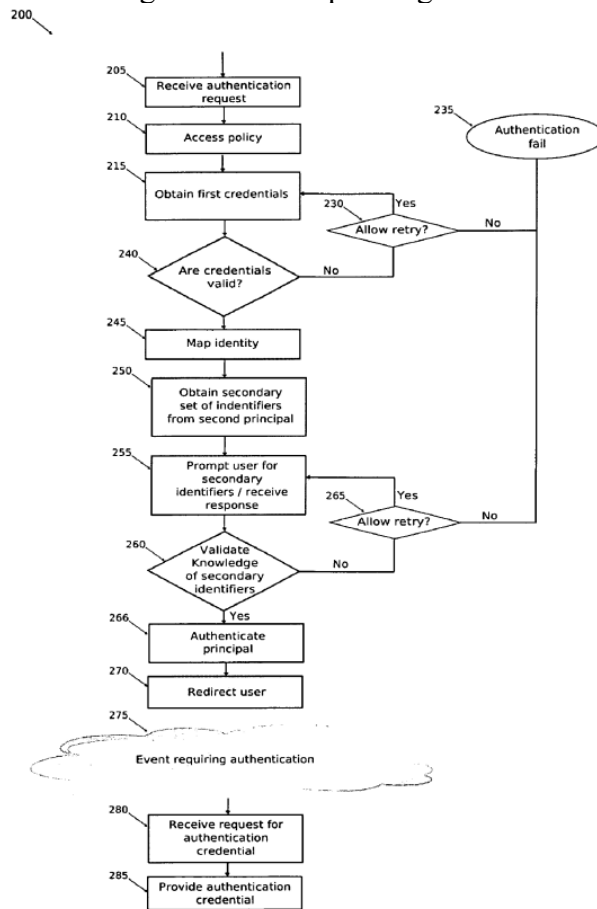


Figure 2

**Claim**

**Exemplary Citation from Burch**

Burch at Fig. 3 and corresponding text:

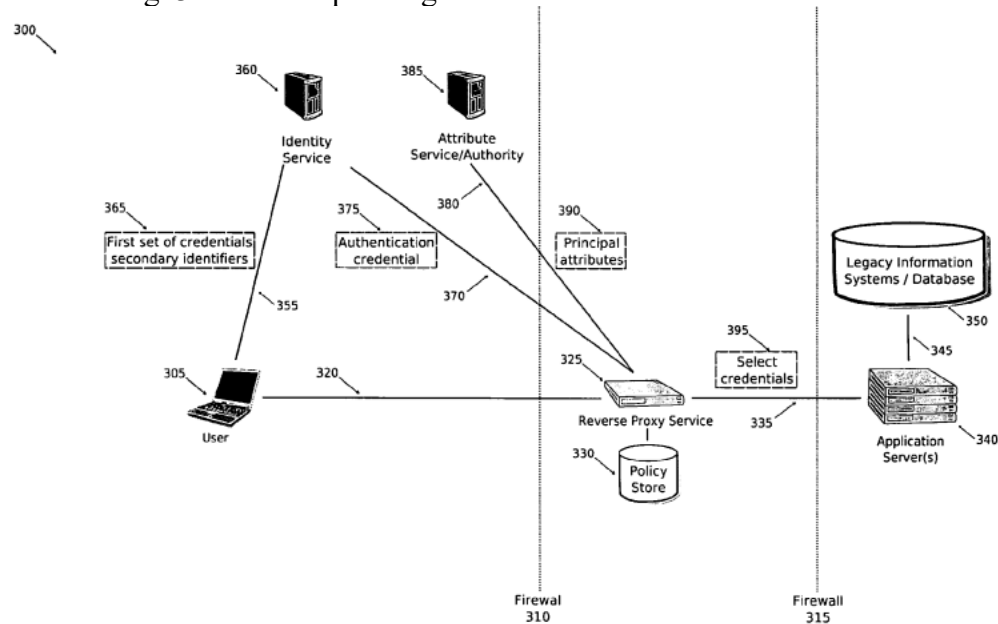


Figure 3

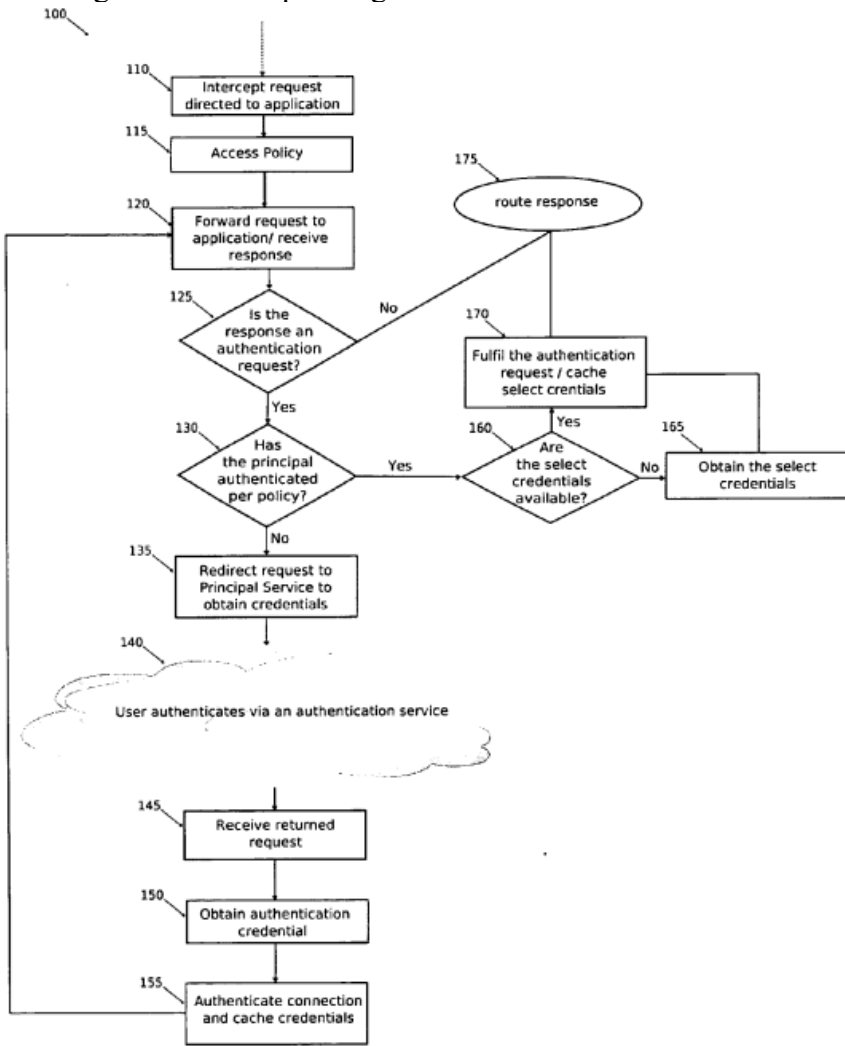
Burch at ¶45-46:

"After successfully authenticating the session of the first principal, at 155, the first principal's request may be reissued to the second principal back at the processing depicted in 120. Reissuing the original request may be done using a variety of different mechanisms. For example, in the Liberty and SAML context, the received request, at 145, may contain a URL tag indicating a resource that the first principal ultimately wishes to access after authentication; in the Liberty and SAML specifications this is referred to as the target URL. In such a scenario, the front-end service may parse the target URL from the request URL received, at 145, and reissue it to the second principal, at 120. Alternatively, the front-end service may cache the request URL before redirecting the first principal to the authentication service, at 135. The request is reissued to the second principal, at 120, and the response from the second principal is obtained. The front-end service may determine, at 125, whether

Claim	Exemplary Citation from Burch
	<p>the response from the second principal is an authentication request. Since the reissued request is the same that prompted an authentication request previously, the response, at 120, will likely be an authentication request. At 130, the front-end service determines whether the first principal has authenticated its identity, as required by the policy information. Since the first principal was redirected to the authentication service, at 140, and a credential authenticating the principal's identity was retrieved, at 150, this determining is positive."</p> <p>Burch at ¶ 50:  "Once the select credentials associated with the first principals are obtained, at 165, they may be cached for use in subsequent sessions or to authenticate the first principal to other principals associated with the front-end service. The select credentials may then be used to perform a proxy-authentication, as depicted at 170. Proxy-authentication includes authenticating the connection between the front-end service and the second principal with the identity of the first principal, thereby allowing the front-end service to make requests as a "proxy" of the first principal. This may be done in a variety of different manners. In an example, using the HTTP protocol, the "authorization header" of subsequent HTTP requests made on behalf of the first principal may be processed to include the select credentials required by the second principal."</p> <p>Burch at ¶ 51-52:  "Thus, as part of the message processing function associated with the front-end service, the front-end service may set the authorization header on each message received from the first principal as it is routed to the second principal. Under this scenario, the second principal validates the information provided in the HTTP authorization header and authenticates the connection of the front-end service as if it were the first principal. The front-end service may include the HTTP authorization header in subsequent requests from the first principal to the second principal for the duration of the first principal's session, as may also be stipulated by the policy information. From the perspective of the second principal, it appears that the first principal itself has directly authenticated using its own means of authentication, and the front-end service may route the reply messages from the second principal to the first principal, at 175. Thus, the first principal is granted seamless access to the second principal. And access to the second principal has been protected using a multifactor authentication mechanism, without altering the processing flow or processing aspects associated with the second principal. In other words, the multifactor authentication technique was implemented and</p>

Claim	Exemplary Citation from Burch
	<p>enforced with no changes or modifications to the second principal. Using this technique, a high level of security is achievable with a minimal impact on existing or new application services that have no native multifactor authentication functionality."</p> <p>Burch at ¶69:  "The redirection, at 270, may have directed the first principal to a resource provided by a second principal, as depicted at 275. In this case, the receiver of the request (a front-end service, such as the one presented with respect to the method 100 of the FIG. 1 that provides access control to the second principal) may identify the request as including authentication information. As such, the receiver may issue a request to obtain the authentication credential associated with the first principal, the request being received at 280. The request may be issued over a communication link secured via mutually authenticated SSL. The request, at 280, may include identifying information included in the redirection URL, at 270. Alternatively, if no authentication credential for the first principal was generated, at 240, but instead a connection or session based authentication scheme was used, the authentication service may generate a credential for the first principal when the request, at 275, is received. At 285 the authentication credential may be provided to the requester who may then verify the credential and authenticate the first principal, providing the first principal with access to the application (second principal or resources of the second principal)."</p> <p>Burch at ¶ 92:  "In an embodiment, user 405 (first principal) may wish to access an application hosted by application server 440 (second principal). The user 405 may attempt to access the application via a WWW browser using an HTTP protocol via communication channel 420. The firewall 410 may be configured to allow the incoming connection to reach front-end service 425 configured to receive, process, and route traffic intended for application servers 440. Firewall 415 may be configured to allow the front-end service 425 to use communication link 435 to access application servers 440. Upon receipt of the request, the front-end service 425 may determine that access to the application server 440 is predicated upon the proper authentication of user 405. Alternatively, the applications running in conjunction with front-end service 425 may detect an authentication request from application servers 440 is responsive to a request from user 405. In order to authenticate the identity of user 405, the front-end service 425 may cause the request from user 405 to be redirected over communications channel 455 to an authentication service or an identity service 460."</p>

Claim	Exemplary Citation from Burch
<p>[1E] the principal's access sessions occur indirectly through the identity service and transparently to the principal,</p>	<p>Burch discloses and/or renders obvious that the principal's access sessions occur indirectly through the identity service and transparently to the principal.</p>

Claim	Exemplary Citation from Burch
	<p data-bbox="598 240 1102 267">Burch at Fig. 1 and corresponding text:</p>  <pre data-bbox="651 267 1491 1307"> graph TD     100(( )) --&gt; 110[Intercept request directed to application]     110 --&gt; 115[Access Policy]     115 --&gt; 120[Forward request to application / receive response]     120 --&gt; 125{Is the response an authentication request?}     125 -- No --&gt; 175([route response])     125 -- Yes --&gt; 130{Has the principal authenticated per policy?}     130 -- Yes --&gt; 160{Are the select credentials available?}     130 -- No --&gt; 135[Redirect request to Principal Service to obtain credentials]     135 --&gt; 140     140 --- 145[Receive returned request]     145 --&gt; 150[Obtain authentication credential]     150 --&gt; 155[Authenticate connection and cache credentials]     155 --&gt; 120     160 -- Yes --&gt; 170[Fulfill the authentication request / cache select credentials]     160 -- No --&gt; 165[Obtain the select credentials]     165 --&gt; 170     170 --&gt; 175   </pre> <p data-bbox="1039 1331 1144 1356">Figure 1</p>

Claim

Exemplary Citation from Burch

Burch at Fig. 2 and corresponding text:

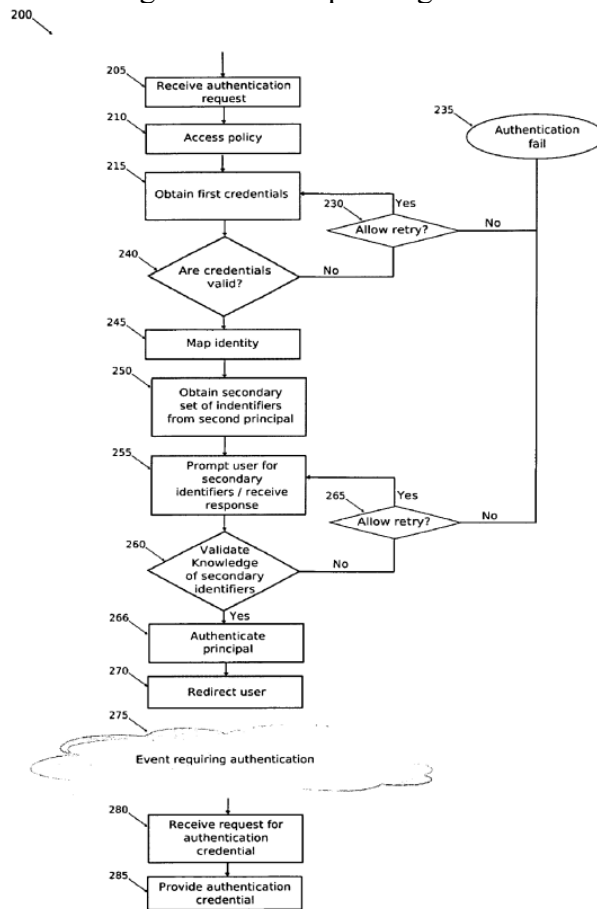


Figure 2

**Claim**

**Exemplary Citation from Burch**

Burch at Fig. 3 and corresponding text:

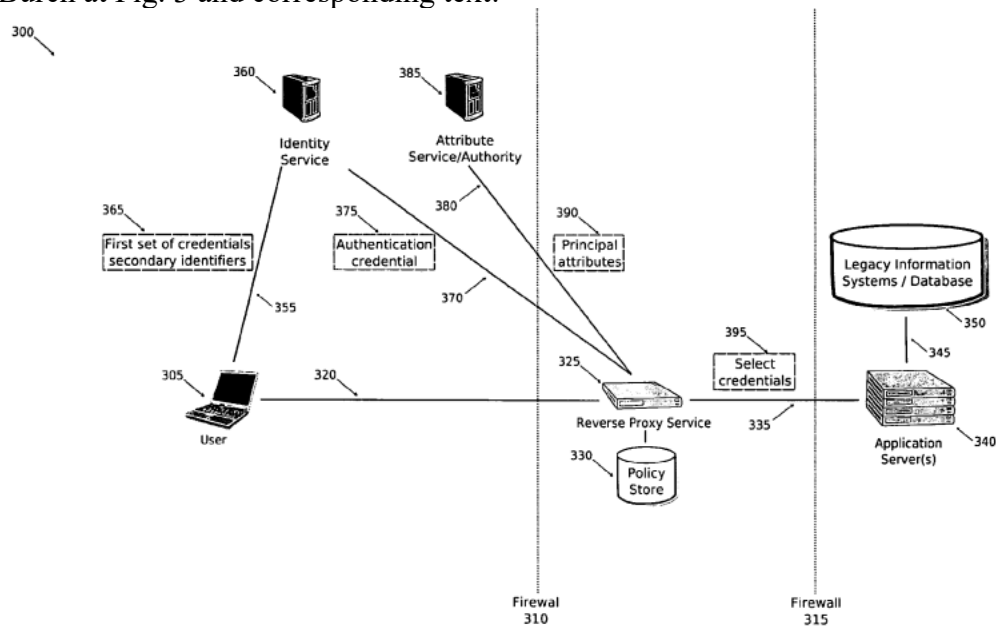


Figure 3

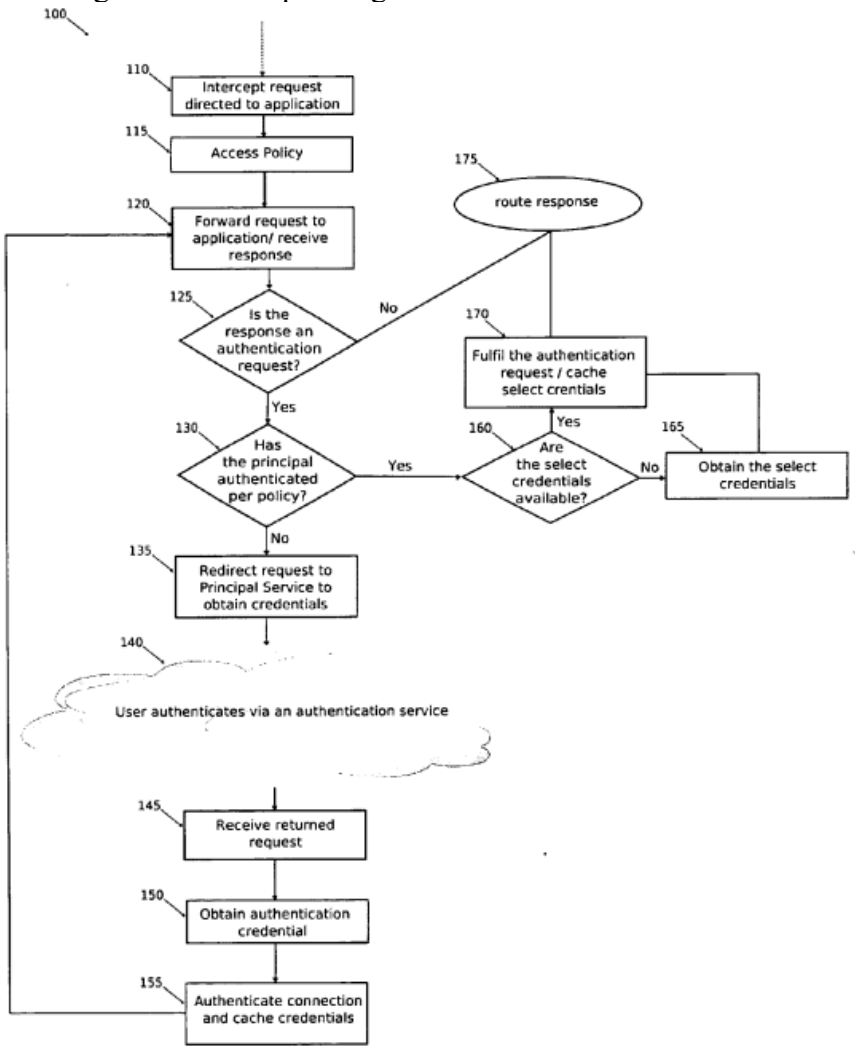
Burch at ¶45-46:

"After successfully authenticating the session of the first principal, at 155, the first principal's request may be reissued to the second principal back at the processing depicted in 120. Reissuing the original request may be done using a variety of different mechanisms. For example, in the Liberty and SAML context, the received request, at 145, may contain a URL tag indicating a resource that the first principal ultimately wishes to access after authentication; in the Liberty and SAML specifications this is referred to as the target URL. In such a scenario, the front-end service may parse the target URL from the request URL received, at 145, and reissue it to the second principal, at 120. Alternatively, the front-end service may cache the request URL before redirecting the first principal to the authentication service, at 135. The request is reissued to the second principal, at 120, and the response from the second principal is obtained. The front-end service may determine, at 125, whether

Claim	Exemplary Citation from Burch
	<p>the response from the second principal is an authentication request. Since the reissued request is the same that prompted an authentication request previously, the response, at 120, will likely be an authentication request. At 130, the front-end service determines whether the first principal has authenticated its identity, as required by the policy information. Since the first principal was redirected to the authentication service, at 140, and a credential authenticating the principal's identity was retrieved, at 150, this determining is positive."</p> <p>Burch at ¶ 50:  "Once the select credentials associated with the first principals are obtained, at 165, they may be cached for use in subsequent sessions or to authenticate the first principal to other principals associated with the front-end service. The select credentials may then be used to perform a proxy-authentication, as depicted at 170. Proxy-authentication includes authenticating the connection between the front-end service and the second principal with the identity of the first principal, thereby allowing the front-end service to make requests as a "proxy" of the first principal. This may be done in a variety of different manners. In an example, using the HTTP protocol, the "authorization header" of subsequent HTTP requests made on behalf of the first principal may be processed to include the select credentials required by the second principal."</p> <p>Burch at ¶ 51-52:  "Thus, as part of the message processing function associated with the front-end service, the front-end service may set the authorization header on each message received from the first principal as it is routed to the second principal. Under this scenario, the second principal validates the information provided in the HTTP authorization header and authenticates the connection of the front-end service as if it were the first principal. The front-end service may include the HTTP authorization header in subsequent requests from the first principal to the second principal for the duration of the first principal's session, as may also be stipulated by the policy information. From the perspective of the second principal, it appears that the first principal itself has directly authenticated using its own means of authentication, and the front-end service may route the reply messages from the second principal to the first principal, at 175. Thus, the first principal is granted seamless access to the second principal. And access to the second principal has been protected using a multifactor authentication mechanism, without altering the processing flow or processing aspects associated with the second principal. In other words, the multifactor authentication technique was implemented and</p>

Claim	Exemplary Citation from Burch
	<p>enforced with no changes or modifications to the second principal. Using this technique, a high level of security is achievable with a minimal impact on existing or new application services that have no native multifactor authentication functionality."</p> <p>Burch at ¶69:  "The redirection, at 270, may have directed the first principal to a resource provided by a second principal, as depicted at 275. In this case, the receiver of the request (a front-end service, such as the one presented with respect to the method 100 of the FIG. 1 that provides access control to the second principal) may identify the request as including authentication information. As such, the receiver may issue a request to obtain the authentication credential associated with the first principal, the request being received at 280. The request may be issued over a communication link secured via mutually authenticated SSL. The request, at 280, may include identifying information included in the redirection URL, at 270. Alternatively, if no authentication credential for the first principal was generated, at 240, but instead a connection or session based authentication scheme was used, the authentication service may generate a credential for the first principal when the request, at 275, is received. At 285 the authentication credential may be provided to the requester who may then verify the credential and authenticate the first principal, providing the first principal with access to the application (second principal or resources of the second principal)."</p> <p>Burch at ¶ 92:  "In an embodiment, user 405 (first principal) may wish to access an application hosted by application server 440 (second principal). The user 405 may attempt to access the application via a WWW browser using an HTTP protocol via communication channel 420. The firewall 410 may be configured to allow the incoming connection to reach front-end service 425 configured to receive, process, and route traffic intended for application servers 440. Firewall 415 may be configured to allow the front-end service 425 to use communication link 435 to access application servers 440. Upon receipt of the request, the front-end service 425 may determine that access to the application server 440 is predicated upon the proper authentication of user 405. Alternatively, the applications running in conjunction with front-end service 425 may detect an authentication request from application servers 440 is responsive to a request from user 405. In order to authenticate the identity of user 405, the front-end service 425 may cause the request from user 405 to be redirected over communications channel 455 to an authentication service or an identity service 460."</p>

Claim	Exemplary Citation from Burch
<p>[1F] wherein the authentication message includes the new authentication request made on behalf of the principal and the authentication message also includes a new authentication response that satisfies the new authentication request,</p>	<p>Burch discloses and/or renders obvious that the authentication message includes the new authentication request made on behalf of the principal and the authentication message also includes a new authentication response that satisfies the new authentication request.</p>

Claim	Exemplary Citation from Burch
	<p data-bbox="598 240 1102 267">Burch at Fig. 1 and corresponding text:</p>  <pre data-bbox="651 267 1501 1307"> graph TD     100(( )) --&gt; 110[Intercept request directed to application]     110 --&gt; 115[Access Policy]     115 --&gt; 120[Forward request to application / receive response]     120 --&gt; 125{Is the response an authentication request?}     125 -- No --&gt; 175([route response])     125 -- Yes --&gt; 130{Has the principal authenticated per policy?}     130 -- Yes --&gt; 160{Are the select credentials available?}     130 -- No --&gt; 135[Redirect request to Principal Service to obtain credentials]     135 --&gt; 140     140 --- 145[Receive returned request]     145 --&gt; 150[Obtain authentication credential]     150 --&gt; 155[Authenticate connection and cache credentials]     155 --&gt; 120     160 -- Yes --&gt; 170[Fulfill the authentication request / cache select credentials]     160 -- No --&gt; 165[Obtain the select credentials]     165 --&gt; 170     170 --&gt; 175   </pre> <p data-bbox="1039 1331 1144 1356">Figure 1</p>

**Claim**

**Exemplary Citation from Burch**

Burch at Fig. 2 and corresponding text:

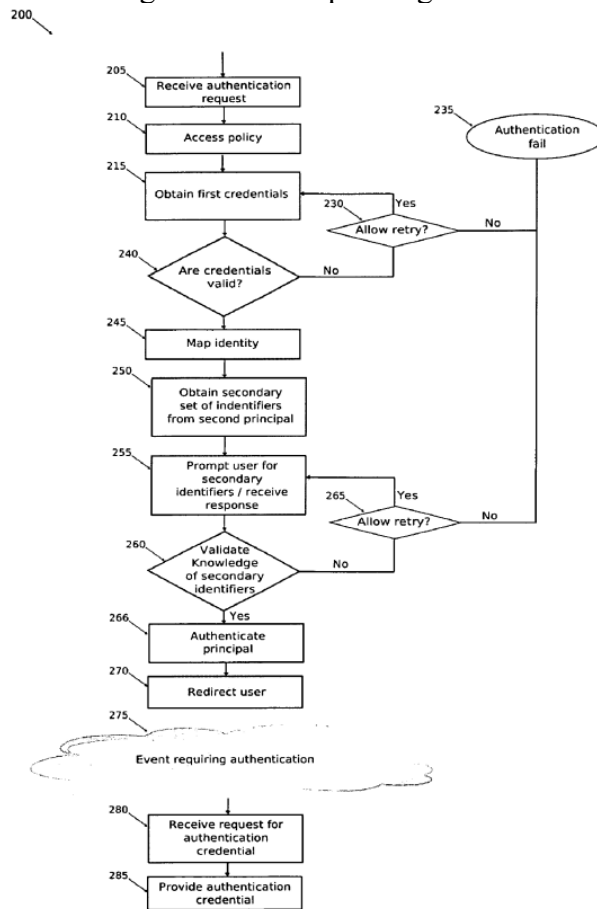


Figure 2

**Claim**

**Exemplary Citation from Burch**

Burch at Fig. 3 and corresponding text:

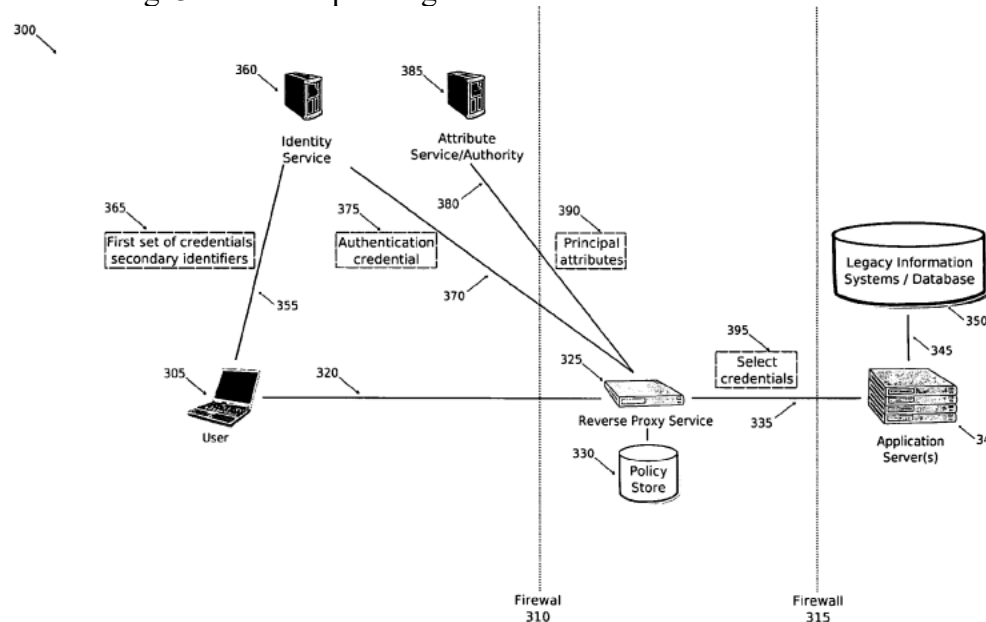


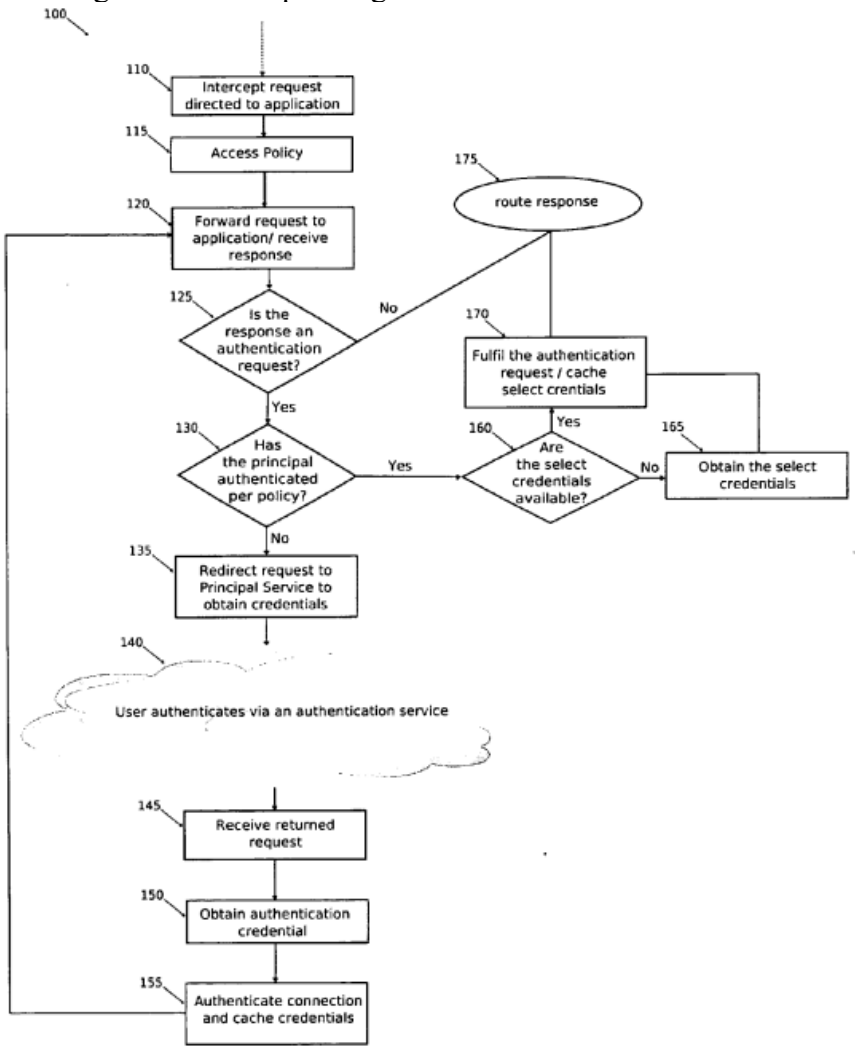
Figure 3

Burch at ¶45-46:

"After successfully authenticating the session of the first principal, at 155, the first principal's request may be reissued to the second principal back at the processing depicted in 120. Reissuing the original request may be done using a variety of different mechanisms. For example, in the Liberty and SAML context, the received request, at 145, may contain a URL tag indicating a resource that the first principal ultimately wishes to access after authentication; in the Liberty and SAML specifications this is referred to as the target URL. In such a scenario, the front-end service may parse the target URL from the request URL received, at 145, and reissue it to the second principal, at 120. Alternatively, the front-end service may cache the request URL before redirecting the first principal to the authentication service, at 135. The request is reissued to the second principal, at 120, and the response from the second principal is obtained. The front-end service may determine, at 125, whether

Claim	Exemplary Citation from Burch
	<p>the response from the second principal is an authentication request. Since the reissued request is the same that prompted an authentication request previously, the response, at 120, will likely be an authentication request. At 130, the front-end service determines whether the first principal has authenticated its identity, as required by the policy information. Since the first principal was redirected to the authentication service, at 140, and a credential authenticating the principal's identity was retrieved, at 150, this determining is positive."</p> <p>Burch at ¶ 51-52:  "Thus, as part of the message processing function associated with the front-end service, the front-end service may set the authorization header on each message received from the first principal as it is routed to the second principal. Under this scenario, the second principal validates the information provided in the HTTP authorization header and authenticates the connection of the front-end service as if it were the first principal. The front-end service may include the HTTP authorization header in subsequent requests from the first principal to the second principal for the duration of the first principal's session, as may also be stipulated by the policy information. From the perspective of the second principal, it appears that the first principal itself has directly authenticated using its own means of authentication, and the front-end service may route the reply messages from the second principal to the first principal, at 175. Thus, the first principal is granted seamless access to the second principal. And access to the second principal has been protected using a multifactor authentication mechanism, without altering the processing flow or processing aspects associated with the second principal. In other words, the multifactor authentication technique was implemented and enforced with no changes or modifications to the second principal. Using this technique, a high level of security is achievable with a minimal impact on existing or new application services that have no native multifactor authentication functionality."</p> <p>Burch at ¶69:  "The redirection, at 270, may have directed the first principal to a resource provided by a second principal, as depicted at 275. In this case, the receiver of the request (a front-end service, such as the one presented with respect to the method 100 of the FIG. 1 that provides access control to the second principal) may identify the request as including authentication information. As such, the receiver may issue a request to obtain the authentication credential associated with the first principal, the request being received at 280. The request may be issued over a communication link secured via mutually</p>

Claim	Exemplary Citation from Burch
	<p>authenticated SSL. The request, at 280, may include identifying information included in the redirection URL, at 270. Alternatively, if no authentication credential for the first principal was generated, at 240, but instead a connection or session based authentication scheme was used, the authentication service may generate a credential for the first principal when the request, at 275, is received. At 285 the authentication credential may be provided to the requester who may then verify the credential and authenticate the first principal, providing the first principal with access to the application (second principal or resources of the second principal)."</p> <p>Burch at ¶ 100-101:  "After obtaining the one or more secondary identifiers 497, user 405 may be prompted to verify its knowledge of the identifiers 497 via communications link 455. The communications link 455 may be secured via SSL to prevent disclosure of the credentials. The user 405 may respond to the request for the identifiers 497 via the link 455. Upon receipt of the identifiers 497 the authentication service 460 may verify the user's 405 knowledge of the identifiers 497. If successful, the user 405 may be expected, depending upon the policy information 465, to respond to additional authentication queries regarding the secondary identifiers 497. Further, upon verification of a first set of secondary identifiers 470, the authentication service 460 may be required by the policy 465, to obtain and verify the user's 405 knowledge of additional secondary identifiers 497. Once user 405 has authenticated in accordance with the policy 465, the authentication service 460 may generate an authentication credential 480 for the user 405. The credential 480 may be used to authenticate the identity of user 405 to other sites and/or principals 440. User 405 may then be redirected back to the application server 440. The URL used to redirect user 405 may include a token identifying the authentication service 460 and authenticated credential 480 generated for user 405. The token may be encoded into the URL string in accordance with the Liberty or SAML specifications. Alternatively, the authentication credential 480 may be encoded into the request itself. For instance, if the authentication service 460 were acting as a SAML Authentication Authority performing the SAML POST profile, the credential 480 transported via the as the payload of an HTTP POST to the application server 440."</p>
<p>[1G] that response vouches for authentication of the principal to the identity service for the single sign-</p>	<p>Burch discloses and/or renders obvious that that response vouches for authentication of the principal to the identity service for the single sign-on access of the principal, the principal believing interactions are with the external service, which is one of the other services that the identity service controls access to.</p>

Claim	Exemplary Citation from Burch
<p>on access of the principal, the principal believing interactions are with the external service, which is one of the other services that the identity service controls access to,</p>	<p>Burch at Fig. 1 and corresponding text:</p>  <pre> graph TD     100(( )) --&gt; 110[Intercept request directed to application]     110 --&gt; 115[Access Policy]     115 --&gt; 120[Forward request to application / receive response]     120 --&gt; 125{Is the response an authentication request?}     125 -- No --&gt; 175([route response])     125 -- Yes --&gt; 130{Has the principal authenticated per policy?}     130 -- Yes --&gt; 160{Are the select credentials available?}     130 -- No --&gt; 135[Redirect request to Principal Service to obtain credentials]     135 --&gt; 140     140 --- 145[Receive returned request]     145 --&gt; 150[Obtain authentication credential]     150 --&gt; 155[Authenticate connection and cache credentials]     155 --&gt; 120     160 -- Yes --&gt; 170[Fulfill the authentication request / cache select credentials]     160 -- No --&gt; 165[Obtain the select credentials]     165 --&gt; 170     170 --&gt; 175   </pre> <p>Figure 1</p>

**Claim**

**Exemplary Citation from Burch**

Burch at Fig. 2 and corresponding text:

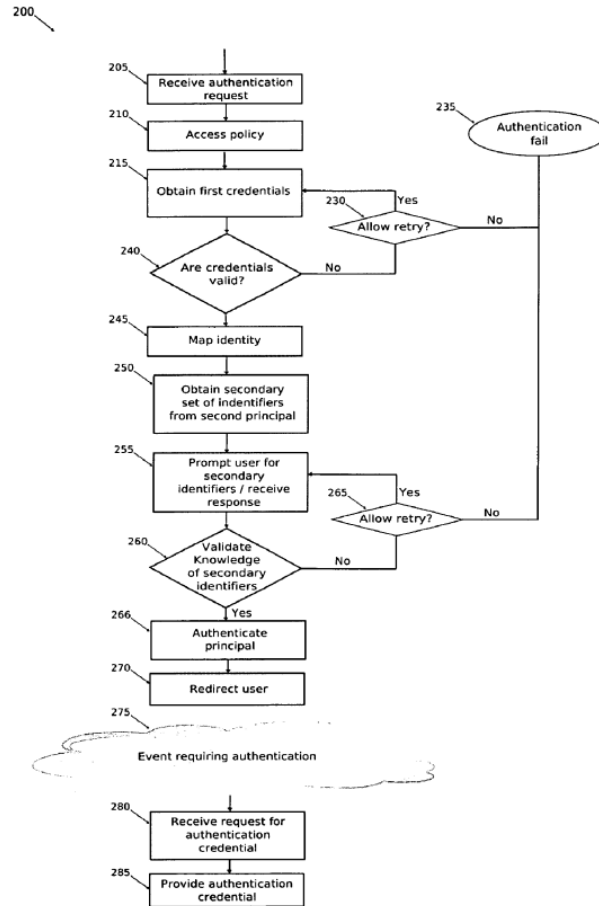


Figure 2

**Claim**

**Exemplary Citation from Burch**

Burch at Fig. 3 and corresponding text:

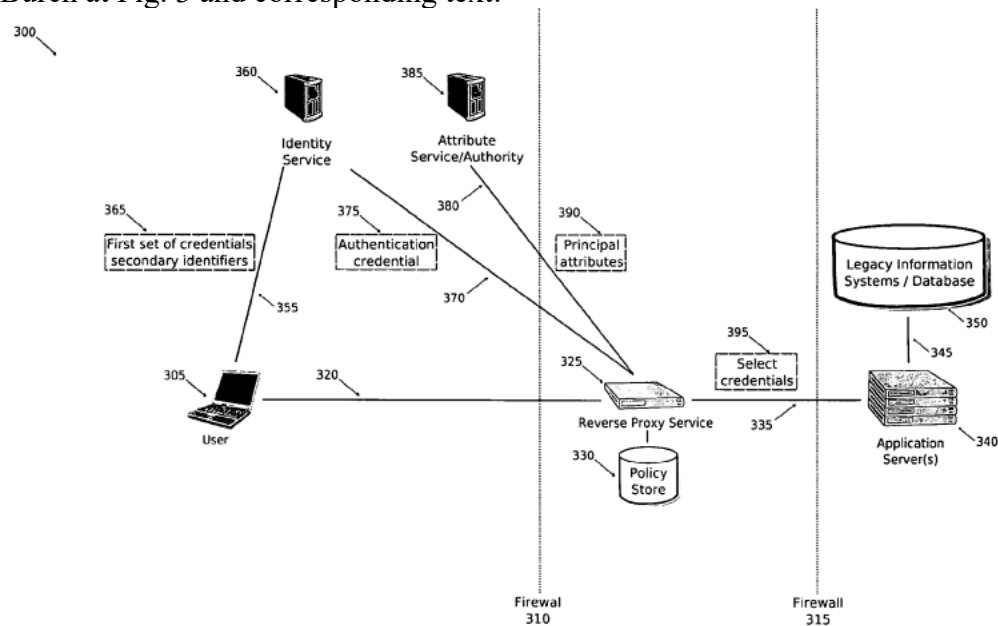


Figure 3

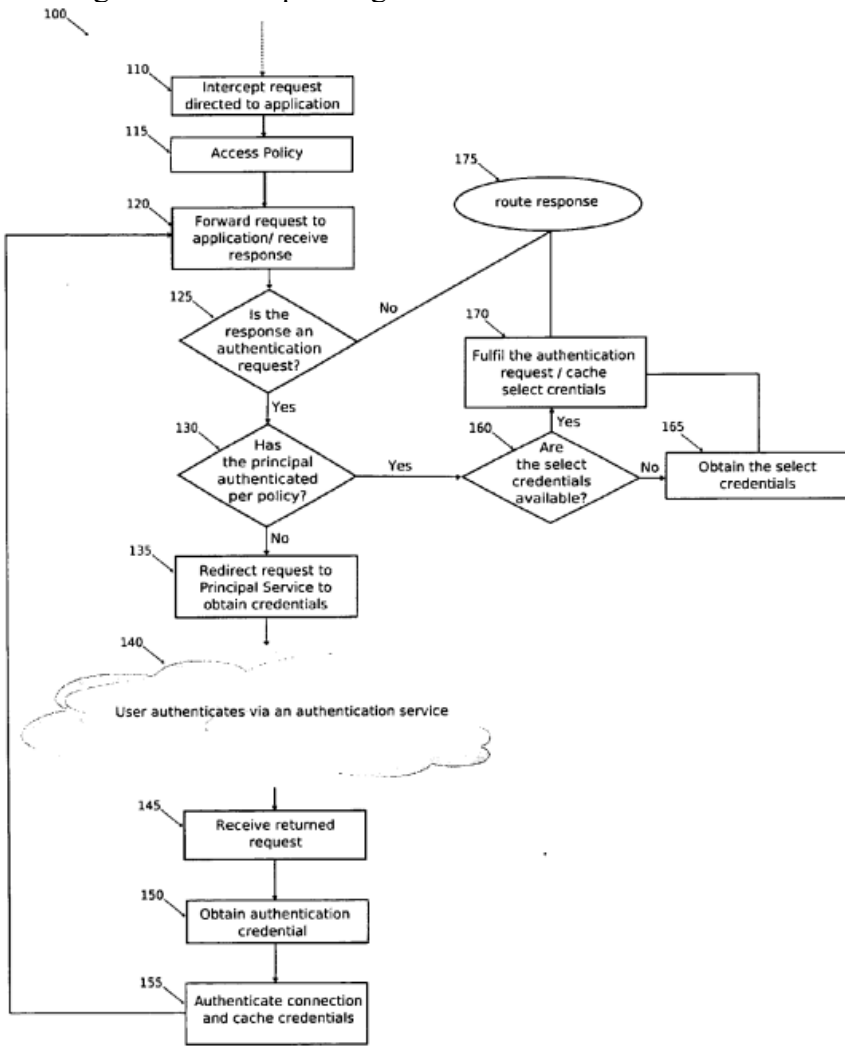
Burch at ¶45-46:

"After successfully authenticating the session of the first principal, at 155, the first principal's request may be reissued to the second principal back at the processing depicted in 120. Reissuing the original request may be done using a variety of different mechanisms. For example, in the Liberty and SAML context, the received request, at 145, may contain a URL tag indicating a resource that the first principal ultimately wishes to access after authentication; in the Liberty and SAML specifications this is referred to as the target URL. In such a scenario, the front-end service may parse the target URL from the request URL received, at 145, and reissue it to the second principal, at 120. Alternatively, the front-end service may cache the request URL before redirecting the first principal to the authentication service, at 135. The request is reissued to the second principal, at 120, and the response from the second principal is obtained. The front-end service may determine, at 125, whether

Claim	Exemplary Citation from Burch
	<p>the response from the second principal is an authentication request. Since the reissued request is the same that prompted an authentication request previously, the response, at 120, will likely be an authentication request. At 130, the front-end service determines whether the first principal has authenticated its identity, as required by the policy information. Since the first principal was redirected to the authentication service, at 140, and a credential authenticating the principal's identity was retrieved, at 150, this determining is positive."</p> <p>Burch at ¶ 51-52:  "Thus, as part of the message processing function associated with the front-end service, the front-end service may set the authorization header on each message received from the first principal as it is routed to the second principal. Under this scenario, the second principal validates the information provided in the HTTP authorization header and authenticates the connection of the front-end service as if it were the first principal. The front-end service may include the HTTP authorization header in subsequent requests from the first principal to the second principal for the duration of the first principal's session, as may also be stipulated by the policy information. From the perspective of the second principal, it appears that the first principal itself has directly authenticated using its own means of authentication, and the front-end service may route the reply messages from the second principal to the first principal, at 175. Thus, the first principal is granted seamless access to the second principal. And access to the second principal has been protected using a multifactor authentication mechanism, without altering the processing flow or processing aspects associated with the second principal. In other words, the multifactor authentication technique was implemented and enforced with no changes or modifications to the second principal. Using this technique, a high level of security is achievable with a minimal impact on existing or new application services that have no native multifactor authentication functionality."</p> <p>Burch at ¶69:  "The redirection, at 270, may have directed the first principal to a resource provided by a second principal, as depicted at 275. In this case, the receiver of the request (a front-end service, such as the one presented with respect to the method 100 of the FIG. 1 that provides access control to the second principal) may identify the request as including authentication information. As such, the receiver may issue a request to obtain the authentication credential associated with the first principal, the request being received at 280. The request may be issued over a communication link secured via mutually</p>

Claim	Exemplary Citation from Burch
	<p>authenticated SSL. The request, at 280, may include identifying information included in the redirection URL, at 270. Alternatively, if no authentication credential for the first principal was generated, at 240, but instead a connection or session based authentication scheme was used, the authentication service may generate a credential for the first principal when the request, at 275, is received. At 285 the authentication credential may be provided to the requester who may then verify the credential and authenticate the first principal, providing the first principal with access to the application (second principal or resources of the second principal)."</p> <p>Burch at ¶79:  "Upon receipt of the redirected request, the reverse proxy 325 may identify authentication information embedded within it. This information could include a token identifying an authentication credential 375 encoded within the URL as in the Liberty and SAML specifications. Alternatively, the request itself may include an authentication credential 375 as an HTTP POST parameter. There are a number of ways authentication information may be included in the redirected request. The embodiments of the invention should not be read as limited to any particular technique."</p> <p>Burch at ¶ 83:  "Before issuing the target URL request to the application servers 340 , the reverse proxy server 325 may access the policy store 330 . The policy information 330 may indicate that the application servers 340 themselves require select credentials 395 in order to authenticate the user 305 . This situation could arise if the application servers 340 originally included a proprietary authentication system. Thus, rather than modifying application server code, the reverse proxy 325 may be configured to provide the select credentials 395 that the application servers 340 expect via 335 . Similarly, the application servers 340 may themselves need to access a legacy information system (IS) or database 350 via link 345 in order to provide the services requested by user 305 . As such, the application servers 340 may need the select credentials 395 in order to access data specific to user 305 as required by the application. Given the disparate uses of the credentials 395 by the application servers 340 (some of which may not be strictly security related), the select credentials 395 may be completely distinct from those employed by user 305 to authenticate its identity to identity service 360 ."</p>

Claim	Exemplary Citation from Burch
	<p>Burch at ¶ 100-101:            "After obtaining the one or more secondary identifiers 497, user 405 may be prompted to verify its knowledge of the identifiers 497 via communications link 455. The communications link 455 may be secured via SSL to prevent disclosure of the credentials. The user 405 may respond to the request for the identifiers 497 via the link 455. Upon receipt of the identifiers 497 the authentication service 460 may verify the user's 405 knowledge of the identifiers 497. If successful, the user 405 may be expected, depending upon the policy information 465, to respond to additional authentication queries regarding the secondary identifiers 497. Further, upon verification of a first set of secondary identifiers 470, the authentication service 460 may be required by the policy 465, to obtain and verify the user's 405 knowledge of additional secondary identifiers 497. Once user 405 has authenticated in accordance with the policy 465, the authentication service 460 may generate an authentication credential 480 for the user 405. The credential 480 may be used to authenticate the identity of user 405 to other sites and/or principals 440. User 405 may then be redirected back to the application server 440. The URL used to redirect user 405 may include a token identifying the authentication service 460 and authenticated credential 480 generated for user 405. The token may be encoded into the URL string in accordance with the Liberty or SAML specifications. Alternatively, the authentication credential 480 may be encoded into the request itself. For instance, if the authentication service 460 were acting as a SAML Authentication Authority performing the SAML POST profile, the credential 480 transported via the as the payload of an HTTP POST to the application server 440."</p>
<p>[1H] and a determination as to whether to use a single interaction or multiple interactions for authentication of the principal to the other services is automatically communicated in the new authentication response.</p>	<p>Burch discloses and/or renders obvious a determination as to whether to use a single interaction or multiple interactions for authentication of the principal to the other services is automatically communicated in the new authentication response.</p>

Claim	Exemplary Citation from Burch
	<p data-bbox="598 240 1102 267">Burch at Fig. 1 and corresponding text:</p>  <pre data-bbox="651 267 1491 1307"> graph TD     100(( )) --&gt; 110[Intercept request directed to application]     110 --&gt; 115[Access Policy]     115 --&gt; 120[Forward request to application / receive response]     120 --&gt; 125{Is the response an authentication request?}     125 -- No --&gt; 175([route response])     125 -- Yes --&gt; 130{Has the principal authenticated per policy?}     130 -- Yes --&gt; 160{Are the select credentials available?}     130 -- No --&gt; 135[Redirect request to Principal Service to obtain credentials]     135 --&gt; 140     140 --- 145[Receive returned request]     145 --&gt; 150[Obtain authentication credential]     150 --&gt; 155[Authenticate connection and cache credentials]     155 --&gt; 120     160 -- Yes --&gt; 170[Fulfill the authentication request / cache select credentials]     160 -- No --&gt; 165[Obtain the select credentials]     165 --&gt; 170     170 --&gt; 175 </pre> <p data-bbox="1039 1331 1144 1356">Figure 1</p>

**Claim**

**Exemplary Citation from Burch**

Burch at Fig. 2 and corresponding text:

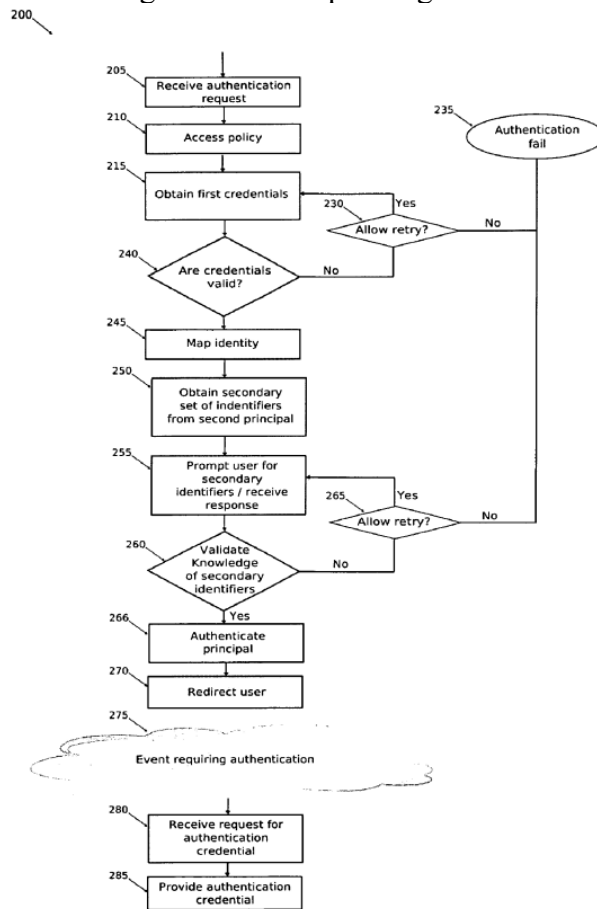


Figure 2

**Claim**

**Exemplary Citation from Burch**

Burch at Fig. 3 and corresponding text:

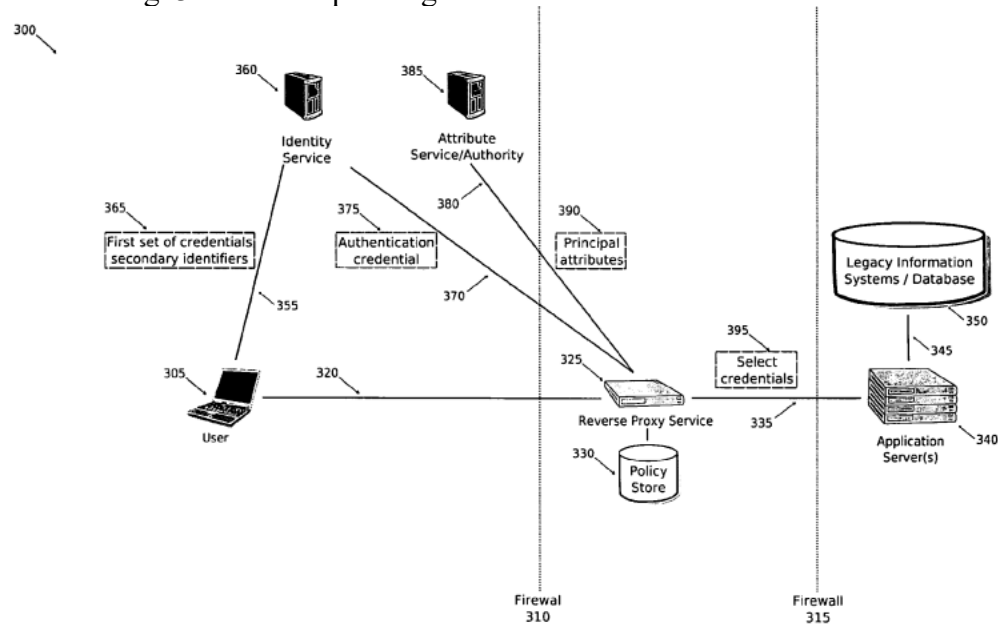


Figure 3

Burch at ¶ 47-49:

"At 160, the front-end service determines whether the select credentials, required by the second principal to authenticate the identity of the first principal, are available. This processing may be used because the select credentials requested by the second principal may be completely distinct from those used at the authentication service discussed in connection with the processing at 140, 150, or 155. This situation may arise if developers of the application, associated with the second principal, desired to increase security, but determined that updates to the legacy authentication service of the second principal would too expensive and/or time consuming to be practical. In such a situation, multifactor authentication may take place at the authentication service while the legacy authentication scheme of the second principal may remain unchanged. Similarly, in developing a new application, developers may determine that it may be more cost effective to leverage an existing or

Claim	Exemplary Citation from Burch
	<p>standardized authentication service; rather than to develop or buy a completely new proprietary security system or service. In such cases, the authentication service itself may be very simple (i.e., just enough to make the association between a first principal and an identity on the system). But, by enforcing a security policy, at 115, and by utilizing the authentication service, at 140, the second principal may be protected using a multifactor authentication mechanism. This is so, because no first principal may access the second principal directly; instead access is routed through the front-end service where the front-end service requires that the first principals authenticate using mechanisms or techniques that comply with the second principal's policy, which identifies an additional multifactor authentication technique. By way of example, the second principal may be an application server of an on-line banking system or service. In this case, the authentication service of the second principal may only require an account number and PIN to obtain access to the on-line banking service. However, in order to increase security, a policy may stipulate that the first principal is to also authenticate using a sophisticated multifactor authentication mechanism, depicted at 140. The multifactor authentication mechanism may include a password and, perhaps, the principal's last bank transaction (or any other additional authentication criterion or criteria). As such, it may be common and even desirable, that there be a mismatch between the authentication credentials required for authentication to the authentication service from those that are sent to the second principal, at 170. However, in order for this scenario to function seamlessly, the front-end service provides the second principal with the credentials it expects. Yet, the first principal never gains true access unabated until the additional credentials are also provided (last bank transaction). If the check, at 160, indicates that the select credentials required by the second principal are not available, the select credentials have be obtained, at 165. There are a number of ways the select credentials may be obtained. For example, a Liberty or SAML attribute request could be issued to a Liberty or SAML attribute authority associated with the first principal. As per the proceeding example, an attribute request, at 165, may be issued for the first principal's account number and PIN; the attribute request including information, which allows the attribute authority to verify that the request was made from a trusted entity. Responsive to this request, the attribute authority may return an attribute assertion containing the required select credentials. The communication of the request and response may be secured using mutually authenticated secure socket layer (SSL) transport, IPSec, or any other secure communications protocol. The contents of the attribute authority may be authenticated using PKI tools or other message authentication mechanisms. Alternatively, the policy accessed, at 125, may have specified the select credentials used for authentication to the second principal. Thus, the authentication service</p>

Claim	Exemplary Citation from Burch
	<p>may have provided the credentials as part of the authentication credential message received at 150 and cached at 155."</p> <p>Burch at ¶ 51-52:  "Thus, as part of the message processing function associated with the front-end service, the front-end service may set the authorization header on each message received from the first principal as it is routed to the second principal. Under this scenario, the second principal validates the information provided in the HTTP authorization header and authenticates the connection of the front-end service as if it were the first principal. The front-end service may include the HTTP authorization header in subsequent requests from the first principal to the second principal for the duration of the first principal's session, as may also be stipulated by the policy information. From the perspective of the second principal, it appears that the first principal itself has directly authenticated using its own means of authentication, and the front-end service may route the reply messages from the second principal to the first principal, at 175. Thus, the first principal is granted seamless access to the second principal. And access to the second principal has been protected using a multifactor authentication mechanism, without altering the processing flow or processing aspects associated with the second principal. In other words, the multifactor authentication technique was implemented and enforced with no changes or modifications to the second principal. Using this technique, a high level of security is achievable with a minimal impact on existing or new application services that have no native multifactor authentication functionality."</p> <p>Burch at ¶79:  "Upon receipt of the redirected request, the reverse proxy 325 may identify authentication information embedded within it. This information could include a token identifying an authentication credential 375 encoded within the URL as in the Liberty and SAML specifications. Alternatively, the request itself may include an authentication credential 375 as an HTTP POST parameter. There are a number of ways authentication information may be included in the redirected request. The embodiments of the invention should not be read as limited to any particular technique."</p> <p>Burch at ¶ 83:  "Before issuing the target URL request to the application servers 340 , the reverse proxy server 325 may access the policy store 330 . The policy information 330 may indicate that the application</p>

Claim	Exemplary Citation from Burch
	<p>servers 340 themselves require select credentials 395 in order to authenticate the user 305 . This situation could arise if the application servers 340 originally included a proprietary authentication system. Thus, rather than modifying application server code, the reverse proxy 325 may be configured to provide the select credentials 395 that the application servers 340 expect via 335 . Similarly, the application servers 340 may themselves need to access a legacy information system (IS) or database 350 via link 345 in order to provide the services requested by user 305 . As such, the application servers 340 may need the select credentials 395 in order to access data specific to user 305 as required by the application. Given the disparate uses of the credentials 395 by the application servers 340 (some of which may not be strictly security related), the select credentials 395 may be completely distinct from those employed by user 305 to authenticate its identity to identity service 360 ."</p> <p>Burch at ¶ 93-94:  "The authentication service 460 may implemented as the authentication service discussed with respect to the method 200 in the FIG. 2 and/or the identity service 360 discussed with respect to the multifactor authentication system 300 of the FIG. 3. The authentication service 460 may support identity federation protocols and techniques such as those defined in the Liberty and SAML specifications. As such, the authentication service 460 may act as a Liberty Identity Service or Service Provider, SAML Authentication Authority, or the like. Similarly, the authentication service 460 may implement a proprietary or custom authentication protocol. Thus, the embodiments of this invention should not be read as limited to just the disclosed authentication systems and/or protocols. Upon receipt of the redirected authentication request, authentication service 460 may access policy store 465. The policy information 465 may indicate the particular authentication mechanisms that user 405 is to perform in order to authenticate its identity. In particular, the policy information 465 may specify a first set of credentials 470 and one or more secondary identifiers 497 that are to be produced by user 405 in order to authenticate its identity. In addition to specifying the one or more secondary identifiers 497, the policy information 465 may also indicate how these identifiers 497 may be obtained. After accessing policy information 465, the authentication service 460 may obtain from user 405 a first set of credentials 470. As discussed above, the first set of credentials 470 may include any number of different authentication materials, including, but not limited to, a username and password combination, a Smartcard, a token, PKI signature data, biometric information, etc. Upon receipt of this first set of credentials 470, the authentication service 460 may validate the</p>

Claim	Exemplary Citation from Burch
	<p>received credentials 470. A number of different verification mechanisms may be employed, as discussed above, such mechanisms may include: a hash based comparison for password based authentication, PKI signature verification, biometric methods, etc."</p> <p>Burch at ¶ 100-101:  "After obtaining the one or more secondary identifiers 497, user 405 may be prompted to verify its knowledge of the identifiers 497 via communications link 455. The communications link 455 may be secured via SSL to prevent disclosure of the credentials. The user 405 may respond to the request for the identifiers 497 via the link 455. Upon receipt of the identifiers 497 the authentication service 460 may verify the user's 405 knowledge of the identifiers 497. If successful, the user 405 may be expected, depending upon the policy information 465, to respond to additional authentication queries regarding the secondary identifiers 497. Further, upon verification of a first set of secondary identifiers 470, the authentication service 460 may be required by the policy 465, to obtain and verify the user's 405 knowledge of additional secondary identifiers 497. Once user 405 has authenticated in accordance with the policy 465, the authentication service 460 may generate an authentication credential 480 for the user 405. The credential 480 may be used to authenticate the identity of user 405 to other sites and/or principals 440. User 405 may then be redirected back to the application server 440. The URL used to redirect user 405 may include a token identifying the authentication service 460 and authenticated credential 480 generated for user 405. The token may be encoded into the URL string in accordance with the Liberty or SAML specifications. Alternatively, the authentication credential 480 may be encoded into the request itself. For instance, if the authentication service 460 were acting as a SAML Authentication Authority performing the SAML POST profile, the credential 480 transported via the as the payload of an HTTP POST to the application server 440."</p>
<p>[2] The method of claim 1 further comprising, making, by the machine, a target service available to interactions between the principal and an external service, the target service is directly accessible from an</p>	<p>Burch discloses and/or renders obvious making, by the machine, a target service available to interactions between the principal and an external service, the target service is directly accessible from an environment of the identity service.</p>

Claim	Exemplary Citation from Burch
<p>environment of the identity service.</p>	<p>Burch at Fig. 1 and corresponding text:</p> <pre> graph TD     100(( )) --&gt; 110[Intercept request directed to application]     110 --&gt; 115[Access Policy]     115 --&gt; 120[Forward request to application / receive response]     120 --&gt; 125{Is the response an authentication request?}     125 -- No --&gt; 175([route response])     125 -- Yes --&gt; 130{Has the principal authenticated per policy?}     130 -- Yes --&gt; 160{Are the select credentials available?}     130 -- No --&gt; 135[Redirect request to Principal Service to obtain credentials]     135 --&gt; 140     140 --- 145[Receive returned request]     145 --&gt; 150[Obtain authentication credential]     150 --&gt; 155[Authenticate connection and cache credentials]     155 --&gt; 120     160 -- Yes --&gt; 170[Fulfill the authentication request / cache select credentials]     160 -- No --&gt; 165[Obtain the select credentials]     165 --&gt; 170     170 --&gt; 175   </pre> <p style="text-align: center;">Figure 1</p>

Claim

Exemplary Citation from Burch

Burch at Fig. 2 and corresponding text:

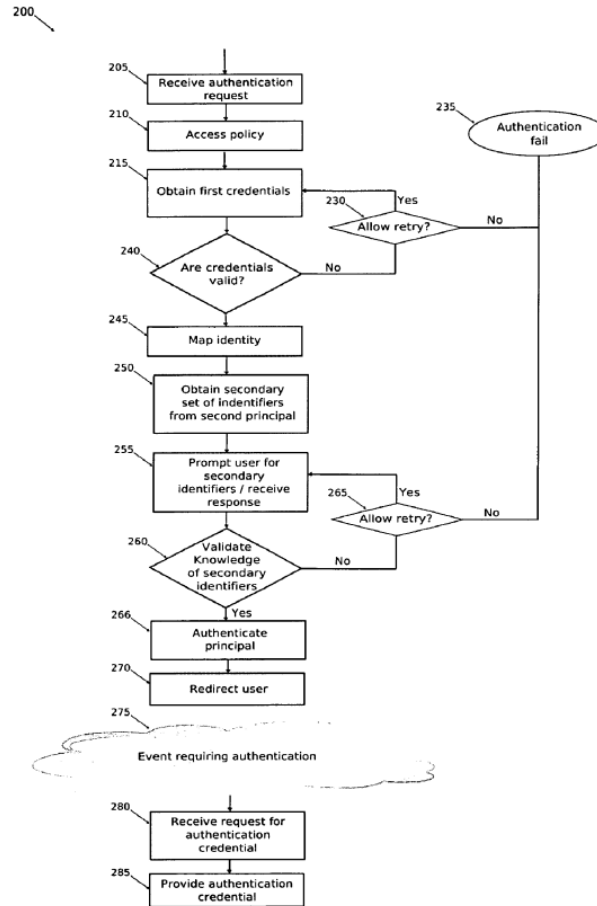


Figure 2

**Claim**

**Exemplary Citation from Burch**

Burch at Fig. 3 and corresponding text:

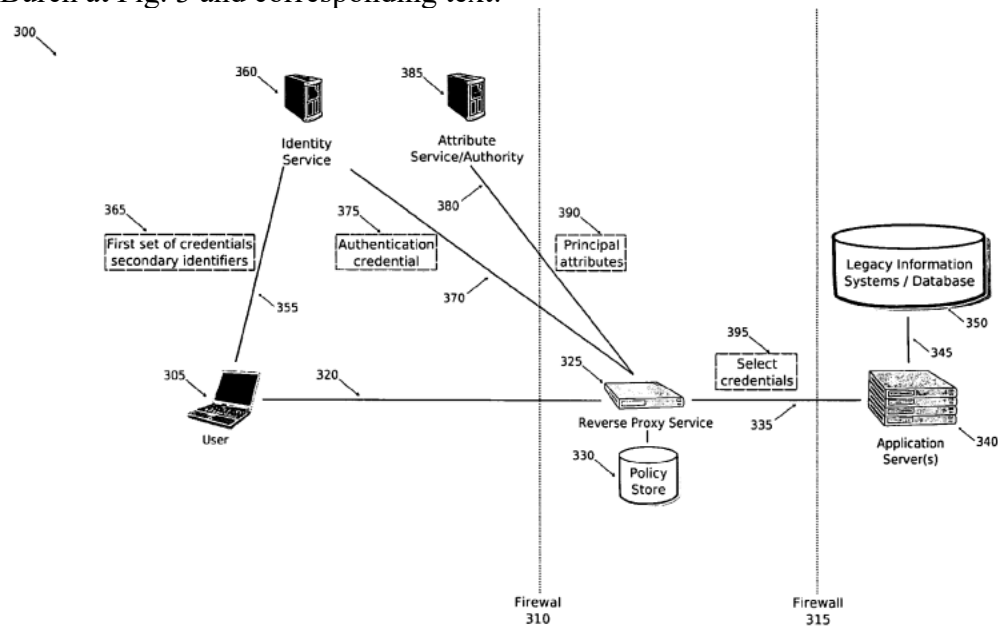


Figure 3

Burch at ¶45-46:

"After successfully authenticating the session of the first principal, at 155, the first principal's request may be reissued to the second principal back at the processing depicted in 120. Reissuing the original request may be done using a variety of different mechanisms. For example, in the Liberty and SAML context, the received request, at 145, may contain a URL tag indicating a resource that the first principal ultimately wishes to access after authentication; in the Liberty and SAML specifications this is referred to as the target URL. In such a scenario, the front-end service may parse the target URL from the request URL received, at 145, and reissue it to the second principal, at 120. Alternatively, the front-end service may cache the request URL before redirecting the first principal to the authentication service, at 135. The request is reissued to the second principal, at 120, and the response from the second principal is obtained. The front-end service may determine, at 125, whether

Claim	Exemplary Citation from Burch
	<p>the response from the second principal is an authentication request. Since the reissued request is the same that prompted an authentication request previously, the response, at 120, will likely be an authentication request. At 130, the front-end service determines whether the first principal has authenticated its identity, as required by the policy information. Since the first principal was redirected to the authentication service, at 140, and a credential authenticating the principal's identity was retrieved, at 150, this determining is positive."</p> <p>Burch at ¶ 51-52:  "Thus, as part of the message processing function associated with the front-end service, the front-end service may set the authorization header on each message received from the first principal as it is routed to the second principal. Under this scenario, the second principal validates the information provided in the HTTP authorization header and authenticates the connection of the front-end service as if it were the first principal. The front-end service may include the HTTP authorization header in subsequent requests from the first principal to the second principal for the duration of the first principal's session, as may also be stipulated by the policy information. From the perspective of the second principal, it appears that the first principal itself has directly authenticated using its own means of authentication, and the front-end service may route the reply messages from the second principal to the first principal, at 175. Thus, the first principal is granted seamless access to the second principal. And access to the second principal has been protected using a multifactor authentication mechanism, without altering the processing flow or processing aspects associated with the second principal. In other words, the multifactor authentication technique was implemented and enforced with no changes or modifications to the second principal. Using this technique, a high level of security is achievable with a minimal impact on existing or new application services that have no native multifactor authentication functionality."</p> <p>Burch at ¶69:  "The redirection, at 270, may have directed the first principal to a resource provided by a second principal, as depicted at 275. In this case, the receiver of the request (a front-end service, such as the one presented with respect to the method 100 of the FIG. 1 that provides access control to the second principal) may identify the request as including authentication information. As such, the receiver may issue a request to obtain the authentication credential associated with the first principal, the request being received at 280. The request may be issued over a communication link secured via mutually</p>

Claim	Exemplary Citation from Burch
	<p>authenticated SSL. The request, at 280, may include identifying information included in the redirection URL, at 270. Alternatively, if no authentication credential for the first principal was generated, at 240, but instead a connection or session based authentication scheme was used, the authentication service may generate a credential for the first principal when the request, at 275, is received. At 285 the authentication credential may be provided to the requester who may then verify the credential and authenticate the first principal, providing the first principal with access to the application (second principal or resources of the second principal)."</p> <p>Burch at ¶ 100-101:  "After obtaining the one or more secondary identifiers 497, user 405 may be prompted to verify its knowledge of the identifiers 497 via communications link 455. The communications link 455 may be secured via SSL to prevent disclosure of the credentials. The user 405 may respond to the request for the identifiers 497 via the link 455. Upon receipt of the identifiers 497 the authentication service 460 may verify the user's 405 knowledge of the identifiers 497. If successful, the user 405 may be expected, depending upon the policy information 465, to respond to additional authentication queries regarding the secondary identifiers 497. Further, upon verification of a first set of secondary identifiers 470, the authentication service 460 may be required by the policy 465, to obtain and verify the user's 405 knowledge of additional secondary identifiers 497. Once user 405 has authenticated in accordance with the policy 465, the authentication service 460 may generate an authentication credential 480 for the user 405. The credential 480 may be used to authenticate the identity of user 405 to other sites and/or principals 440. User 405 may then be redirected back to the application server 440. The URL used to redirect user 405 may include a token identifying the authentication service 460 and authenticated credential 480 generated for user 405. The token may be encoded into the URL string in accordance with the Liberty or SAML specifications. Alternatively, the authentication credential 480 may be encoded into the request itself. For instance, if the authentication service 460 were acting as a SAML Authentication Authority performing the SAML POST profile, the credential 480 transported via the as the payload of an HTTP POST to the application server 440."</p>
<p><b>[3]</b> The method of claim 1, wherein supplying further includes redirecting the principal to the identity</p>	<p>Burch discloses and/or renders obvious that supplying further includes redirecting the principal to the identity service and including with the redirection the new authentication request and the new authentication response represented by the authentication message, and the identity service</p>

<b>Claim</b>	<b>Exemplary Citation from Burch</b>
<p>service and including with the redirection the new authentication request and the new authentication response represented by the authentication message, and the identity service authenticates the principal automatically in response to the new authentication response included with the authentication message.</p>	<p>authenticates the principal automatically in response to the new authentication response included with the authentication message.</p>

Claim

Exemplary Citation from Burch

Burch at Fig. 1 and corresponding text:

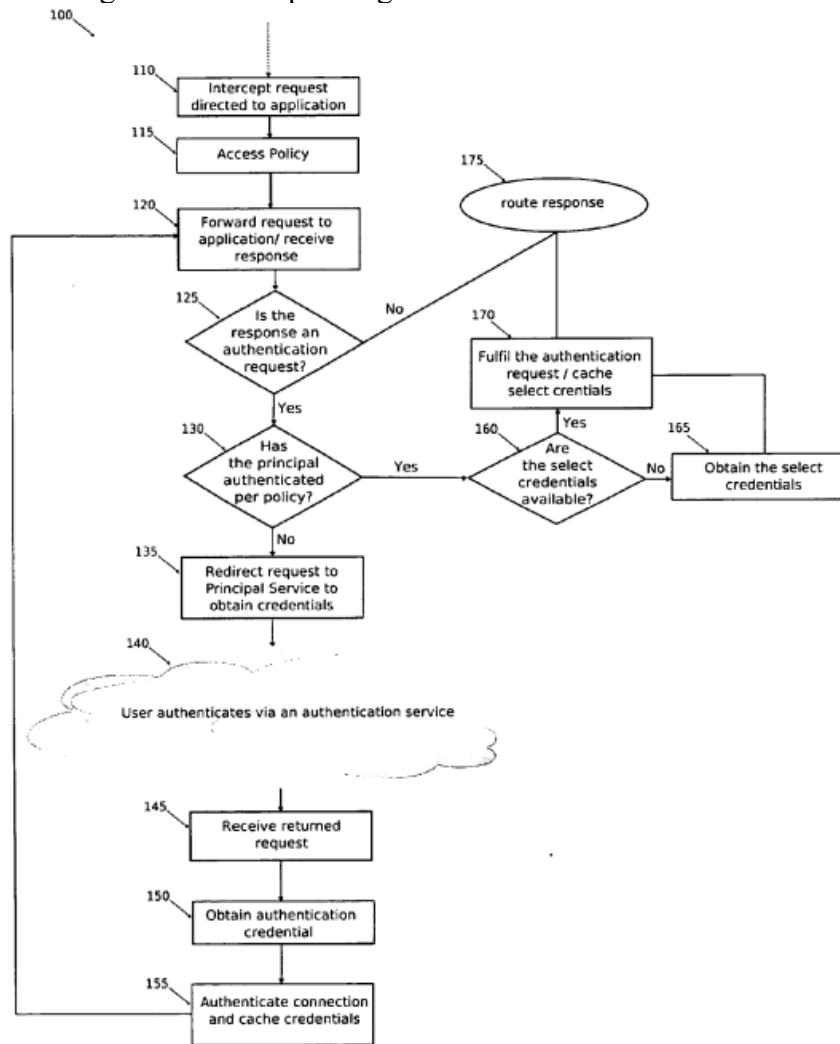


Figure 1

**Claim**

**Exemplary Citation from Burch**

Burch at Fig. 2 and corresponding text:

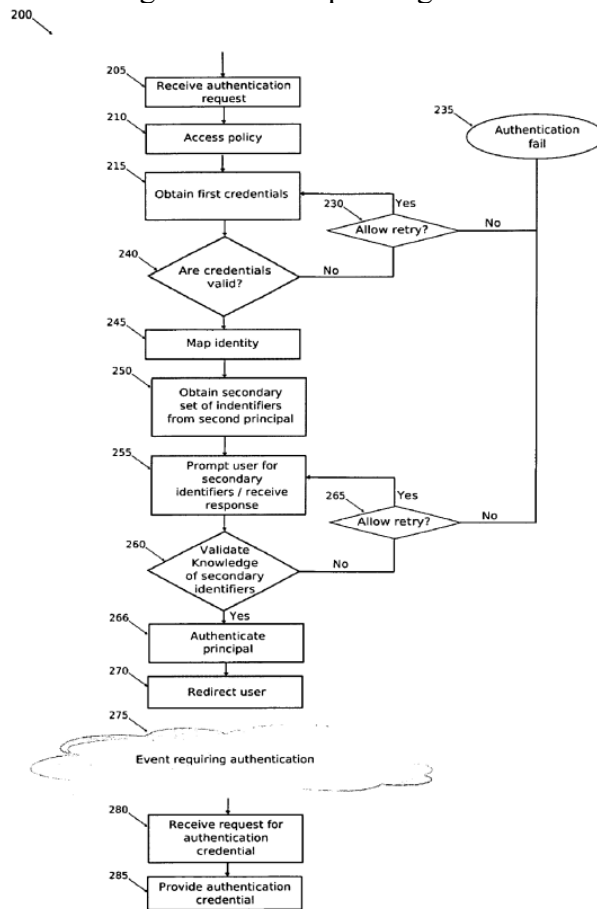


Figure 2

**Claim**

**Exemplary Citation from Burch**

Burch at Fig. 3 and corresponding text:

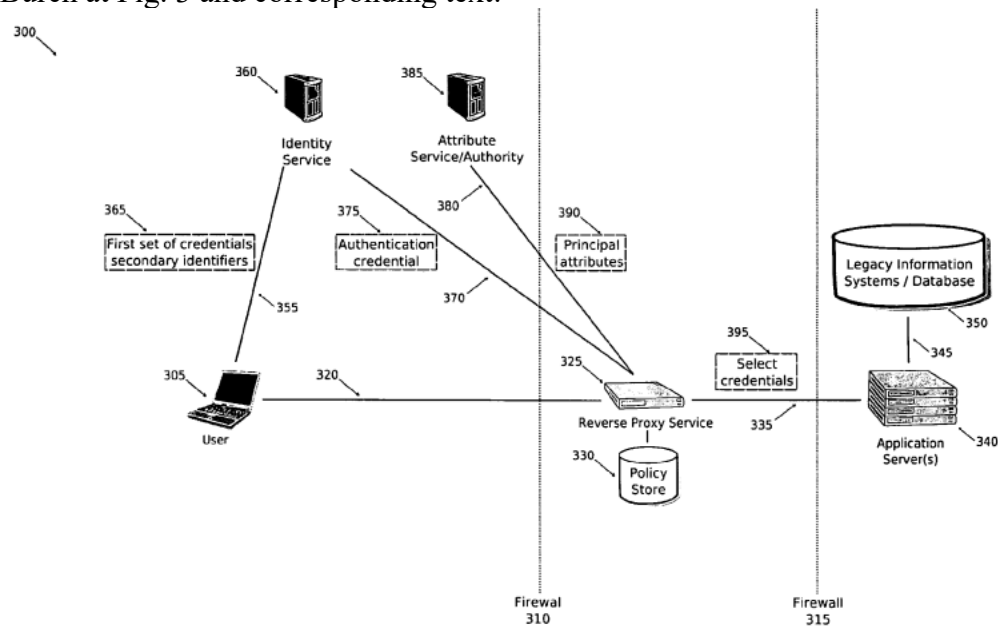


Figure 3

Burch at ¶45-46:

"After successfully authenticating the session of the first principal, at 155, the first principal's request may be reissued to the second principal back at the processing depicted in 120. Reissuing the original request may be done using a variety of different mechanisms. For example, in the Liberty and SAML context, the received request, at 145, may contain a URL tag indicating a resource that the first principal ultimately wishes to access after authentication; in the Liberty and SAML specifications this is referred to as the target URL. In such a scenario, the front-end service may parse the target URL from the request URL received, at 145, and reissue it to the second principal, at 120. Alternatively, the front-end service may cache the request URL before redirecting the first principal to the authentication service, at 135. The request is reissued to the second principal, at 120, and the response from the second principal is obtained. The front-end service may determine, at 125, whether

Claim	Exemplary Citation from Burch
	<p>the response from the second principal is an authentication request. Since the reissued request is the same that prompted an authentication request previously, the response, at 120, will likely be an authentication request. At 130, the front-end service determines whether the first principal has authenticated its identity, as required by the policy information. Since the first principal was redirected to the authentication service, at 140, and a credential authenticating the principal's identity was retrieved, at 150, this determining is positive."</p> <p>Burch at ¶ 51-52:  "Thus, as part of the message processing function associated with the front-end service, the front-end service may set the authorization header on each message received from the first principal as it is routed to the second principal. Under this scenario, the second principal validates the information provided in the HTTP authorization header and authenticates the connection of the front-end service as if it were the first principal. The front-end service may include the HTTP authorization header in subsequent requests from the first principal to the second principal for the duration of the first principal's session, as may also be stipulated by the policy information. From the perspective of the second principal, it appears that the first principal itself has directly authenticated using its own means of authentication, and the front-end service may route the reply messages from the second principal to the first principal, at 175. Thus, the first principal is granted seamless access to the second principal. And access to the second principal has been protected using a multifactor authentication mechanism, without altering the processing flow or processing aspects associated with the second principal. In other words, the multifactor authentication technique was implemented and enforced with no changes or modifications to the second principal. Using this technique, a high level of security is achievable with a minimal impact on existing or new application services that have no native multifactor authentication functionality."</p> <p>Burch at ¶69:  "The redirection, at 270, may have directed the first principal to a resource provided by a second principal, as depicted at 275. In this case, the receiver of the request (a front-end service, such as the one presented with respect to the method 100 of the FIG. 1 that provides access control to the second principal) may identify the request as including authentication information. As such, the receiver may issue a request to obtain the authentication credential associated with the first principal, the request being received at 280. The request may be issued over a communication link secured via mutually</p>

Claim	Exemplary Citation from Burch
	<p>authenticated SSL. The request, at 280, may include identifying information included in the redirection URL, at 270. Alternatively, if no authentication credential for the first principal was generated, at 240, but instead a connection or session based authentication scheme was used, the authentication service may generate a credential for the first principal when the request, at 275, is received. At 285 the authentication credential may be provided to the requester who may then verify the credential and authenticate the first principal, providing the first principal with access to the application (second principal or resources of the second principal)."</p> <p>Burch at ¶ 100-101:  "After obtaining the one or more secondary identifiers 497, user 405 may be prompted to verify its knowledge of the identifiers 497 via communications link 455. The communications link 455 may be secured via SSL to prevent disclosure of the credentials. The user 405 may respond to the request for the identifiers 497 via the link 455. Upon receipt of the identifiers 497 the authentication service 460 may verify the user's 405 knowledge of the identifiers 497. If successful, the user 405 may be expected, depending upon the policy information 465, to respond to additional authentication queries regarding the secondary identifiers 497. Further, upon verification of a first set of secondary identifiers 470, the authentication service 460 may be required by the policy 465, to obtain and verify the user's 405 knowledge of additional secondary identifiers 497. Once user 405 has authenticated in accordance with the policy 465, the authentication service 460 may generate an authentication credential 480 for the user 405. The credential 480 may be used to authenticate the identity of user 405 to other sites and/or principals 440. User 405 may then be redirected back to the application server 440. The URL used to redirect user 405 may include a token identifying the authentication service 460 and authenticated credential 480 generated for user 405. The token may be encoded into the URL string in accordance with the Liberty or SAML specifications. Alternatively, the authentication credential 480 may be encoded into the request itself. For instance, if the authentication service 460 were acting as a SAML Authentication Authority performing the SAML POST profile, the credential 480 transported via the as the payload of an HTTP POST to the application server 440."</p>
<p>[4] The method of claim 3, wherein supplying further includes representing the new authentication response</p>	<p>Burch discloses and/or renders obvious that supplying further includes representing the new authentication response as a first authentication token that informs the identity service that the principal is currently already properly authenticated to the processing associated with the method.</p>

Claim	Exemplary Citation from Burch
<p>as a first authentication token that informs the identity service that the principal is currently already properly authenticated to the processing associated with the method.</p>	<p>Burch at Fig. 1 and corresponding text:</p> <pre> graph TD     100(( )) --&gt; 110[Intercept request directed to application]     110 --&gt; 115[Access Policy]     115 --&gt; 120[Forward request to application / receive response]     120 --&gt; 125{Is the response an authentication request?}     125 -- No --&gt; 175([route response])     125 -- Yes --&gt; 130{Has the principal authenticated per policy?}     130 -- Yes --&gt; 160{Are the select credentials available?}     160 -- No --&gt; 165[Obtain the select credentials]     165 --&gt; 170[Fulfill the authentication request / cache select credentials]     160 -- Yes --&gt; 170     170 --&gt; 175     130 -- No --&gt; 135[Redirect request to Principal Service to obtain credentials]     135 --&gt; 140     140 --- 145[Receive returned request]     145 --&gt; 150[Obtain authentication credential]     150 --&gt; 155[Authenticate connection and cache credentials]     155 --&gt; 120   </pre> <p>Figure 1</p>

**Claim**

**Exemplary Citation from Burch**

Burch at Fig. 2 and corresponding text:

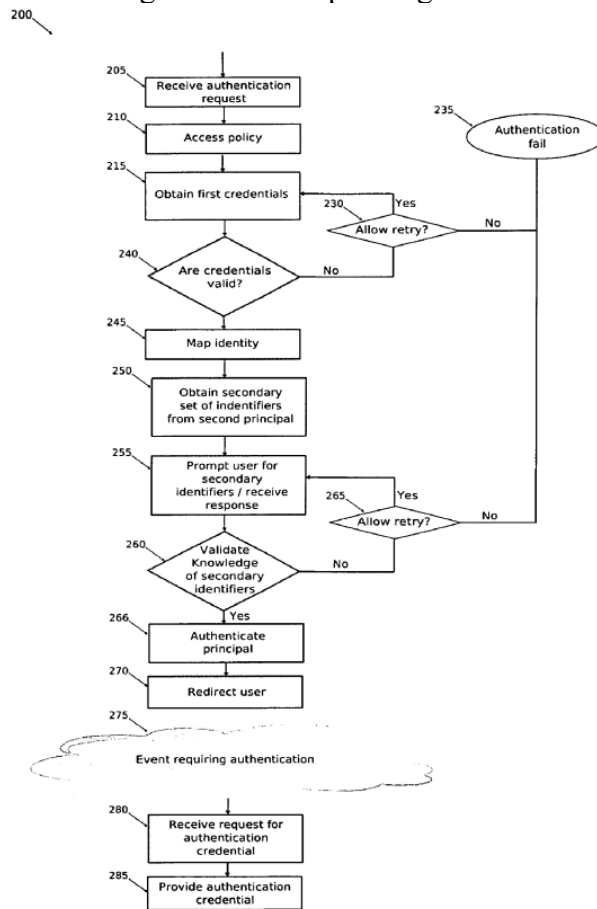


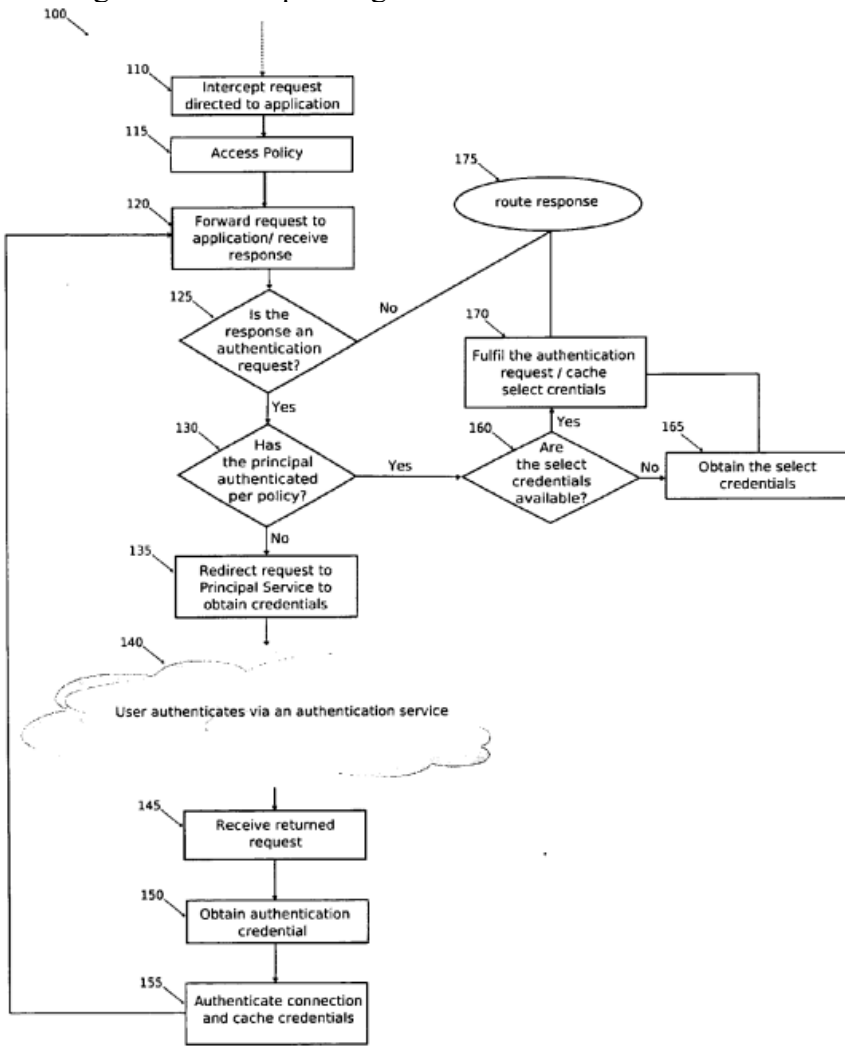
Figure 2

Burch at Fig. 3 and corresponding text:

Claim	Exemplary Citation from Burch
	<div data-bbox="604 240 1596 852" data-label="Diagram"> </div> <p data-bbox="1050 885 1144 917">Figure 3</p> <p data-bbox="598 966 819 998">Burch at ¶45-46:</p> <p data-bbox="598 998 1900 1396">"After successfully authenticating the session of the first principal, at 155, the first principal's request may be reissued to the second principal back at the processing depicted in 120. Reissuing the original request may be done using a variety of different mechanisms. For example, in the Liberty and SAML context, the received request, at 145, may contain a URL tag indicating a resource that the first principal ultimately wishes to access after authentication; in the Liberty and SAML specifications this is referred to as the target URL. In such a scenario, the front-end service may parse the target URL from the request URL received, at 145, and reissue it to the second principal, at 120. Alternatively, the front-end service may cache the request URL before redirecting the first principal to the authentication service, at 135. The request is reissued to the second principal, at 120, and the response from the second principal is obtained. The front-end service may determine, at 125, whether the response from the second principal is an authentication request. Since the reissued request is the</p>

Claim	Exemplary Citation from Burch
	<p>same that prompted an authentication request previously, the response, at 120, will likely be an authentication request. At 130, the front-end service determines whether the first principal has authenticated its identity, as required by the policy information. Since the first principal was redirected to the authentication service, at 140, and a credential authenticating the principal's identity was retrieved, at 150, this determining is positive."</p> <p>Burch at ¶ 51-52:  "Thus, as part of the message processing function associated with the front-end service, the front-end service may set the authorization header on each message received from the first principal as it is routed to the second principal. Under this scenario, the second principal validates the information provided in the HTTP authorization header and authenticates the connection of the front-end service as if it were the first principal. The front-end service may include the HTTP authorization header in subsequent requests from the first principal to the second principal for the duration of the first principal's session, as may also be stipulated by the policy information. From the perspective of the second principal, it appears that the first principal itself has directly authenticated using its own means of authentication, and the front-end service may route the reply messages from the second principal to the first principal, at 175. Thus, the first principal is granted seamless access to the second principal. And access to the second principal has been protected using a multifactor authentication mechanism, without altering the processing flow or processing aspects associated with the second principal. In other words, the multifactor authentication technique was implemented and enforced with no changes or modifications to the second principal. Using this technique, a high level of security is achievable with a minimal impact on existing or new application services that have no native multifactor authentication functionality."</p> <p>Burch at ¶69:  "The redirection, at 270, may have directed the first principal to a resource provided by a second principal, as depicted at 275. In this case, the receiver of the request (a front-end service, such as the one presented with respect to the method 100 of the FIG. 1 that provides access control to the second principal) may identify the request as including authentication information. As such, the receiver may issue a request to obtain the authentication credential associated with the first principal, the request being received at 280. The request may be issued over a communication link secured via mutually authenticated SSL. The request, at 280, may include identifying information included in the</p>

Claim	Exemplary Citation from Burch
	<p>redirection URL, at 270. Alternatively, if no authentication credential for the first principal was generated, at 240, but instead a connection or session based authentication scheme was used, the authentication service may generate a credential for the first principal when the request, at 275, is received. At 285 the authentication credential may be provided to the requester who may then verify the credential and authenticate the first principal, providing the first principal with access to the application (second principal or resources of the second principal)."</p> <p>Burch at ¶ 100-101:  "After obtaining the one or more secondary identifiers 497, user 405 may be prompted to verify its knowledge of the identifiers 497 via communications link 455. The communications link 455 may be secured via SSL to prevent disclosure of the credentials. The user 405 may respond to the request for the identifiers 497 via the link 455. Upon receipt of the identifiers 497 the authentication service 460 may verify the user's 405 knowledge of the identifiers 497. If successful, the user 405 may be expected, depending upon the policy information 465, to respond to additional authentication queries regarding the secondary identifiers 497. Further, upon verification of a first set of secondary identifiers 470, the authentication service 460 may be required by the policy 465, to obtain and verify the user's 405 knowledge of additional secondary identifiers 497. Once user 405 has authenticated in accordance with the policy 465, the authentication service 460 may generate an authentication credential 480 for the user 405. The credential 480 may be used to authenticate the identity of user 405 to other sites and/or principals 440. User 405 may then be redirected back to the application server 440. The URL used to redirect user 405 may include a token identifying the authentication service 460 and authenticated credential 480 generated for user 405. The token may be encoded into the URL string in accordance with the Liberty or SAML specifications. Alternatively, the authentication credential 480 may be encoded into the request itself. For instance, if the authentication service 460 were acting as a SAML Authentication Authority performing the SAML POST profile, the credential 480 transported via the as the payload of an HTTP POST to the application server 440."</p>
<p>[5] The method of claim 4, wherein supplying further includes adding a second authentication to a second redirection of the principal,</p>	<p>Burch discloses and/or renders obvious that supplying further includes adding a second authentication to a second redirection of the principal, wherein the second authentication represents authentication of the principal to the identity service and wherein the second redirection directs the principal to request a target service that is to be proxied on behalf of the principal from the identity service.</p>

Claim	Exemplary Citation from Burch
<p>wherein the second authentication represents authentication of the principal to the identity service and wherein the second redirection directs the principal to request a target service that is to be proxied on behalf of the principal from the identity service.</p>	<p>Burch at Fig. 1 and corresponding text:</p>  <pre> graph TD     100(( )) --&gt; 110[Intercept request directed to application]     110 --&gt; 115[Access Policy]     115 --&gt; 120[Forward request to application / receive response]     120 --&gt; 125{Is the response an authentication request?}     125 -- No --&gt; 175([route response])     125 -- Yes --&gt; 130{Has the principal authenticated per policy?}     130 -- Yes --&gt; 160{Are the select credentials available?}     130 -- No --&gt; 135[Redirect request to Principal Service to obtain credentials]     135 --&gt; 140     140 --- 145[Receive returned request]     145 --&gt; 150[Obtain authentication credential]     150 --&gt; 155[Authenticate connection and cache credentials]     155 --&gt; 120     160 -- Yes --&gt; 170[Fulfill the authentication request / cache select credentials]     160 -- No --&gt; 165[Obtain the select credentials]     165 --&gt; 170     170 --&gt; 175   </pre> <p>Figure 1</p>

**Claim**

**Exemplary Citation from Burch**

Burch at Fig. 2 and corresponding text:

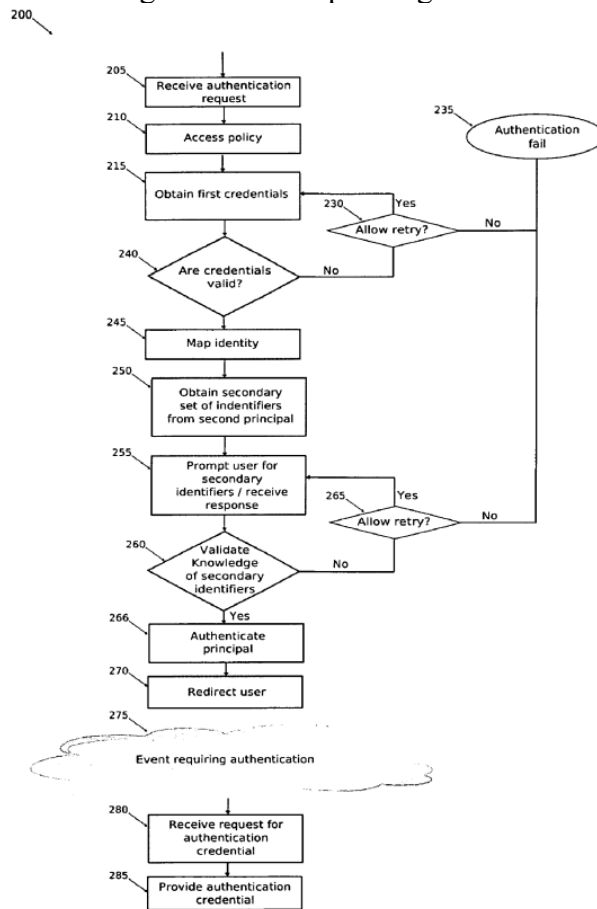


Figure 2

**Claim**

**Exemplary Citation from Burch**

Burch at Fig. 3 and corresponding text:

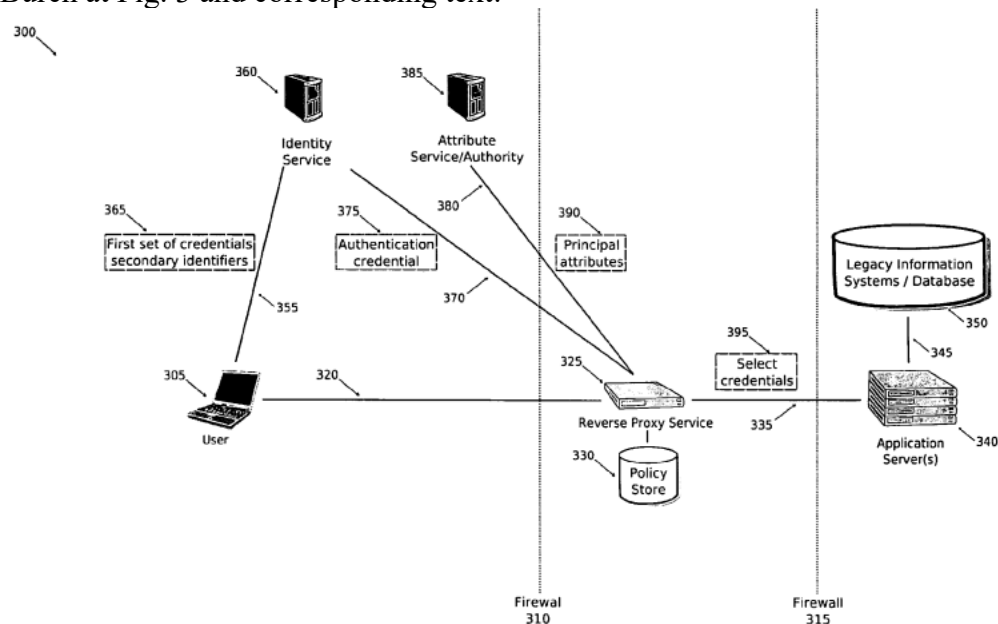


Figure 3

Burch at ¶ 47-49:

"At 160, the front-end service determines whether the select credentials, required by the second principal to authenticate the identity of the first principal, are available. This processing may be used because the select credentials requested by the second principal may be completely distinct from those used at the authentication service discussed in connection with the processing at 140, 150, or 155. This situation may arise if developers of the application, associated with the second principal, desired to increase security, but determined that updates to the legacy authentication service of the second principal would too expensive and/or time consuming to be practical. In such a situation, multifactor authentication may take place at the authentication service while the legacy authentication scheme of the second principal may remain unchanged. Similarly, in developing a new application, developers may determine that it may be more cost effective to leverage an existing or

Claim	Exemplary Citation from Burch
	<p>standardized authentication service; rather than to develop or buy a completely new proprietary security system or service. In such cases, the authentication service itself may be very simple (i.e., just enough to make the association between a first principal and an identity on the system). But, by enforcing a security policy, at 115, and by utilizing the authentication service, at 140, the second principal may be protected using a multifactor authentication mechanism. This is so, because no first principal may access the second principal directly; instead access is routed through the front-end service where the front-end service requires that the first principals authenticate using mechanisms or techniques that comply with the second principal's policy, which identifies an additional multifactor authentication technique. By way of example, the second principal may be an application server of an on-line banking system or service. In this case, the authentication service of the second principal may only require an account number and PIN to obtain access to the on-line banking service. However, in order to increase security, a policy may stipulate that the first principal is to also authenticate using a sophisticated multifactor authentication mechanism, depicted at 140. The multifactor authentication mechanism may include a password and, perhaps, the principal's last bank transaction (or any other additional authentication criterion or criteria). As such, it may be common and even desirable, that there be a mismatch between the authentication credentials required for authentication to the authentication service from those that are sent to the second principal, at 170. However, in order for this scenario to function seamlessly, the front-end service provides the second principal with the credentials it expects. Yet, the first principal never gains true access unabated until the additional credentials are also provided (last bank transaction). If the check, at 160, indicates that the select credentials required by the second principal are not available, the select credentials have be obtained, at 165. There are a number of ways the select credentials may be obtained. For example, a Liberty or SAML attribute request could be issued to a Liberty or SAML attribute authority associated with the first principal. As per the proceeding example, an attribute request, at 165, may be issued for the first principal's account number and PIN; the attribute request including information, which allows the attribute authority to verify that the request was made from a trusted entity. Responsive to this request, the attribute authority may return an attribute assertion containing the required select credentials. The communication of the request and response may be secured using mutually authenticated secure socket layer (SSL) transport, IPsec, or any other secure communications protocol. The contents of the attribute authority may be authenticated using PKI tools or other message authentication mechanisms. Alternatively, the policy accessed, at 125, may have specified the select credentials used for authentication to the second principal. Thus, the authentication service</p>

Claim	Exemplary Citation from Burch
	<p>may have provided the credentials as part of the authentication credential message received at 150 and cached at 155."</p> <p>Burch at ¶ 51-52:  "Thus, as part of the message processing function associated with the front-end service, the front-end service may set the authorization header on each message received from the first principal as it is routed to the second principal. Under this scenario, the second principal validates the information provided in the HTTP authorization header and authenticates the connection of the front-end service as if it were the first principal. The front-end service may include the HTTP authorization header in subsequent requests from the first principal to the second principal for the duration of the first principal's session, as may also be stipulated by the policy information. From the perspective of the second principal, it appears that the first principal itself has directly authenticated using its own means of authentication, and the front-end service may route the reply messages from the second principal to the first principal, at 175. Thus, the first principal is granted seamless access to the second principal. And access to the second principal has been protected using a multifactor authentication mechanism, without altering the processing flow or processing aspects associated with the second principal. In other words, the multifactor authentication technique was implemented and enforced with no changes or modifications to the second principal. Using this technique, a high level of security is achievable with a minimal impact on existing or new application services that have no native multifactor authentication functionality."</p> <p>Burch at ¶79:  "Upon receipt of the redirected request, the reverse proxy 325 may identify authentication information embedded within it. This information could include a token identifying an authentication credential 375 encoded within the URL as in the Liberty and SAML specifications. Alternatively, the request itself may include an authentication credential 375 as an HTTP POST parameter. There are a number of ways authentication information may be included in the redirected request. The embodiments of the invention should not be read as limited to any particular technique."</p> <p>Burch at ¶ 83:  "Before issuing the target URL request to the application servers 340 , the reverse proxy server 325 may access the policy store 330 . The policy information 330 may indicate that the application</p>

Claim	Exemplary Citation from Burch
	<p>servers 340 themselves require select credentials 395 in order to authenticate the user 305 . This situation could arise if the application servers 340 originally included a proprietary authentication system. Thus, rather than modifying application server code, the reverse proxy 325 may be configured to provide the select credentials 395 that the application servers 340 expect via 335 . Similarly, the application servers 340 may themselves need to access a legacy information system (IS) or database 350 via link 345 in order to provide the services requested by user 305 . As such, the application servers 340 may need the select credentials 395 in order to access data specific to user 305 as required by the application. Given the disparate uses of the credentials 395 by the application servers 340 (some of which may not be strictly security related), the select credentials 395 may be completely distinct from those employed by user 305 to authenticate its identity to identity service 360 ."</p> <p>Burch at ¶ 93-94:  "The authentication service 460 may implemented as the authentication service discussed with respect to the method 200 in the FIG. 2 and/or the identity service 360 discussed with respect to the multifactor authentication system 300 of the FIG. 3. The authentication service 460 may support identity federation protocols and techniques such as those defined in the Liberty and SAML specifications. As such, the authentication service 460 may act as a Liberty Identity Service or Service Provider, SAML Authentication Authority, or the like. Similarly, the authentication service 460 may implement a proprietary or custom authentication protocol. Thus, the embodiments of this invention should not be read as limited to just the disclosed authentication systems and/or protocols. Upon receipt of the redirected authentication request, authentication service 460 may access policy store 465. The policy information 465 may indicate the particular authentication mechanisms that user 405 is to perform in order to authenticate its identity. In particular, the policy information 465 may specify a first set of credentials 470 and one or more secondary identifiers 497 that are to be produced by user 405 in order to authenticate its identity. In addition to specifying the one or more secondary identifiers 497, the policy information 465 may also indicate how these identifiers 497 may be obtained. After accessing policy information 465, the authentication service 460 may obtain from user 405 a first set of credentials 470. As discussed above, the first set of credentials 470 may include any number of different authentication materials, including, but not limited to, a username and password combination, a Smartcard, a token, PKI signature data, biometric information, etc. Upon receipt of this first set of credentials 470, the authentication service 460 may validate the</p>

Claim	Exemplary Citation from Burch
	<p>received credentials 470. A number of different verification mechanisms may be employed, as discussed above, such mechanisms may include: a hash based comparison for password based authentication, PKI signature verification, biometric methods, etc."</p> <p>Burch at ¶ 100-101:  "After obtaining the one or more secondary identifiers 497, user 405 may be prompted to verify its knowledge of the identifiers 497 via communications link 455. The communications link 455 may be secured via SSL to prevent disclosure of the credentials. The user 405 may respond to the request for the identifiers 497 via the link 455. Upon receipt of the identifiers 497 the authentication service 460 may verify the user's 405 knowledge of the identifiers 497. If successful, the user 405 may be expected, depending upon the policy information 465, to respond to additional authentication queries regarding the secondary identifiers 497. Further, upon verification of a first set of secondary identifiers 470, the authentication service 460 may be required by the policy 465, to obtain and verify the user's 405 knowledge of additional secondary identifiers 497. Once user 405 has authenticated in accordance with the policy 465, the authentication service 460 may generate an authentication credential 480 for the user 405. The credential 480 may be used to authenticate the identity of user 405 to other sites and/or principals 440. User 405 may then be redirected back to the application server 440. The URL used to redirect user 405 may include a token identifying the authentication service 460 and authenticated credential 480 generated for user 405. The token may be encoded into the URL string in accordance with the Liberty or SAML specifications. Alternatively, the authentication credential 480 may be encoded into the request itself. For instance, if the authentication service 460 were acting as a SAML Authentication Authority performing the SAML POST profile, the credential 480 transported via the as the payload of an HTTP POST to the application server 440."</p>
<p>[6] The method of claim 1, wherein supplying further includes representing the new authentication response as an instruction to the identity service to enforce its own independent authentication with the</p>	<p>Burch discloses and/or renders obvious that supplying further includes representing the new authentication response as an instruction to the identity service to enforce its own independent authentication with the principal before considering the principal authenticated to the identity service.</p>

Claim	Exemplary Citation from Burch
<p>principal before considering the principal authenticated to the identity service.</p>	<p>Burch at Fig. 1 and corresponding text:</p> <pre> graph TD     100(( )) --&gt; 110[Intercept request directed to application]     110 --&gt; 115[Access Policy]     115 --&gt; 120[Forward request to application / receive response]     120 --&gt; 125{Is the response an authentication request?}     125 -- No --&gt; 175([route response])     125 -- Yes --&gt; 130{Has the principal authenticated per policy?}     130 -- Yes --&gt; 160{Are the select credentials available?}     160 -- No --&gt; 165[Obtain the select credentials]     165 --&gt; 170[Fulfill the authentication request / cache select credentials]     160 -- Yes --&gt; 170     170 --&gt; 175     130 -- No --&gt; 135[Redirect request to Principal Service to obtain credentials]     135 --&gt; 140     subgraph 140 [User authenticates via an authentication service]         145[Receive returned request]         150[Obtain authentication credential]         155[Authenticate connection and cache credentials]     end     140 --&gt; 120   </pre> <p>Figure 1</p>

**Claim**

**Exemplary Citation from Burch**

Burch at Fig. 2 and corresponding text:

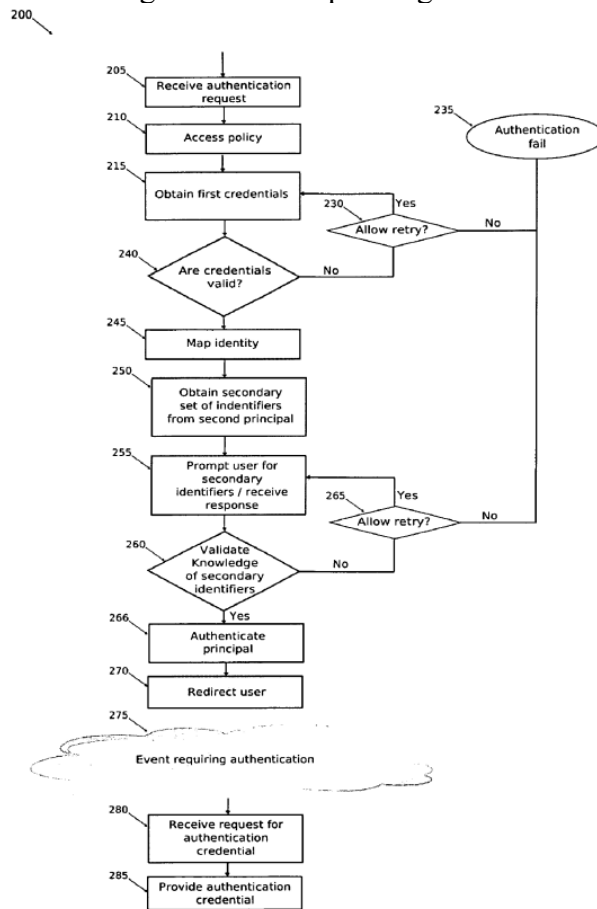


Figure 2

**Claim**

**Exemplary Citation from Burch**

Burch at Fig. 3 and corresponding text:

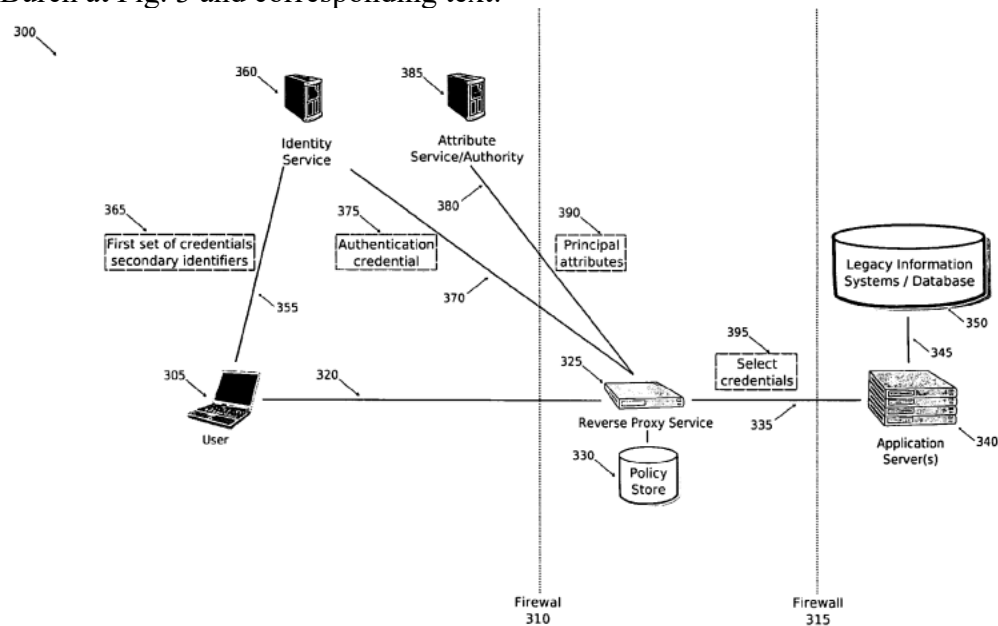


Figure 3

Burch at ¶ 47-49:

"At 160, the front-end service determines whether the select credentials, required by the second principal to authenticate the identity of the first principal, are available. This processing may be used because the select credentials requested by the second principal may be completely distinct from those used at the authentication service discussed in connection with the processing at 140, 150, or 155. This situation may arise if developers of the application, associated with the second principal, desired to increase security, but determined that updates to the legacy authentication service of the second principal would too expensive and/or time consuming to be practical. In such a situation, multifactor authentication may take place at the authentication service while the legacy authentication scheme of the second principal may remain unchanged. Similarly, in developing a new application, developers may determine that it may be more cost effective to leverage an existing or

Claim	Exemplary Citation from Burch
	<p>standardized authentication service; rather than to develop or buy a completely new proprietary security system or service. In such cases, the authentication service itself may be very simple (i.e., just enough to make the association between a first principal and an identity on the system). But, by enforcing a security policy, at 115, and by utilizing the authentication service, at 140, the second principal may be protected using a multifactor authentication mechanism. This is so, because no first principal may access the second principal directly; instead access is routed through the front-end service where the front-end service requires that the first principals authenticate using mechanisms or techniques that comply with the second principal's policy, which identifies an additional multifactor authentication technique. By way of example, the second principal may be an application server of an on-line banking system or service. In this case, the authentication service of the second principal may only require an account number and PIN to obtain access to the on-line banking service. However, in order to increase security, a policy may stipulate that the first principal is to also authenticate using a sophisticated multifactor authentication mechanism, depicted at 140. The multifactor authentication mechanism may include a password and, perhaps, the principal's last bank transaction (or any other additional authentication criterion or criteria). As such, it may be common and even desirable, that there be a mismatch between the authentication credentials required for authentication to the authentication service from those that are sent to the second principal, at 170. However, in order for this scenario to function seamlessly, the front-end service provides the second principal with the credentials it expects. Yet, the first principal never gains true access unabated until the additional credentials are also provided (last bank transaction). If the check, at 160, indicates that the select credentials required by the second principal are not available, the select credentials have be obtained, at 165. There are a number of ways the select credentials may be obtained. For example, a Liberty or SAML attribute request could be issued to a Liberty or SAML attribute authority associated with the first principal. As per the proceeding example, an attribute request, at 165, may be issued for the first principal's account number and PIN; the attribute request including information, which allows the attribute authority to verify that the request was made from a trusted entity. Responsive to this request, the attribute authority may return an attribute assertion containing the required select credentials. The communication of the request and response may be secured using mutually authenticated secure socket layer (SSL) transport, IPSec, or any other secure communications protocol. The contents of the attribute authority may be authenticated using PKI tools or other message authentication mechanisms. Alternatively, the policy accessed, at 125, may have specified the select credentials used for authentication to the second principal. Thus, the authentication service</p>

Claim	Exemplary Citation from Burch
	<p>may have provided the credentials as part of the authentication credential message received at 150 and cached at 155."</p> <p>Burch at ¶ 51-52:  "Thus, as part of the message processing function associated with the front-end service, the front-end service may set the authorization header on each message received from the first principal as it is routed to the second principal. Under this scenario, the second principal validates the information provided in the HTTP authorization header and authenticates the connection of the front-end service as if it were the first principal. The front-end service may include the HTTP authorization header in subsequent requests from the first principal to the second principal for the duration of the first principal's session, as may also be stipulated by the policy information. From the perspective of the second principal, it appears that the first principal itself has directly authenticated using its own means of authentication, and the front-end service may route the reply messages from the second principal to the first principal, at 175. Thus, the first principal is granted seamless access to the second principal. And access to the second principal has been protected using a multifactor authentication mechanism, without altering the processing flow or processing aspects associated with the second principal. In other words, the multifactor authentication technique was implemented and enforced with no changes or modifications to the second principal. Using this technique, a high level of security is achievable with a minimal impact on existing or new application services that have no native multifactor authentication functionality."</p> <p>Burch at ¶ 93:  "The authentication service 460 may implemented as the authentication service discussed with respect to the method 200 in the FIG. 2 and/or the identity service 360 discussed with respect to the multifactor authentication system 300 of the FIG. 3. The authentication service 460 may support identity federation protocols and techniques such as those defined in the Liberty and SAML specifications. As such, the authentication service 460 may act as a Liberty Identity Service or Service Provider, SAML Authentication Authority, or the like. Similarly, the authentication service 460 may implement a proprietary or custom authentication protocol. Thus, the embodiments of this invention should not be read as limited to just the disclosed authentication systems and/or protocols. Upon receipt of the redirected authentication request, authentication service 460 may access policy store 465. The policy information 465 may indicate the particular authentication mechanisms that</p>

Claim	Exemplary Citation from Burch
	<p>user 405 is to perform in order to authenticate its identity. In particular, the policy information 465 may specify a first set of credentials 470 and one or more secondary identifiers 497 that are to be produced by user 405 in order to authenticate its identity. In addition to specifying the one or more secondary identifiers 497, the policy information 465 may also indicate how these identifiers 497 may be obtained. After accessing policy information 465, the authentication service 460 may obtain from user 405 a first set of credentials 470. As discussed above, the first set of credentials 470 may include any number of different authentication materials, including, but not limited to, a username and password combination, a Smartcard, a token, PKI signature data, biometric information, etc. Upon receipt of this first set of credentials 470, the authentication service 460 may validate the received credentials 470. A number of different verification mechanisms may be employed, as discussed above, such mechanisms may include: a hash based comparison for password based authentication, PKI signature verification, biometric methods, etc."</p> <p>Burch at ¶ 100-101:  "After obtaining the one or more secondary identifiers 497, user 405 may be prompted to verify its knowledge of the identifiers 497 via communications link 455. The communications link 455 may be secured via SSL to prevent disclosure of the credentials. The user 405 may respond to the request for the identifiers 497 via the link 455. Upon receipt of the identifiers 497 the authentication service 460 may verify the user's 405 knowledge of the identifiers 497. If successful, the user 405 may be expected, depending upon the policy information 465, to respond to additional authentication queries regarding the secondary identifiers 497. Further, upon verification of a first set of secondary identifiers 470, the authentication service 460 may be required by the policy 465, to obtain and verify the user's 405 knowledge of additional secondary identifiers 497. Once user 405 has authenticated in accordance with the policy 465, the authentication service 460 may generate an authentication credential 480 for the user 405. The credential 480 may be used to authenticate the identity of user 405 to other sites and/or principals 440. User 405 may then be redirected back to the application server 440. The URL used to redirect user 405 may include a token identifying the authentication service 460 and authenticated credential 480 generated for user 405. The token may be encoded into the URL string in accordance with the Liberty or SAML specifications. Alternatively, the authentication credential 480 may be encoded into the request itself. For instance, if the authentication service 460 were acting as a SAML Authentication Authority performing the SAML</p>

Claim	Exemplary Citation from Burch
	POST profile, the credential 480 transported via the as the payload of an HTTP POST to the application server 440."
<p>[7] The method of claim 1 further comprising, interacting, by the machine, with the principal via a World-Wide Web (WWW) browser over the Internet using at least one of a Security Assertion Markup Language (SAML), a Liberty Alliance markup language, and Web Services (WS) Foundation markup language.</p>	<p>Burch discloses and/or renders obvious interacting, by the machine, with the principal via a World-Wide Web (WWW) browser over the Internet using at least one of a Security Assertion Markup Language (SAML), a Liberty Alliance markup language, and Web Services (WS) Foundation markup language.</p> <p>Burch at ¶15-17:  "A SAML encoded statement includes an assertion, a protocol, and a binding. There are generally three types of assertions: an authentication assertion used to validate a principal's electronic identity, an attribute assertion that includes specific attributes about the principal, an authorization assertion that identifies what the principal is permitted to do (e.g. policies). The protocol defines how a SAML processing application will ask for and receive the assertions. The binding defines how SAML message exchanges are mapped to Simple Object Access Protocol (SOAP) exchanges, or other protocol exchanges. In general terms, SAML techniques improve security between business-to-business (B2B) electronic transactions and business-to-consumer (B2C) electronic transactions. The techniques permit one principal to log in with a single transaction to a receiving principal and then use a variety of the receiving principal's disparate services by providing the SAML statements when needed. SAML techniques are not limited to inter-organization relationships (e.g., B2B or B2C); the techniques can be used within a single organization (intra-organization). SAML techniques are supported with a variety of network protocols, such as Hypertext Transfer Protocol (HTTP), Simple Mail Transfer Protocol (SMTP), File Transfer Protocol (FTP), SOAP, BizTalk, and Electronic Business XML (ebXML). The Organization for the Advancement of Structured Information Standards (OASIS) is the standards group for SAML. The techniques of Liberty are enhancements to the SAML techniques and may be used in connection with various embodiments to the SAML techniques and may also be used in connection with various embodiments of this invention. However, it is to be understood that SAML and Liberty techniques are not needed to perform the teachings of all embodiments of the invention. In this sense, the integration of SAML and Liberty techniques with some of the embodiments presented herein is intended to be enhancements or extensions to certain aspects of this invention, but other embodiments of this invention do not rely on or use the SAML and/or Liberty technologies."</p>