

UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE PATENT TRIAL AND APPEAL BOARD

WALMART INC. and WALMART STORES TEXAS, LLC.
Petitioner

v.

RAVENWHITE SECURITY, INC.
Patent Owner

Inter Partes Review No.: IPR2025-00810

**PETITION FOR INTER PARTES REVIEW OF
U.S. PATENT NO. 10,594,823
UNDER 35 U.S.C. §§311-319 AND 37 C.F.R. §§42.1-100, ET SEQ**

TABLE OF CONTENTS

	Page
I. Introduction.....	1
II. Statement of Precise Relief Requested.....	1
III. The '823 Patent.....	1
A. Overview of the '823 Patent.....	1
1. Effective Filing Date.....	1
2. Summary	3
B. Overview of the '823 Patent's File History	4
C. Person of Ordinary Skill in the Art	6
D. Claim Construction Under 37 C.F.R. §§42.104(b)(3)	7
E. Patent Owner's Infringement Contentions.....	8
IV. Ground 1: Hinton renders obvious claims 1-10	8
A. Overview of Hinton.....	8
B. Motivation to Combine	10
C. Claim 1	10
1. [1.pre].....	10
2. [1.a]	11
3. [1.b.i].....	12
4. [1.b.ii].....	14
5. [1.b.iii].....	16
6. [1.b.iv].....	18
7. [1.b.v].....	20
8. [1.c]	21
9. [1.d.i].....	22
10. [1.d.ii].....	25
11. [1.e]	26
12. [1.f].....	28
13. [1.g]	29

D.	Claim 2	30
E.	Claim 3	31
F.	Claim 4	31
G.	Claim 5	32
H.	Claim 6	33
	1. [6.pre]	33
	2. [6.a.i]	33
	3. [6.a.ii]	33
	4. [6.a.iii]	33
	5. [6.a.iv]	33
	6. [6.a.v]	34
	7. [6.b]	34
	8. [6.c.i]	34
	9. [6.c.ii]	34
	10. [6.d]	34
	11. [6.e]	34
I.	Claim 7	34
J.	Claim 8	35
K.	Claim 9	35
L.	Claim 10	35
V.	Ground 2: Varghese renders obvious claims 1, 3-6, and 8-10	36
	A. Overview of Varghese	36
	B. Motivation to Combine	39
	C. Claim 1	40
	1. [1.pre]	40
	2. [1.a]	41
	3. [1.b.i]	41
	4. [1.b.ii]	42
	5. [1.b.iii]	45

6.	[1.b.iv]	46
7.	[1.b.v]	48
8.	[1.c]	49
9.	[1.d.i] and [1.d.ii]	50
10.	[1.e]	53
11.	[1.f]	56
12.	[1.g]	59
D.	Claim 3	60
E.	Claim 4	60
F.	Claim 5	61
G.	Claim 6	61
1.	[6.pre]	61
2.	[6.a.i]	62
3.	[6.a.ii]	62
4.	[6.a.iii]	62
5.	[6.a.iv]	62
6.	[6.a.v]	62
7.	[6.b]	62
8.	[6.c.i] and [6.c.ii]	62
9.	[6.d]	63
10.	[6.e]	63
H.	Claim 8	63
I.	Claim 9	63
J.	Claim 10	63
VI.	Ground 3: Varghese In view of Hinton renders obvious claims 2 and 7	64
A.	Overview of Varghese	64
B.	Overview of Hinton	64
C.	Motivation to Combine	64
D.	Claim 2	66

E.	Claim 7	67
VII.	Compliance with Formal Requirements.....	68
A.	Mandatory Notices Under 37 C.F.R. §§42.8(b)(1)-(4).....	68
1.	Real Party-In-Interest.....	68
2.	Related Matters	68
3.	Lead and Backup Counsel	69
4.	Service Information.....	69
B.	Proof of Service on the Patent Owner.....	69
C.	Power of Attorney	70
D.	Standing.....	70
E.	Fees.....	70
VIII.	Conclusion	70

INDEX OF EXHIBITS

Exhibit No.	Description
1001	U.S. Patent No. 10,594,823 (the “ 823 patent ”).
1002	Declaration of Dr. Craig Wills
1003	File history of U.S. Patent No. 10,594,823.
1004	U.S. Patent No. 7,908,645 (“ Varghese ”).
1005	U.S. Patent Publication No. 2003/0115267 (“ Hinton ”)
1006	U.S. District Courts – Combined Civil and Criminal Federal Court Management Statistics (June 30, 2024).
1007	<i>Ravenwhite Licensing LLC v. The Home Depot, Inc.</i> , 2:24-cv-00688, Dkt. 1 (EDTX Aug. 21, 2024) (Complaint).
1008	<i>Ravenwhite Licensing LLC v. The Home Depot, Inc.</i> , 2:23-cv-00423, Infringement Contentions Cover Pleading (EDTX Dec. 4, 2023).
1009	<i>Ravenwhite Licensing LLC v. The Home Depot, Inc.</i> , 2:23-cv-00423, ’823 Infringement Contentions (EDTX Dec. 4, 2023).
1010	Provisional Application No. 60/732,025 (“ Provisional ”)
1011	<i>Ravenwhite Licensing LLC v. The Home Depot, Inc.</i> , 2:23-cv-00423, Plaintiff’s P.R. 4-1 Disclosures (EDTX Aug. 8, 2024).
1012	Balachander Krishnamurthy and Craig E. Wills, <i>Generating a privacy footprint on the Internet</i> , In Proceedings of the ACM SIGCOMM Internet Measurement Conference, pages 65-70, Rio de Janeiro, Brazil (Oct. 2006).
1013	Jon Purdy, <i>Session Management for Clustered Applications</i> (Feb. 2005) (available at https://www.oracle.com/technical-resources/articles/enterprise-architecture/session-management.html).
1014	“Local Shared Objects—‘Flash Cookies,’” Electronic Privacy Information Center (EPIC) (July 21, 2005) (available at https://archive.epic.org/privacy/cookies/flash.html).
1015	“The Pharming Guide (part 2)” (Dec. 14, 2004) (available at: http://www.technicalinfo.net/papers/Pharming2.html).
1016	“Misfortune Cookies: Adjusting Internet Explorer to Block Tracking Web Cookies,” Gibson Research Corporation (last modified Aug. 13, 2005) (available at https://www.grc.com/cookies.htm).
1017	Ileene Chernoff, “Cookie crumbs, an introduction to cookies,” SANS Institute (2005) (available at https://www.giac.org/paper/gsec/226/cookie-crumbs-introduction-cookies/100727).

Exhibit No.	Description
1018	Michael Nelte and Elton Saul, “Cookies: Weaving the Web into a State,” <i>Crossroads, The ACM Magazine for Students</i> , Vol. 7, Issue 1, pp. 10-13, (Sept. 1, 2000) (available at https://dl.acm.org/doi/10.1145/351092.351097).
1019	Edward W. Felten and Michael A. Schneider, “Timing Attacks on Web Privacy” (Nov. 25, 2002) (available at https://web.archive.org/web/20021125051243/http://www.cs.princeton.edu/sip/pub/webtiming.pdf).
1020	SecuriTeam.com, “Timing Attacks of Web Privacy (Paper and Specific Issue)” (Feb. 20, 2002) (available at https://web.archive.org/web/20021020062537/http://www.securiteam.com/securityreviews/5GP020A6LG.html).
1021	Martin Pool, “meantime: non-consensual http user tracking using caches” (last revised March 29, 2000) (available at https://sourcefrog.net/projects/meantime).
1022	United Virtualities, “United Virtualities Develops ID Backup to Cookies” (March 31, 2005) (available at https://web.archive.org/web/20050408075600/http://www.unitedvirtualities.com/UV-Pressrelease03-31-05.htm).
1023	Antone Gonsalves, “Company Bypasses Cookie-Deleting Consumers,” <i>Information Week</i> (March 31, 2005) (available at https://www.informationweek.com/it-leadership/company-bypasses-cookie-deleting-consumers).
1024	Dr. Craig Wills, CS3013 Course Notes Week 4 (2004) (available at https://web.cs.wpi.edu/~cs3013/c04/week4-memmgmt.pdf).
1025	Prof. Howard Hamilton, CS330 Course Notes Week 4 (2003) (available at https://www2.cs.uregina.ca/~hamilton/courses/330/notes/memory/MemoryHierarchy.html).
1026	John Schwartz, “Giving Web a Memory Cost Its Users Privacy,” <i>New York Times</i> (Sept. 4, 2001) (available at https://www.nytimes.com/2001/09/04/business/giving-web-a-memory-cost-its-users-privacy.html).
1027	Thomas Chung, “HOWTO: Installing Flash Plugin in Firefox Way,” <i>FedoraNEWS.ORG</i> (Sept. 16, 2004) (available at https://fedoranews.org/tchung/firefox-flash/).

Exhibit No.	Description
1028	Adrian Ludwig, “Macromedia® Flash® Platform Security and Macromedia Enterprise Solutions” (Sept. 2005) (available at https://www.adobe.com/platform/whitepapers/flashplatform_security_enterprise.pdf).

CHART OF CLAIMS

[1.pre] A system, comprising:
[1.a] one or more processors configured to:
[1.b.i] receive a network resource request from a client device, wherein the network resource request corresponds to a first cookie of a first type that was caused to be stored to the client device during a first previous network session,
[1.b.ii] wherein the first cookie of the first type was caused to be stored to the client device at least in part by causing the client device to initiate a set of network resource requests determined during the first previous network session,
[1.b.iii] wherein the client device initiating the set of network resource requests caused data representative of the set of network resource requests to be stored at the client device,
[1.b.iv] wherein a second cookie of a second type different from the first type was caused to be stored at the client device during a second previous network session, and
[1.b.v] wherein the first cookie of the first type is stored in a first client device browser storage area and the second cookie of the second type is stored in a second client device browser storage area different from the first client device browser storage area;
[1.c] based at least in part on the network resource request from the client device corresponding to the first cookie of the first type caused to be stored at the client device during the first previous network session, determine information that was encoded and stored in the client device;
[1.d.i] perform a first identification of at least one of the client device and a user of the client device using the first cookie of the first type,
[1.d.ii] wherein the first identification is performed using the first cookie of the first type at least in part by using the determined information that was encoded and stored in the client device;
[1.e] perform a second identification of at least one of the client device and the user of the client device using the second cookie of the second type; and
[1.f] perform a determination based at least in part on (1) a presence of a network resource request associated with one of the first cookie and the second cookie, and (2) an absence of a network resource request associated with the other of the first cookie and the second cookie; and
[1.g] a memory coupled to the one or more processors and configured to provide the one or more processors with instructions.
[2] The system of claim 1 wherein the client device is identified by a server in a domain different from a server that caused the two different types of cookies to

be stored to the client device during the first and second previous network sessions.
[3] The system of claim 1 wherein the client device is identifiable based at least in part on a presence of either one of the first cookie and the second cookie in the respective first and second client device browser storage areas.
[5] The system of claim 1 wherein in response to the performed determination, the one or more processors are configured to cause a third cookie to be stored at the client device, wherein the third cookie is one of the first type and the second type.
[6.pre] A method, comprising:
[6.a.i] receiving a network resource request from a client device, wherein the network resource request corresponds to a first cookie of a first type that was caused to be stored to the client device during a first previous network session,
[6.a.ii] wherein the first cookie of the first type was caused to be stored to the client device at least in part by causing the client device to initiate a set of network resource requests determined during the first previous network session,
[6.a.iii] wherein the client device initiating the set of network resource requests caused data representative of the set of network resource requests to be stored at the client device,
[6.a.iv] wherein a second cookie of a second type different from the first type was caused to be stored at the client device during a second previous network session, and
[6.a.v] wherein the first cookie of the first type is stored in a first client device browser storage area and the second cookie of the second type is stored in a second client device browser storage area different from the first client device browser storage area;
[6.b] based at least in part on the network resource request from the client device corresponding to the first cookie of the first type caused to be stored at the client device during the first previous network session, determining information that was encoded and stored in the client device; and
[6.c.i] performing a first identification of at least one of the client device and a user of the client device using the first cookie of the first type
[6.c.ii] wherein the first identification is performed using the first cookie of the first type at least in part by using the determined information that was encoded and stored in the client device; and
[6.d] performing a second identification of at least one of the client device and the user of the client device using the second cookie of the second type; and
[6.e] performing a determination based at least in part on (1) a presence of a network resource request associated with one of the first cookie and the second

cookie, and (2) an absence of a network resource request associated with the other of the first cookie and the second cookie.

[7] The method of claim 6 wherein the client device is identified by a server in a domain different from a server that caused the two different types of cookies to be stored to the client device during the first and second previous network sessions.

[8] The method of claim 6 wherein the client device is identifiable based at least in part on a presence of either one of the first cookie and the second cookie in the respective first and second client device browser storage areas.

[10] The method of claim 6 wherein in response to the performed determination, further comprising causing a third cookie to be stored at the client device, wherein the third cookie is one of the first type and the second type.

I. INTRODUCTION

Walmart Inc. and Walmart Stores Texas, LLC (“Petitioner”) petitions for *inter partes* review of U.S. Patent No. 10,594,823 (“’823 patent”) (EX1001). As shown below, the systems and methods described in the ’823 patent were known in the art.

II. STATEMENT OF PRECISE RELIEF REQUESTED

In accordance with 35 U.S.C. §311, Petitioner requests cancellation of claims 1-10 of the ’823 patent in view of the following grounds:

Ground	Claims	Stat. Basis	Prior Art
1	1-10	35 U.S.C. §103	Hinton in view of the knowledge of a POSITA
2	1, 3-6, 8-10	35 U.S.C. §103	Varghese in view of the knowledge of a POSITA
3	2, 7	35 U.S.C. §103	Varghese in view of Hinton

III. THE ’823 PATENT

A. Overview of the ’823 Patent

1. Effective Filing Date

The ’823 patent claims priority to a series of applications, the earliest of which is U.S. Patent Application Serial No. 11/590,083, filed October 31, 2006. Also, the ’823 patent claims priority to provisional 60/732,025 filed November 1, 2005 (“Provisional”). In litigation, Patent Owner’s purported exclusive licensee only asserted a priority claim of October 31, 2006. EX1008, 5. EX1002, ¶57.

“It is elementary patent law that a patent application is entitled to the benefit of the filing date of an earlier filed application only if the disclosure of the earlier application provides support for the claims of the later application, as required by 35 U.S.C. §112.” *In re Chu*, 66 F.3d 292, 297 (Fed. Cir. 1995). “In other words, the specification of the *provisional* must ‘contain a written description of the invention and the manner and process of making and using it, in such full, clear, concise, and exact terms,’ 35 U.S.C. § 112 ¶1, to enable an ordinarily skilled artisan to practice the invention *claimed* in the *non-provisional* application.” *New Railhead Mfg., LLC v. Vermeer Mfg. Co.*, 298 F. 3d 1290, 1294 (Fed. Cir. 2002) (emphasis in original). A prior application that merely renders the later-claimed invention obvious is not sufficient to meet the written description requirement—it must describe the claimed invention with all its limitations. *Tronzo v. Biomet, Inc.*, 156 F.3d 1154, 1158 (Fed. Cir. 1998); *Lockwood v. Am. Airlines, Inc.*, 107 F.3d 1565, 1572 (Fed. Cir. 1997). EX1002, ¶58.

The Provisional does not provide support for the challenged claims of the ’823 patent—it does not include disclosure of elements 1.b.ii, 1.b.iii, 1.b.iv, 1.f, 6.a.ii, 6.a.iii, or 6.e. EX1010. Accordingly, the ’823 patent is not entitled to the filing date of the Provisional. EX1002, ¶59.

2. Summary

The '823 patent “relates generally to client-server communications and more specifically to causing a browser to store information in a browser storage area of a client device.” EX1001, 1:31-34. The information stored in the browser storage area of the client device includes cache cookies, which, unlike a standard cookie, “cannot be blocked or cleared via spyware or a browser setting.” EX1001, 5:21-22. The cache cookie is similar to a standard cookie in that it can identify the client device. EX1001, 5:16-18. For example, when “browser 116 establishes a second network session with the server 104 ... the server 104 ‘reads’ the data (the cache cookie 174) ... to identify the client device.” EX1001, 5:29-34. EX1002, ¶60.

The '823 patent discloses two types of cache cookies stored in different areas of the browser storage area. EX1001, Fig. 2. The first type of cache cookie takes the form of URLs stored in the history cache of the browser storage area that the server causes the client device to visit. EX1001, 6:24-27 (“[o]ne way a server can “write” to the client's history cache 204 is by redirecting the user to other URLs (within or external to the server's domain space)”), 6:27-32 (“a server operating the domain www.server.com can redirect a browser to a URL of the form “www.server.com?Z” for any desired value of Z when the browser visits www.server.com, thereby inserting “www.server.com?Z” into the history cache

204 of the client”), 6:39-41 (“the URLs written to the history cache 204 by the server are an embodiment of the cache cookies”). EX1002, ¶61.

The second type of cache cookie takes the form of temporary internet files (TIF) that are stored in a TIF storage area of the browser. EX1001, 7:1-6 (“[i]n order to place an object X in the TIF area 212, a server can serve content to the browser that causes the browser to download object X” and the “server can verify whether the browser contains object X in its browser storage area 200 by, for example, redirecting the browser to a URL that contains object X”), 7:6-13. EX1002, ¶62.

B. Overview of the ’823 Patent’s File History

During prosecution, the Examiner issued Office Actions rejecting the pending claims under 35 U.S.C. §102 and §103. EX1003, 82-106, 147-171, 200-224. In response, the Applicant amended the claims adding the following limitations:

- “wherein the first cookie of the first type was caused to be stored to the client device at least in part by causing the client device to initiate a set of network resource requests determined during the previous network session, wherein the client device initiating the set of network resource request caused the data representative of the set of network resource requests to be stored at the client device.” EX1003, 130-139.

- “perform a first identification of at least one of the client device and a user of the client device using one of the first cookie of the first type and the second cookie of the second type; perform a second identification using a cookie not used in the first identification; and wherein one of the first and second identifications is performed using the first cookie of the first type at least in part by using the determined information that was encoded and stored in the client device” EX1003, 182-192.
- “wherein the first cookie of the first type is stored in a first client device storage area and the second cookie of the second type is stored in a second client device storage area different from the first client device storage area ... the cookie not used in the first identification comprising one of the first cookie of the first type and the second cookie of the second type ... perform a determination based at least in part on (1) a presence of a network resource request associated with one of the first cookie and the second cookie, and (2) an absence of a network resource request associated with the other of the first cookie and the second cookie.” EX1003, 273-282.

The Applicant filed an after-allowance amendment amending the claims to include two previous network sessions—“first previous network session” and “second

previous network session.” EX1003, 345-350. The Applicant also amended the claims to recite:

- “wherein the first identification is performed using the first cookie of the first type at least in part by using the determined information that was encoded and stored in the client device ... perform a second identification of at least one of the client device and the user of the client device using the second cookie of the second type.” EX1003, 345-350.
- “client device browser storage area.” EX1003, 345-350.

The Examiner entered the after-allowance amendment. EX1003, 356-357.

EX1002, ¶63-64.

C. Person of Ordinary Skill in the Art

A person of ordinary skill in the art at the time of the alleged invention of the '823 patent (October 31, 2006) (“POSITA”) would have had a Bachelors’ degree in electrical or computer engineering or a comparable field of study, plus approximately two to three years of professional experience with computer networks and digital information transmission techniques or other relevant industry experience. Additional graduate education could substitute for professional experience, and significant experience in the field could substitute for formal education. EX1002, ¶65.

D. Claim Construction Under 37 C.F.R. §§42.104(b)(3)

The challenged claims are interpreted using the same claim construction standard that is used to construe the claim in a civil action in federal district court. 37 C.F.R. § 42.100(b). Petitioner has applied Patent Owner’s purported exclusive licensee’s proposed claim constructions of plain meaning in *RavenWhite Licensing LLC v. Home Depot, Inc.*, 2:23-cv-00423 (EDTX) (the “Prior Litigation”) for all claim terms.¹ EX1007; EX1011. The Board and Federal Circuit have approved of this procedure in several matters. *See, e.g., Abbott Diabetes Care Inc. v. Dexcom, Inc.*, IPR2022-00913, Paper 14 (PTAB November 3, 2022); *Spherix Inc. v. Matal*, 703 F. App’x 982, 983 (Fed. Cir. 2017). Petitioner does not waive its right to later argue that some of the claim terms should be construed, for example in a litigation where additional claim construction issues may need to be resolved to determine infringement issues. Petitioner also does not waive its right to later argue that the challenged claims are invalid for reasons not raised in this Petition, including for reasons based on statutory grounds that are unavailable in *inter partes* review such as indefiniteness and ineligibility arguments under 35 U.S.C. 112 and 101,

¹ In subsequent litigation, the same plaintiff argued that a determination of subject matter eligibility was premature prior to claim construction, but did not provide claim construction proposals for any identified claim terms.

respectively. EX1002, ¶¶66-68.

E. Patent Owner’s Infringement Contentions

Infringement allegations are “probative” of whether Patent Owner is taking “inconsistent positions” between this proceeding and district court litigation and are “relevant of the credibility” of Patent Owner’s “characterization of the [asserted] Patent in this proceeding.” *Ericsson Inc. v. Intellectual Ventures II LLC*, IPR2014-00919, Paper 37 at 9-10 (PTAB Dec. 7, 2015). Accordingly, Petitioner includes the exclusive licensee’s current Complaint from *Ravenwhite Licensing LLC v. The Home Depot, Inc.*, 2:24-cv-00688, Dkt. 1 (EDTX Aug. 21, 2024) and its infringement contentions from the Prior Litigation. EX1007, ¶¶47-74; EX1009. EX1002, ¶69.

IV. GROUND 1: HINTON RENDERS OBVIOUS CLAIMS 1-10

A. Overview of Hinton

Hinton was filed December 19, 2001, published as U.S. Publication No. 2003/0115267 on June 19, 2003, and identifies IBM Corporation as its assignee. EX1005, cover page. Hinton is prior art under at least 35 U.S.C. §§102(a), 102(b), and 102(e). EX1002, ¶70.

Hinton generally relates to a system and method for “online user identification, authentication, and authorization” as they relate to “cross-domain log on technologies and technologies which create and manage virtual communities of online users.” EX1005, ¶7. For example, Hinton provides a

solution to “cross-domain single-sign-on system and method which allows an Internet user to establish a long-term relationship with participating domains, which gives the user the ability to go directly to participating domains, via bookmarks or direct URLs for example, without having to go through a home domain first.” EX1005, ¶15. Hinton’s e-community includes a user’s home domain and at least one other domain. EX1005, ¶49, Fig. 1. EX1002, ¶71.

To implement the cross-domain single-sign-on process, Hinton discloses the use of two cookies: a domain identity cookie (DIDC) and an e-community cookie (eCC). The DIDC is generated when the user enrolls in the e-community and is used to identify the user’s home domain. EX1005, ¶¶70-71. The eCC is a cookie that “acts as an ‘authenticator bookmark’ within a given DNS domain.” EX1005, ¶130. The user has “one e-community cookie set for each domain at which it has a current, authenticated (or vouch-for) session.” EX1005, ¶132. The eCC “indicates the security server or other plug-in location, and a URI at a plug-in location that can provide an authentication ‘vouch for’ token for that user.” EX1005, 133. Accordingly, Hinton discloses the use of two cookies – DIDC and eCC – to facilitate a single-sign-on process across domains. EX1002, ¶72.

Hinton is analogous art to the ’823 patent and was not considered during prosecution of the ’823 patent. EX1002, ¶73-74.

B. Motivation to Combine

As shown below, no combinations are required to arrive at the claimed invention. Therefore, no showing of a motivation to combine or reasonable expectation of success is required. *See Unification Techs. LLC v. Micron Tech Inc.*, 23-1348, 2024 WL 3738401, at *6 (Fed. Cir. Aug. 9, 2024). EX1002, ¶75.

C. Claim 1

Hinton discloses and renders claim 1 obvious. EX1002, ¶¶76-111.

1. [1.pre]

Hinton discloses *a system* (an affiliated domain server). EX1005, ¶¶45, 49, Fig. 1, Abstract. EX1002, ¶77-80.

Hinton discloses an e-community that includes at least two domains—home domain 103 and an affiliated domain (e.g., other domain 106 and/or another domain 108), each of which is associated with and implemented by one or more servers. EX1005, ¶45 (“an ‘e-community’ has many different ‘participants,’ including e-community members, or domains corresponding to the business units that are participating in the e-community”), ¶49 (“FIG. 1 illustrates a simple e-community architecture, where a user (100) accesses the e-community from their browser” where “[i]n this example, there are three participants in the e-community: the user’s home domain (103), an ‘other’ domain (106) and ‘another’ domain (108)”), (“e-community has, in general, more than two participants”), Abstract (“An Internet user transfers directly to a domain within an e-community without

returning to a home domain or re-authenticating. The user's home domain server prepares and forwards a home domain identity cookie (DIDC) with an enrollment request to a user's browser, with the enrollment request being redirected to an affiliated domain server in the e-community.'").

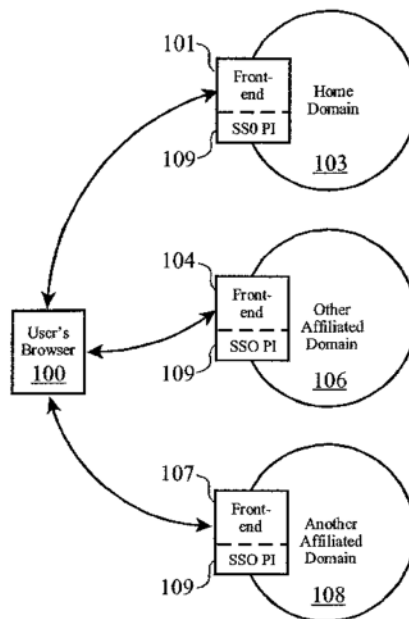


Figure 1

EX1005, FIG. 1. EX1002, ¶79.

2. [1.a]

Hinton discloses and renders obvious *one or more processors* (processors associated with an affiliated domain server). EX1005, Abstract, claim 10.

Affiliated domain (*e.g.*, other affiliated domain 106 or another affiliated domain 108) is associated with a server. EX1005, Abstract (“user's home domain server

prepares and forwards a home domain identity cookie ... with an enrollment request to a user's browser, with the enrollment request being redirected to an affiliated domain server in the e-community). Hinton does not explicitly state that the affiliated domain server includes a processor, but a POSITA would have understood and found obvious that a server includes a processor. EX1002, ¶81-82.

3. [1.b.i]

Hinton discloses and renders obvious this limitation. EX1002, ¶83-85.

Hinton discloses and renders obvious that the affiliated domain's processor is configured to *receive a network resource request* (access request triggering a vouch-for process) *from a client device* (user's browser 100) *wherein the network resource request corresponds to a first cookie of a first type* (DIDC). EX1005, ¶¶71, 81, 87-88, 93-97, 137. Hinton's affiliated domain receives a request from user 100 to access contents of its domain, which a POSITA would have understood to be a network resource request. EX1005, ¶137 ("vouch-for process occurs when an on-home front-end receives a request from a user that includes a domain identity cookie (DIDC) but not an e-community cookie (ECC) generated by that front end"). The request corresponds to a first cookie of the DIDC type, which has a specific format:

[0093] DIDC(x)={home domain=103,
[0094] vouch for URI=www.103.com/101/vouch_
for.htm,
[0095] e-community=sample,
[0096] creation date=Nov 1, 2000,
[0097] extensible data(x)=data(x)}, hash(info)

EX1005, ¶¶93-97. To the extent that Hinton does not explicitly state that the request is received *from a client device*, a POSITA would have understood that the request would have to come from a client device since the request originates from one, as shown by “user’s browser 100” in Fig. 1. A POSITA would have understood and found it obvious that the affiliated domain’s processor is configured to perform this function. EX1002, ¶84.

Hinton’s DIDC *was caused to be stored to the client device* (browser’s persistent cookie store) *during a first previous network session* (enrollment session on the home domain). Hinton’s system includes an enrollment process in which the user’s home domain authenticates the user and generates a DIDC to be stored at the user device to be used for subsequent network sessions. EX1005, ¶81 (“[f]irst, the user (100) accesses an ‘enroll in e-community’ resource at domain (103), at which time the SSO plug-in (109) at home domain (101) receives (52) this request and checks (53) if the user (100) has authenticated to the home domain (101)”), ¶87 (“plug-in (109) then builds an identity cookie DIDC (103)”), ¶88 (“user’s browser extracts (62) the DIDC and stores it in the browser’s persistent cookie

store, such as storing it in a cookie folder on a hard disk drive”). Hinton’s enrollment process is performed before the request triggering the vouch-for process. EX1005, ¶71 (“[t]he purpose of the domain identity cookie is to identify the user’s ‘home’ domain, to identify a URL in the user’s home domain that can ‘vouch for’ the user’s identity, and to identify the e-community in which this user is a participant”). This is further supported by the fact that Hinton’s “vouch-for” process requires the presence of a DIDC. EX1005, ¶137 (“vouch-for process occurs when a non-home front-end receives a request from a user that includes a domain identity cookie (DIDC)”). EX1002, ¶85.

4. [1.b.ii]

Hinton discloses this limitation. Hinton’s *first cookie of the first type (i.e., DIDC) was caused to be stored to the client device at least in part by causing the client device to initiate a set of network resource requests (redirections to other domains and back to the home domain) determined during the first previous network session (enrollment process)*. EX1005, ¶¶87-91. As part of the enrollment process, Hinton’s home domain builds the DIDC and sends the DIDC to the user device. EX1005, ¶87 (“plug-in (109) ... builds an identity cookie DIDC (103) and an “enrollment token” for the user (100), and creates a response, re-directed to the other community domain (106)”), ¶25 (“[t]he enrollment request is sent via the user’s browser using HTTP redirection” and includes “a home domain identity

cookie (DIDC) set by the home domain” which the “user’s browser extracts and stores the home DIDC”). The home domain’s response to the user causes the user to initiate a set of network resource requests by redirecting the response from the home domain to the other domains. EX1005, ¶88 (“user’s browser extracts ... DIDC ... and redirects the response to other community domain” and “plug-in (109) at the other domain (106) front-end (104) receives (63) the enrollment request from the home domain front-end (101) which was redirected through the user (100)”), Fig. 6 (element 62), ¶89 (“plug-in (109) at the other domain's front end (104) ‘unpacks’ the enrollment token, and builds a domain identity cookie for the user for the other domain (106)” which triggers “[a]n ‘enrollment successful’ message ... sent to the home domain’s front end (101) via redirection (63) through the user’s browser (100) along with the domain identity cookie for the other domain (106),” which “the user’s browser extracts (64) ... and puts in the browser’s persistent cookie store”). The network requests sent from the user to the other domains trigger the other domains to enroll the user at each of the other domains. EX1005, ¶90 (“The home domain (103) plug-in (109) at the first front-end (101) receives (65) the redirected ‘enrollment successful at other domain’ message”). As a result of this process, a DIDC, with indicators that the user was enrolled at the other domains, is caused to be stored at the user device. EX1005, ¶26 (“[a]n enrollment success message is sent by the affiliated domain to the home domain,

including the affiliated DIDC and a success indicator. Again, the message is sent via the user's browser using redirection. The user's browser extracts and stores the affiliated DIDC"), ¶90 ("home domain ... modifies (65) the home domain DIDC to include an 'enrollment success at other domain' symbol in the extensible attribute data" which "is then returned (65) to the user in the next response from the first front-end (101)"), ¶91 ("[i]n this manner, the home domain DIDC is 'built up' or accumulated to include indicators of successful enrollments at affiliated domains within the e-community"). In addition, Hinton teaches that "[i]f the user has not already been authenticated to the home domain, then the SSO plug-in (109) prompts (24) the user (100) for authentication information (e.g. user name and password), and performs (25) authentication verification. EX1005, ¶108. A POSITA would have understood that this authentication verification requires the client device to initiate a set of network resource requests determined during the first previous network session to provide such authentication information at the prompting of the SSO plug-in of the home domain. EX1002, ¶86-87.

5. [1.b.iii]

Hinton discloses this limitation, the *data representative of the set of network resource requests that is stored at the client device*. EX1005, ¶¶87-92. During the generation of the DIDC that is stored on the user device based on a client device's network resource requests, the home domain generates an enrollment token for the

user which is sent to other domains within the e-community, through a series of network redirects, in order to build-up the home domain DIDC. EX1005, ¶87 (“plug-in (109) ... builds ... an ‘enrollment token’ for the user (100), and creates a response, re-directed to the other community domain (106”), ¶89 (“plug-in (109) at the other domain’s front end (104) ‘unpacks’ the enrollment token, and builds an domain identity cookie for the user for the other domain (106)” and an “‘enrollment successful’ message is then sent to the home domain’s front-end (101) via redirection (63) through the user’s browser (100) along with the domain identity cookie for the other domain 106” which “the user’s browser extracts (64) ... and puts ... in the browser’s persistent cookie store”), ¶90 (“the home domain (103) plug-in (109) at the first front-end (101) receives (65) the redirected ‘enrollment successful at other domain’ message” and “modifies (65) the home domain DIDC to include ‘enrollment success at other domain’ symbol in the extensible attribute data” which “is then returned (65) to the user in the next response from the first front-end”), ¶91 (“the home domain DIDC is ‘built up’ or accumulated to include indicators of successful enrollments at affiliated domains within the e-community”), ¶91 (“[t]his process may continue for additional domains in the e-community, using the user's browser as a re-direction node in the communication path to pass enrollment tokens and success tokens between the home domain and the affiliated domain”). At the end of the network re-redirects, on

the user's device "the user's browser receives a persistent domain identity cookie set by each of the e-community members (103, 106, 108) in which the user has successfully been enrolled." EX1005, ¶92. This cookie is data representative of the set of network resource requests and it is stored on the client device, and the cookie was received because the client device initiated the set of network resource requests. The home DIDC that is built up to "include the multiple indicators of successful enrollments" constitutes the *data representative of the set of network resource requests that is stored at the client device*. EX1002, ¶88.

6. [1.b.iv]

Hinton discloses and renders obvious *a second cookie of a second type ("eCC") different from the first type was caused to be stored at the client device during a second previous network session (limitation 1.b.iv)*. EX1005, ¶35, ¶67, ¶¶93-97, ¶129, ¶130, ¶137, ¶¶147-148, ¶153, ¶¶171-178, ¶249. EX1002, ¶¶89-93.

The eCC cookie is a second cookie of a second type different than the DIDC. Hinton discloses an "e-Community Cookie" (eCC) that "is a-memory cookie that is valid within the DNS domain." EX1005, ¶130, ¶137, ¶147, ¶153.

The DIDC cookie type includes the following information:

[0093] DIDC(x)={home domain=103,
[0094] vouch for URI=www.103.com/101/vouch_
for.htm,
[0095] e-community=sample,
[0096] creation date=Nov 1, 2000,
[0097] extensible data(x)=data(x)}, hash(info)

EX1005, ¶¶93-97 (below row 2, Table 1). In contrast, the eCC cookie type includes the following information different from the DIDC:

[0172] eCC(1104)={Auth Server=104,
[0173] URI at Auth Server=www.106.com/104/
vouch_for.htm, e-community=sample,
[0174] creation date=Nov 1, 2000,
[0175] extensible attribute=value pairs}, hash-
(info)

EX1005, ¶¶172-175 (below row 1, Table 1). As shown in these two excerpts, the cookies contain different information. A POSITA would have understood and found it obvious that the eCC cookie is a session cookie type that is a different type of cookie from the DIDC cookie of the persistent cookie type. EX1002, ¶¶90-92.

During a session subsequent to the session that created the DIDC, when a user requests access to an affiliated domain, the affiliated domain generates an eCC for that domain and causes the eCC cookie to be stored at the user's device by sending the eCC cookie to the user. EX1005, ¶137 (“the vouch-for process occurs when a non-home front-end receives a request from a user that includes a domain identity cookie (DIDC) but not an e-community cookie (ECC) generated by that

front-end”), ¶177 (“affiliated domain plug-in responds to the user’s browser (100) based on the results of the access control decision, including the eCC for the affiliated domain front-end (104)”), ¶178 (“user’s browser (100) receives the response and stores the eCC for the affiliated domain's front-end (104) in its cookie store”). Thus, Hinton’s system discloses saving the eCC cookie in a different session than the DIDC cookie. EX1002, ¶93.

7. [1.b.v]

Hinton discloses and renders obvious *the first cookie of the first type (DIDC cookie) is stored in a first client device browser storage area and the second cookie of the second type (eCC cookie) is stored in a second client device browser storage area different from the first client device browser storage area*. EX1005, ¶¶88, 233, 245. Hinton discloses that the DIDC cookie is stored in the browser’s persistent cookie store while the eCC is stored the browser’s cookie memory, which a POSITA would have understood is different from the browser’s persistent cookie store. EX1005, ¶88 (“the user’s browser extracts (62) the DIDC and stores it in the browser’s persistent cookie store, such as storing it in a cookie folder on a hard disk drive”), ¶233 (“domain identity cookie (“DIDC”) is a persistent cookie that resides in the user's cookie ‘jar’, such as a cookie.txt file”), ¶248 (“the e-community cookie (eCC) ... resides in the user’s browser cookie memory”). As Hinton expressly discloses that the storage areas for the two cookies have different

names, a POSITA would have understood that the two storage areas are different browser storage areas. A POSITA would have understood and found it obvious that because the browser's cookie store is persistent and the browser cookie memory is temporary, they are different storage areas for the browser. EX1002, ¶94.

8. [1.c]

Hinton discloses and renders obvious that the affiliated domain's processor is configured to, *based at least in part of the network resource request from the client device corresponding to the first cookie of the first type caused to be stored at the client device during the first previous network session (limitation 1.b.i), determine information (user's home domain information) that was encoded and stored in the client device.* EX1005, ¶¶137, 143, 155, 236, 239, 245. EX1002, ¶95-97.

Hinton's affiliated domain *determines information* from the DIDC cookie (first cookie stored in previous network session, limitation 1.b.i) in the request. Specifically, when the affiliated domain receives the request from the user to access a resource protected by a plug-in at the affiliated domain, the plug-in “‘parses’ the DIDC(106) [sic] cookie to determine the user's home domain, a URI in their home domain that can vouch for the user's identity, the e-community in which they are enrolled, and a creation/update timestamp.” EX1005, ¶155, ¶236

(“vouch for resource will be a URL that contains some form of active content that can authenticate the user ... and build a ‘vouch for’ token”). When the affiliated domain receives a network resource request from a client device that includes a DIDC, the DIDC corresponds to a DIDC created on the client device in a prior network session. A POSITA would have understood and found it obvious that Hinton discloses the affiliated domain’s processor is configured to perform this function. EX1002, ¶96.

The information in the DIDC (stored in the client device) is hashed, which is a type of encoding of stored data. EX1005, ¶239 (“[t]he information is hashed for integrity protection”), ¶245 (“the domain identity cookie may or may not be protected by keyed hash” which “will at most provide integrity protection on the data in the cookie”). A POSITA would have understood and found obvious that hashing the information in the DIDC, such as through a keyed hash, constituted an encoding of the information. EX1002, ¶97.

9. [1.d.i]

Hinton discloses and renders obvious that the affiliated domain’s processor is configured to *perform a first identification of at least one of the client device and a user of the client device using the first cookie of the first type*. EX1005, ¶92, 137-139, 141-144, 161-162. EX1002, ¶¶98-101.

A POSITA would have understood that the affiliated domain's processor identifies the client device using the DIDC, which is stored in a user's browser's persistent cookie store on a user's device. EX1005, ¶92 (“the user's browser receives a persistent domain identity cookie set by each of the e-community members ... in which the user has successfully been enrolled”), ¶27 (“updated home DIDC is then transmitted to the user's browser, where it is stored in the persistent cookie store”), ¶¶88-89. The affiliated domain uses the DIDC cookie in subsequent sessions to identify that the requesting device has been previously authorized. EX1005, ¶143 (“[a] prerequisite for the transfer of authentication information across domains is that the user has already enrolled in the e-community” and “[i]f there is no DIDC cookie, the front-end will treat the user as a ‘normal’ internal user (as opposed to a participant in the e-community) and will attempt to authenticate the user”). In this manner, Hinton's DIDC is used to identify the client device. As a DIDC cookie is stored in a client device's browser, a POSITA would have understood and found it obvious that the use of the DIDC cookie is the performance of an identification of the client device. EX1005, ¶27. A POSITA would have understood and found it obvious that Hinton discloses the affiliated domain's processor is configured to perform this function. EX1002, ¶99.

The affiliated domain's processor also identifies the user using the DIDC. EX1005, ¶71 (“[t]he purpose of the domain identity cookie is to identify the user's

‘home’ domain, to identify a URL in the user’s home domain that can ‘vouch for’ the user’s identity, and to identify the e-community in which this user is a participant”), ¶137 (“vouch-for process occurs when a non-home front-end receives a request from a user that includes a domain identity cookie (DIDC) but not an e-community cookie (ECC) generated by that front end”). Using the DIDC, the affiliated domain is able to identify the user by requesting a vouch-for token from the user’s home domain. EX1005, ¶139 (the vouch for process includes “[r]equesting ... the user’s home domain to ‘vouch for’ the user”), ¶141 (the vouch for process includes “[g]eneration of a ‘vouch for token’ (VT) to transfer back to the requesting domain a redirected response”), ¶144 (“VT is used to vouch for the authenticity of the user’s identity to the other e-community domains”). The vouch for token includes an indication of the identity of the user in the form of a userID. EX1005, ¶¶161-162 (“first front-end (101) builds a ‘Vouch-For Token’ (VT) to provide the vouch-for information to the affiliated domain (106) such as: VT=E{Tag=VT, userid=jsmith, homedomain=103}”). Thus, because the DIDC is used to generate a vouch-for token, and because the vouch-for token includes an identification of the user, the DIDC is used to perform a first identification of the user. EX1002, ¶100.

A POSITA would have understood and found it obvious that Hinton discloses the affiliated domain's processor is configured to perform this function. EX1002, ¶101.

10. [1.d.ii]

Hinton discloses and renders obvious this limitation. EX1002, ¶¶102-104.

Hinton discloses that *the first identification is performed using the first cookie of the first type*. §IV(C)(9) (limitation 1.d.i). The information in the DIDC is encoded and stored in the client device. *See* §IV(C)(3)-(4), (B)(8) (limitations 1.b.i-1.b.ii, 1.c). EX1002, ¶103.

Hinton further discloses that the first identification is performed *in part by using the determined information that was encoded and stored in the client device* (vouch for URI). EX1005, ¶138-139; §IV(C)(8) (limitation 1.c) (showing vouch for URI is determined information from the DIDC cookie). The DIDC (*i.e.*, first cookie) includes information, such as the vouch for URI, which is used by the affiliated domain for the vouch-for process. *See* §IV(C)(3) (limitation 1.b.i) (DIDC includes “vouch for URI=www.103.com/101/vouch_for.htm”); EX1005, ¶¶138-139 (the vouch for process includes “(1) Identification that user is in the e-community but has a different home domain; (2) Requesting (via re-direction) the user's home domain to ‘vouch for’ the user”). Hinton discloses the DIDC's data is hashed (EX1005, ¶245), a POSITA would have understood the hashing to include

the identification information. Thus, the first identification is performed at least in part using the determined information that was encoded and stored in the client device. A POSITA would have understood and found it obvious that Hinton discloses the affiliated domain's processor is configured to perform this function. EX1002, ¶104.

11. [1.e]

Hinton discloses and renders obvious that the affiliated domain's processor *performs a second identification of at least one of the client device and the user of the client device using the second cookie of the second type (eCC cookie)*. EX1005, ¶¶129, 132, 133, 154, 245. EX1002, ¶¶105-109.

When a user requests to access resources in an affiliated domain, the affiliated domain creates an eCC for the affiliated domain front-end once the user is authenticated and vouched for. EX1005, ¶129 (“[a]s a result of authentication, the SSO plug-in generates an ‘e-Community Cookie’ (an eCC or e-community cookie) that acts as an ‘authenticator bookmark’”), ¶132 (“a user has one e-community cookie set for each domain at which it has a current, authenticated (or vouched-for) session”). Once the eCC is created for the affiliated domain, the user can use the eCC to access any server in the domain. EX1005, ¶132-133 (“[o]nly the one instance with a DNS domain that authenticates the user or first receives an authentication ‘vouch-for’ message sets an e-community cookie at the user’s

browser” and the eCC “is a domain cookie and can therefore be sent to any server in the domain that created it” which “allows for simplified single-sign-on capabilities within a domain that is partitioned by multiple security server domains”). When the user requests access to one of the servers in the domain that created the eCC, that server checks whether the user has an eCC to access the DNS domain it is a part of. EX1005, ¶154 (“plug-in at the associated domain front-end (104) looks for an eCC cookie set by a different front-end within the associated domain (106)” and “[i]f present, this would indicate that the user has a session with a different front-end within the associated domain (106)”). A POSITA would have understood and found it obvious that Hinton’s affiliated domain processor is configured to perform this function. EX1002, ¶106.

The eCC is used to identify the user of the client device. EX1005, ¶245 (“[t]he eCC is used to identify the Web server cluster, within a given domain, that can vouch for a user’s identity”), ¶131 (“eCC identifies the server that authenticated the user and a URI pointing to an authentication script that can vouch for the user within a given domain”). A POSITA would have understood and found it obvious that Hinton’s affiliated domain processor is configured to perform this function. EX1002, ¶107.

A POSITA would have understood that the affiliated domain’s processor identifies the client device using the eCC, which is stored in the browser’s cookie

memory on the user's device. *See* §IV(C)(7) (limitation 1.b.v). Hinton discloses the eCC cookie is stored in a client device's browser's cookie memory. EX1005, ¶167, ¶248. A POSITA would have understood and found it obvious that the use of the eCC cookie is the performance of an identification of the client device. EX1002, ¶108.

A POSITA would have understood and found it obvious that Hinton discloses the affiliated domain's processor is configured to perform this function. EX1002, ¶109.

12. [1.f]

Hinton discloses and renders obvious that the affiliated domain's processor is configured to *perform a determination based at least in part of (1) a presence of a network resource request associated with one of the first cookie and the second cookie, and (2) an absence of a network resource request associated with the other of the first cookie and the second cookie.* EX1005, ¶137. Hinton discloses that the DIDC is generated and stored to the client device based on a network resource request. *See* §IV(C)(4) (limitation 1.b.ii). Hinton discloses that an eCC is generated and stored to the client device based on a network resource request. *See* §IV(C)(6) (limitation 1.b.iv). Hinton further discloses that the vouch-for process is performed based on the presence of the DIDC and the absence of the eCC. EX1005, ¶137 (“vouch-for process occurs when a non-home front-end receives a request from a

user that includes a domain identity cookie (DIDC) but not an e-community cookie (ECC) generated by that front-end”). Thus, Hinton’s vouch-for process makes a determination based on the presence of a network request associated with the first cookie (i.e., the network request that triggered the vouch-for process, which included the DIDC) and associated with an absence of a second cookie (i.e., the user does not have an eCC associated with the affiliated domain). A POSITA would have understood and found it obvious that the affiliated domain’s processor is configured to perform this function. EX1002, ¶110.

13. [1.g]

Hinton discloses *a memory (e.g. RAM) coupled to the one or more processors and configured to provide the one or more processors with instructions.* EX1005, Abstract, claim 10; §IV(C)(2) (limitation 1.a). The affiliated domain (e.g., other affiliated domain 106, another affiliated domain 108) includes a server. EX1005, Abstract (“user’s home domain server prepares and forwards a home domain identity cookie ... with an enrollment request to a user’s browser, with the enrollment request being redirected to an affiliated domain server in the e-community). A POSITA would have understood and found it obvious for the affiliated domain’s server to include a memory coupled to its processor and configured to provide the processor with instructions. *See e.g.*, EX1005, Claim 10 (“A computer readable medium encoded with software for allowing an Internet or

intranet browser user to transfer directly to a domain that is participating in an e-community without repetitious and redundant authentication actions, said e-community comprising a plurality of affiliated domain servers, said user being properly registered and authenticated to a home domain server within said e-community, said software causing a processor to perform the steps ...”). EX1002, ¶111.

D. Claim 2

Hinton discloses and renders this claim obvious. *See* §IV(C) (claim 1).

Hinton discloses that “a user will access resources in different (‘participating’) domains on behalf of their home domain.” EX1005, ¶11. For example, “the home domain itself may have ‘long term’ relationships with other domains.” EX1005, ¶10. Hinton’s invention addresses the “need in the art for a cross-domain single-sign-on system and method which allows an Internet user to establish a long-term relationship with participating domains, and which gives the user the ability to go directly to participating domains, via bookmarks or direct URL’s.” EX1005, ¶15.

In the case where a user requests a network resource from an affiliated domain (i.e., not the home domain that generates the DIDC and the eCC), Hinton discloses that the affiliated domain identifies the user. EX1005, ¶147 (“[v]ouch for information is transferred across domains when a user requests a resource in a

domain other than their home domain, where the request requires an authenticated identity [of the user]”). EX1005, ¶¶84-85, ¶143, ¶45. EX1002, ¶¶112-114.

E. Claim 3

Hinton discloses and renders this claim obvious. *See* §IV(C) (claim 1). The client device is identifiable by a first cookie in a first client device browser storage area. §IV(C)(7), (9) (limitations 1.b.v, 1.d.i), The client device is also identifiable by a second cookie in a second client device browser storage area. §IV(C)(7), (11) (limitations 1.b.v, 1.e). EX1002, ¶¶115-117.

F. Claim 4

Hinton renders this claim obvious. *See* §IV(C) (claim 1). Hinton’s DIDC includes various information about the servers in the e-community, such as the URI corresponding to the home domain for purposes of initiating the vouch-for process. EX1005, ¶¶93-97. Hinton’s eCC also includes the URI corresponding to the affiliated domain that issued the eCC. A POSITA would have been familiar with the problem of domain spoofing, or pharming (EX1015) and been motivated to add detection of pharming to Hinton. In order to prevent possible domain spoofing, a POSITA would have been motivated by server load balancing to encode a server identifier, such as an Internet Protocol (IP) address, for the home domain as an additional field in the DIDC and/or a server identifier, such as an IP address, of the affiliated domain as an additional field in the eCC. *See, e.g.,*

EX1013. In this manner, a POSITA would have understood and found obvious that the affiliated domain's processor would use the home domain server's IP address or the affiliated domain server's IP address as an additional check to determine whether a malicious actor spoofed the home domain server's URI or the affiliated domain server's URI, respectively, i.e., perform a detection of pharming, based on the received DIDC or eCC that includes the URI and the IP address of each respective server. EX1002, ¶¶118-120.

G. Claim 5

Hinton discloses and renders this claim obvious. *See* §IV(C) (claim 1). In the case where a user requests a network resource from an affiliated domain for which an eCC was generated and then deleted (*i.e.*, the DIDC is present, but not the eCC), the affiliated domain's processor generates and sends to the client device a new eCC corresponding to its domain, which is a third cookie of the second type. EX1005, ¶45 (“an ‘e-community’ has many different ‘participants’, including e-community members, or domains corresponding to the business units that are participating in the e-community”), ¶143 (“[a] prerequisite for the transfer of authentication information across domains is that the user has already enrolled in the e-community”), ¶147 (“[v]ouch for information is transferred across domains when a user requests a resource in a domain other than their home domain, where the request requires an authenticated identity”), ¶249 (“[a]n eCC is valid for only

for the duration of a browser session, and it is expired when a user invokes logout functionality”). Thus, the replacement token is a third cookie of one of the second type. EX1002, ¶¶121-123.

H. Claim 6

Hinton discloses and renders claim 6 obvious. *See* §IV(C) (claim 1). EX1002, ¶¶124-135.

1. [6.pre]

Hinton discloses and renders obvious the preamble. *See* §IV(C)(1) (limitation 1.pre). EX1002, ¶125.

2. [6.a.i]

Hinton discloses and renders obvious this limitation. *See* §IV(C)(3) (limitation 1.b.i). EX1002, ¶126.

3. [6.a.ii]

Hinton discloses and renders obvious this limitation. *See* §IV(C)(4) (limitation 1.b.ii). EX1002, ¶127.

4. [6.a.iii]

Hinton discloses and renders obvious this limitation. *See* §IV(C)(5) (limitation 1.b.iii). EX1002, ¶128.

5. [6.a.iv]

Hinton discloses and renders obvious this limitation. *See* §IV(C)(6) (limitation 1.b.iv). EX1002, ¶129.

6. [6.a.v]

Hinton discloses and renders obvious this limitation. *See* §IV(C)(7)
(limitation 1.b.v). EX1002, ¶130.

7. [6.b]

Hinton discloses and renders obvious this limitation. *See* §IV(C)(8)
(limitation 1.c). EX1002, ¶131.

8. [6.c.i]

Hinton discloses and renders obvious this limitation. *See* §IV(C)(9)
(limitation 1.d.i). EX1002, ¶132.

9. [6.c.ii]

Hinton discloses and renders obvious this limitation. *See* §IV(C)(10)
(limitation 1.d.ii). EX1002, ¶133.

10. [6.d]

Hinton discloses and renders obvious this limitation. *See* §IV(C)(11)
(limitation 1.e). EX1002, ¶134.

11. [6.e]

Hinton discloses and renders obvious this limitation. *See* §IV(C)(12)
(limitation 1.f). EX1002, ¶135.

I. Claim 7

Hinton discloses and renders obvious this claim. *See* §§IV(D), IV(H) (claims
2, 6). EX1002, ¶136.

J. Claim 8

Hinton discloses and renders obvious this claim. *See* §§IV(E), IV(H) (claims 3, 6). EX1002, ¶137.

K. Claim 9

Hinton discloses and renders obvious this claim. *See* §§IV(F), IV(H) (claims 4, 6). EX1002, ¶138.

L. Claim 10

Hinton discloses and renders obvious this claim. *See* §§IV(G), IV(H) (claims 5-6). EX1002, ¶139.

If Patent Owner argues the portions of Hinton cited above relate to different, incompatible embodiments (which they do not), it would have been obvious to a POSITA to combine such embodiments into a single system for client-server communications at least because such embodiments are described in the same prior art reference, are fully compatible with each other, and could be combined with minimal effort to achieve predictable results. EX1002, ¶140.

If the Board finds that Hinton does not disclose any limitation of the challenged claims, such limitation would have nonetheless been obvious to a POSITA in light of Hinton and a POSITA's knowledge. Practicing any limitation of the challenged claims in light of Hinton would have been within the knowledge and skill of a POSITA, would have required minimal effort, would have yielded

predictable results, would have been fully compatible with the Hinton system, and would have been a mere design choice. Motivation to do so arises from at least common sense and the disclosures of Hinton set forth above. EX1002, ¶141.

V. GROUND 2: VARGHESE RENDERS OBVIOUS CLAIMS 1, 3-6, AND 8-10

A. Overview of Varghese

Varghese was filed April 28, 2006, issued as U.S. Patent No. 7,908,645 on March 15, 2011, and identifies Oracle International Corporation as its assignee. EX1004, cover page. Varghese is prior art under 35 U.S.C. §102(e). EX1002, ¶142.

Varghese is generally directed to “systems and methods for providing protection against identify theft of a computer network.” EX1004, 1:16-18. To protect computer networks from theft, Varghese discloses the use of two cookies – secure cookies and flash cookies. EX1004, 5:64-66 (“[t]he present invention includes secure cookies, flash objects and other technologies to recognize and to fingerprint [] from which device a user access[sic] an application”), 6:2-7 (“[i]nformation concerning these user devices is fingerprinted and stored into a device token or device id for one-time use,” which “is stored on the user device and saved in a database for later comparison with tokens retrieved from subsequent user device accesses”), 6:8-14 (“[t]he present invention also includes user devices tokens or device ids that have a unique number which is randomly generated by

methods of this invention,” which “are ... assigned to the particular user device, stored on the particular user device as persistent data (e.g., a cookie), and also stored so as to be accessible to the authentication services of this invention”).

EX1002, ¶143.

The flash cookie and/or the secure cookie are used to identify the user device on subsequent login or authentication attempts. EX1004, 24:23-27 (“[u]ser devices are preferably identified using secure cookies, Flash cookies, and similar data tokens combined with other data items such as browser characteristics, device hardware configuration (e.g., as acquired using Flash calls), network characteristics, and the like”). Secure cookies are essentially standard cookies that are secured against modification or tampering. EX1004, 25:25-32 (“[a] standard cookie is a data packet sent by a web server to a web browser for saving to a file on the host machine” and “[a] secure cookie refers to a standard cookie that has been secured against modification or tampering”). Flash cookies differ from secure cookies in that they are not as easily removed from the user’s device because they are generated and stored by the Flash software on the user device. EX1004, 25:37-43 (“[Flash] software can create local shared objects, known as ‘flash cookies’, for maintaining locally persistent data on a user’s device akin to the standard ‘cookies’ stored by web browsers” and “have the advantage not being as easily removed from the user’s device as are standard cookies”). Varghese’s server uses both the

flash cookie and the secure cookie to verify the user. For example, Varghese discloses in Table 8 the various scenarios based on the presence or absence of either of the secure cookie or the flash cookie. EX1004, 19:54- 65 (“This table returns a score of “0” (a score indicated a low likelihood of fraud) in case all evaluated data items are present and match in connection with a current user request. If no data item is present or if all data items do not match, a score of “10” (a score indicated a high likelihood of fraud) is returned. In case where some data items are present and match while other data items are absent or do not match, this table invokes further checks. If the retrieved data tokens that were previously stored on a device by this invention, e.g., a secure cookie, a Flash cookie, or Flash data, are not present, a further pattern check is performed.”).

TABLE 8

Primary device decision table					
Data item					
Secure cookie	Flash cookie	Flash data	Browser characteristics	Operating system characteristics	Score
*	*	*	*	*	0
X	*	*	*	*	PATTERN CHECK
M	*	*	*	*	SECONDARY CHECK
X/M	X	*	*	*	PATTERN CHECK
X/M	M	*	*	*	SECONDARY CHECK
X/M	X/M	X	*	*	PATTERN CHECK
X/M	X/M	M	*	*	SECONDARY CHECK
X/M	X/M	X/M	M	*	SECONDARY CHECK
X/M	X/M	X/M	X/M	M	10

Key:
 X = missing;
 M = present and mismatched;
 * = present and matched

EX1004, Table 8 [annotation added]. EX1002, ¶¶144-145.

Varghese is analogous art to the '823 patent and was not considered during prosecution of the '823 patent. EX1002, ¶¶146-147.

B. Motivation to Combine

As shown below, no combinations are required to arrive at the claimed invention. Therefore, no showing of a motivation to combine or reasonable expectation of success is required. *See Unification Techs.*, 2024 WL 3738401, at *6. EX1002, ¶148.

C. Claim 1

As shown below, Varghese renders claim 1 obvious. EX1002, ¶¶149-185.

1. [1.pre]

Varghese discloses *a system* (collectively, system 1302, system 1304, and authentication server 1306). EX1004, 8:40-44 (“FIG. 13A illustrates an exemplary embodiment of the present invention directed to providing authentication services to online service providers who make available to individual users online server applications” that “execute on service-provider machines”), 8:46-48 (Fig. 13A illustrates ... one or more service-provider computer systems, e.g., systems 1302 and 1304 and an authentication system server system 1306, interconnected through network 710 to one or more user devices 720), 8:60-9:3 (“In many preferred embodiments ... authentication processes of the invention are implemented with a client-server-type architecture (or, more generally, a distributed-systems-type architecture.”).

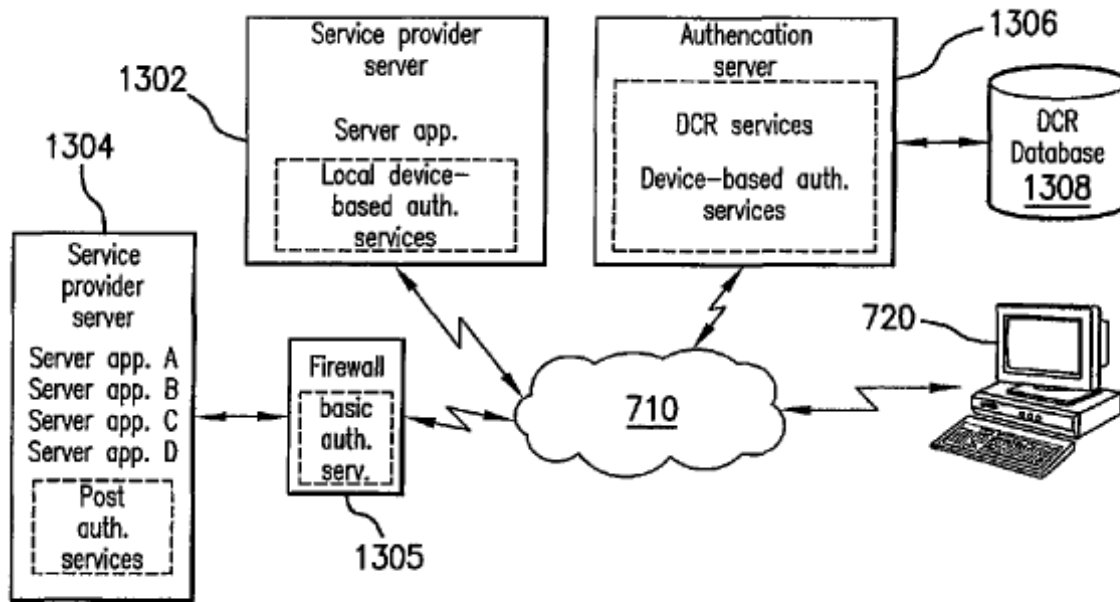


FIG. 13A

EX1004, Fig. 13A. EX1002, ¶¶150-151.

2. [1.a]

Varghese discloses and renders obvious *one or more processors* (processor associated with authentication server 1306). Varghese discloses that authentication server 1306 “is generally structured as known in the art, and includes a CPU.”

EX1004, 8:50-51. EX1002, ¶¶152-153.

3. [1.b.i]

Varghese discloses and renders obvious the authentication server processor configured to *receive a network resource request from a client device*. Varghese discloses that server 1306 receives network resource requests from a client device.

EX1004, Fig. 13A, 6:21-23 (“[t]he present invention enables application service providers score risk for each online login and transaction and to increase

authentication security in real time”), 10:17-19 (“[a]uthentication services are invoked when a server application or services provider computer system receives user request 1320 that needs authentication”), 10:20-21 (“the most common user request is a login request to access an application or system”), 24:40-41 (“request is received at a service provider server from a user device 720.”), 10:20-23, (“user request is a login request to access an application or system ... transaction requests.”). A POSITA would have understood and found it obvious that authentication server’s processor is configured to receive those requests. EX1002, ¶¶154-155.

4. [1.b.ii]

Varghese discloses and renders obvious this limitation. EX1002, ¶¶156-161.

Varghese discloses the *network resource request* (authentication request) *corresponds to a first cookie of a first type* (secure cookie). For example, when a user in Varghese attempts to authenticate themselves, their browser provides the authentication server with the secure cookie that was generated during a previous session with the server. EX1004, 24:40-41 (“[i]n Step 402, a request is received at a service provider server from a user device 720”), 15:31-34 (“[a] further important component of device information when available is a secure token, e.g., a secure cookie, available from a device which has been previously used as a user device”), 15:34-40 (“[w]hen a request is received from and device, at least the available

location and device information can be summarized, condensed, or fingerprinted and stored back on the device as a secure token” and “[i]f another request then originates from this device, the secure token can be retrieved and its contents compared against the currently-collected location and device information.”), 2:4-8, (“[a] cookie generally refers to a packet of information, often sensitive information, sent by a web server to a browser resident on the user’s computer system for saving to a file and for transmitting back to the server whenever the user’s browser makes additional requests from the server”). EX1002, ¶157.

Varghese discloses that the first cookie of the first type *was caused to be stored to the client device*. Varghese discloses that the client device receives and stores the secure cookie that was generated by the server. EX1004, 25:25-32, 25:25-29 (“[a] standard cookie is a data packet sent by a web server to a web browser for saving to a file on the host machine,” which “can be retrieved and submitted back to the server when requested”), EX1004, 26:4-6 (“a new Device ID token is created for the device ... and ... sent to the user device and stored thereon, e.g., as a standard cookie or as a flash cookie”), EX1004 26:6-11 (“[i]f no Device ID was found on the user device, a new Device ID token is created from the gathered identifying information” and “[i]f a Device ID was found, it can be updated, e.g., with a new unique bit string, new timestamp, and so forth”), 24:67-25:3 (“[s]ome or all of the device information (along with identifying information

generated during the fingerprinting process) information is stored in a data token referred to as a ‘Device ID’”). EX1002, ¶158.

Varghese further discloses that the first cookie was caused to be stored to the client device *at least in part by causing the client device to initiate a set of network resource requests*. Varghese discloses that the secure token was generated during a login attempt by the user. EX1004, 24:40-42 (“a request is received at a service provider server from a user device 720 ... for data resident thereon”), 26:13-14 (“[a] feature of the invention relates to the replacement of the cookie on the user’s machine upon each login”), 10:17-19 (“[a]uthentication services are invoked when a server application or services provider computer system receives user request 1320 that needs authentication”). A POSITA would have understood that a login prompt is a network request that the server causes the user to initiate a set of network resource requests. Varghese teaches details of authentication along with illustrating user interfaces for doing so with a preferred embodiment making use of Flash software being sent to the user device, which a POSITA would recognize as requiring a subsequent request for a Flash object (*i.e.*, a network resource request). EX1004, 29:42-58. EX1002, ¶159-160.

That set of network requests were *determined during the first previous network session*. As provided above, the secure token is generated upon an authentication request and then used by the user in subsequent network sessions.

See e.g., EX1004, 15:31-34 (“[a] further important component of device information when available is a secure token, e.g., a secure cookie, available from a device which has been previously used as a user device”). Thus, Varghese discloses that the secured cookie was generated and stored during a network session that preceded the network resource request. EX1002, ¶161.

5. [1.b.iii]

Varghese further discloses *the client device initiating the set of network resource requests caused data representative of the set of network resource requests to be stored at the client device*. Varghese discloses that the secure cookie includes information representative of the login request. EX1004, 10:53-60 (“fingerprinting process then gathers identifying information describing the device from which the user request originated and creates a device identifier (‘Device ID’)” that “is stored on the user device from which it can be retrieved and form part of the device identifying information to be used during a subsequent fingerprinting”), 24:50-51 (during the fingerprinting process “device identity information for the user device is captured”), 24:67-25:3 (“[s]ome or all of the device identity (along with identifying information generated by the fingerprinting process) information is stored in a data token referred to as a ‘Device ID’”). The client device caused this information to be stored at the client device when the client device received the secured cookie for storage. EX1004, 6:2-4

("[i]nformation concerning these user devices is fingerprinted and stored into a device token or device id for one-time use"), 6:4-7 ("[t]he id or token is stored on the user device and saved in a database for later comparison with tokens retrieved from subsequent user device accesses"). EX1002, ¶¶162-163.

6. [1.b.iv]

Varghese discloses and renders obvious this limitation. EX1002, ¶¶164-166.

Varghese discloses *a second cookie of a second type different from the first type*. Varghese discloses use of a flash token. EX1004, 25:33-40. Flash cookies are locally shared objects created by flash software and are thus of a second type different from the first type "secure" cookies. EX1004, 25:40-43 ("[f]lash cookies can be stored locally on a flash plug-in user's device, are updatable, and have the advantage not being as easily removed from the user's device as are standard cookies"), 25:25-32. EX1002, ¶165.

Varghese discloses that the flash cookie *was caused to be stored at the client device during a second previous network session*. A POSITA would recognize that Flash software sent to the user device for use with the authentication interface would have caused the software to store Flash cookies at the client device during a second previous network session. EX1004, 29:51-54. Varghese further discloses that secure cookies can be removed from the browser's memory when the user clears the browser's cookies. EX1004, 25:40-43 ("[f]lash cookies ... have the

advantage not being as easily removed from the user's device as are standard cookies"), 25:25-32. Varghese further discloses that the cookies are routinely replaced with each login (*i.e.*, new network session). EX1004, 26:13-14 ("[a] feature of the invention relates to the replacement of the cookie on the user's machine upon each login"). Thus, a POSITA would have understood and found obvious that Varghese discloses a scenario in which the flash cookie was created and stored in a second previous session than the secure cookie (*e.g.*, the user cleared the browser's cookies). Further Varghese contemplates the cookies being created in different network sessions when only one of the secured cookie or the flash cookie is present. *See, e.g.*, EX1004, Table 8 (showing various scenarios in which one of secure cookie present or flash cookie is present).

TABLE 8

Primary device decision table					
Data item					
Secure cookie	Flash cookie	Flash data	Browser characteristics	Operating system characteristics	Score
*	*	*	*	*	0
X	*	*	*	*	PATTERN CHECK
M	*	*	*	*	SECONDARY CHECK
X/M	X	*	*	*	PATTERN CHECK
X/M	M	*	*	*	SECONDARY CHECK
X/M	X/M	X	*	*	PATTERN CHECK
X/M	X/M	M	*	*	SECONDARY CHECK
X/M	X/M	X/M	M	*	SECONDARY CHECK
X/M	X/M	X/M	X/M	M	10

Key:
 X = missing;
 M = present and mismatched;
 * = present and matched

EX1002, ¶166.

7. [1.b.v]

Varghese discloses and renders obvious *the first cookie of the first type (secure cookie) is stored in a first client device browser storage area and the second cookie of the second type (flash cookie) is stored in a second client device browser storage area different from the first client device browser storage area. A POSITA would have understood and found obvious to store the secure cookie in a first client device browser storage area different from the first client device browser storage area where the flash cookie is stored, because it was obvious to a*

POSITA to organize different types of files in different folders. EX1004, 25:25-27

("[a] standard cookie is a data packet sent by a web server to a web browser for saving to a file on the host machine."), 25:25-32, 25:40-43 ("Flash cookies can be stored locally on a flash plug-in user's device, are updatable, and have the advantage not being as easily removed from the user's device as are standard cookies."); §IV(C)(7) (Hinton limitation 1.b.v). EX1002, ¶¶167-168.

8. [1.c]

Varghese discloses and renders obvious the authentication server's processor configured to, *based at least in part on the network resource request from the client device corresponding to the first cookie of the first type caused to be stored at the client device during the first previous network session (see §V(C)(3)-(4) (limitations 1.b.i-1.b.ii), determine information that was encoded and stored in the client device.* Varghese discloses that the "captured device information," such as the Device ID that is stored as a secure cookie includes persistent data. EX1004, 6:8-14 ("[t]he present invention also includes user device tokens or device ids that have a unique number which is randomly generated by the methods of this invention" and "are then assigned to the particular user device, stored on the particular user device as persistent data (e.g., a cookie), and also stored so as to be accessible to the authentication services of this invention"). Varghese further discloses that the persistent data includes information that are encrypted to secure

against modification. EX1004, 25:10-15 (“[s]ecure persistent data includes generally data elements that are encrypted, signed, or otherwise secured against modification, and remain resident on the user device even when it is not accessing a service provider application”). This persistent data is used by the server to identify the user on subsequent login. EX1004, 25:54-56 (“the captured device identity information (ID), including any previously stored Device ID, is compared to identity information that has previously been stored”), 25:7-14 (“the captured device identifying information includes ... [a] first type of device identifying information [that] is a secure, persistent data token that has been previously stored on the user device” and “includes ... data elements that are encrypted, signed, or otherwise secured against modification, and that remain resident on the user device even when it is not accessing a service provider application”). Thus, Varghese discloses that the server identifies information that was previously encrypted or encoded when the server extracts the device identity information in a subsequent authentication attempt. A POSITA would have understood and found it obvious that Varghese discloses the authentication server’s processor is configured to perform this function. EX1002, ¶¶169-171.

9. [1.d.i] and [1.d.ii]

Varghese discloses and renders obvious this limitation. EX1002, ¶¶172-175.

Varghese discloses and renders obvious the authentication server's processor *performs a first identification of ... the client device ... using the first cookie of the first type ... wherein the first identification is performed using the first cookie of the first type at least in part by using the determined information that was encoded and stored in the client device.* Varghese discloses that the secure cookie is used to identify the client device on subsequent log-in attempts. EX1004, 25:7-14 (“the captured device identifying information includes ... [a] first type of device identifying information [that] is a secure, persistent data token that has been previously stored on the user device” and “includes ... data elements that are encrypted, signed, or otherwise secured against modification, and that remain resident on the user device even when it is not accessing a service provider application”), 25:55-56 (“the captured device identity information (ID), including any previously stored Device ID, is compared to identity information that has previously been stored”), 26:4-6 (“a new Device ID token is created for the device ... is sent to the user device and stored thereon ... as a standard cookie”), 25:25-32, 26:9-12 (“[i]f a Device ID was found, it can be updated, e.g., with a new unique bit string, new timestamp, and so forth”), Table 3, 14:67-15:3. The Device ID includes information that can identify the client device, such as, but not limited to “IP addresses, adapter MAC addresses, local time and/or time zone, network connection speed such as download and/or upload times, microprocessor type

and/or processing and/or serial number, and so forth.” EX1004, 25:48-51. EX1002,
¶173.

Varghese further discloses that the processors *perform a first identification of ... a user of the client device ... using the first cookie of the first type ... wherein the first identification is performed using the first cookie of the first type at least in part by using the determined information that was encoded and stored in the client device.* As shown in Table 3, Varghese discloses that various information is stored in association with the secure cookie, such as post-authentication information, which includes “user identifications.” EX1004, 14:67-15:3 (“Table 3 present a more detailed catalog of device software and hardware characteristics that can be extracted from a device by a browser-hosted process”).

TABLE 3

Example of Request Attributes			
Pre-authentication	Location information	City, State, Country information and confidence factors	
		Connection type	
		Connection speed	
		IP address, routing type, and hop times	
		Internet service provider flag	
		Autonomous system number	
		Carrier name	
		Top-level domain	
		Second-level domain	
		Registering organization	
		A list of anonymizing proxies	
		Hostnames and routers	
		Device information	Secure Cookies
			Flash Cookies
Post-authentication	User information	Digitally signed device	
		Device & display Characteristics:	
		Operating System characteristics	
		Device Characteristics	
		User identifications	
		Valid or not valid user	
		Authentication status	
	Transaction information	Key Value Pairs:	
		Support multiples	
		Keys can be defined using Regular Expressions	
	Values can be defined in ranges		
	Pages accessed		
	Time spent on page		
	Transactions sequences		

EX1004, Table 3 (annotation added). A POSITA would have understood and found it obvious that the encoded information in the secure cookie would be used for the identification of the user because the secure cookie was created as part of the authentication process and Table 3 discloses that after the authentication process, in post-authentication, the user identification is available. EX1002, ¶174.

A POSITA would have understood and found it obvious that Varghese discloses the authentication server's processor is configured to perform this function. EX1002, ¶175.

10. [1.e]

Varghese discloses and renders obvious this limitation. EX1002, ¶¶176-179.

Varghese discloses and renders obvious the authentication server's processor *perform a second identification of ... the client device ... using the second cookie of the second type*. Varghese discloses that the flash cookie is used to identify the client device on subsequent log-in attempts using a Device ID. EX1004, 25:7-14 (“the captured device identifying information includes ... [a] first type of device identifying information [that] is a secure, persistent data token that has been previously stored on the user device” and “includes ... data elements that are encrypted, signed, or otherwise secured against modification, and that remain resident on the user device even when it is not accessing a service provider application”), 25:54-56 (“the captured device identity information (ID), including any previously stored Device ID, is compared to identity information that has previously been stored”), 26:4-6 (“a new Device ID token is created for the device ... is sent to the user device and stored thereon ... as a flash cookie”), 26:8-10 (“[i]f a Device ID was found, it can be updated, e.g., with a new unique bit string, new timestamp, and so forth”). The Device ID includes information that can identify the client device, such as, but not limited to “IP addresses, adapter MAC addresses, local time and/or time zone, network connection speed such as download and/or upload times, microprocessor type and/or processing and/or serial number, and so forth.” EX1004, 25:48-51. EX1002, ¶177.

Varghese further discloses that the processors *perform a second identification of ... a user of the client device ... using the second cookie of the second type*. As shown in Table 3, Varghese discloses that various information is stored in association with the flash cookie, such as post-authentication information, which includes “user identifications.” EX1004, 14:67-15:3 (“Table 3 present a more detailed catalog of device software and hardware characteristics that can be extracted from a device by a browser-hosted process”).

TABLE 3

Example of Request Attributes			
Pre-authentication	Location information	City, State, Country information and confidence factors	
		Connection type	
		Connection speed	
		IP address, routing type, and hop times	
		Internet service provider flag	
		Autonomous system number	
		Carrier name	
		Top-level domain	
		Second-level domain	
		Registering organization	
		A list of anonymizing proxies	
		Hostnames and routers	
		Device information	Secure Cookies
			Flash Cookies
Post-authentication	User information	Digitally signed device	
		Device & display Characteristics:	
		Operating System characteristics	
		Device Characteristics	
		User identifications	
		Valid or not valid user	
		Authentication status	
		Transaction information	Key Value Pairs:
			Support multiples
			Keys can be defined using Regular Expressions
	Values can be defined in ranges		
	Pages accessed		
	Time spent on page		
	Transactions sequences		

EX1004, Table 3 (annotation added). A POSITA would have understood and found it obvious that the encoded information in the flash cookie would be used for the identification of the user because the flash cookie was created as part of the

authentication process and Table 3 discloses that after the authentication process, in post-authentication, the user identification is available. EX1002, ¶178.

A POSITA would have understood and found it obvious that Varghese discloses the authentication server's processor is configured to perform this function. EX1002, ¶179.

11. [1.f]

Varghese discloses and renders obvious this limitation. EX1002, ¶¶180-183.

Varghese discloses and renders obvious the authentication server's processor configured to *perform a determination based at least in part on (1) a presence of a network resource request associated with one of the first cookie and the second cookie, and (2) an absence of a network resource request associated with the other of the first cookie and the second cookie.* A POSITA would have understood that a network resource request disclosed in Varghese is an authentication request.

§V(C)(4) (limitation 1.b.ii). Varghese discloses that the security server utilizes a security model for determining whether a request presents a security problem.

EX1004, 19:25-28. Table 8 (reproduced and annotated below) illustrates the security model which instructs the security server how to act depending on the presence or absence of either of the secure cookie or the flash cookie. For example, if both the secure cookie and the flash cookie are present, a first determination or score is returned. EX1004, 19:54-56 (“[t]his table returns a score of “0” (a score

indicated a low likelihood of fraud) in case all evaluated data items are present and match in connection with a current user request”). In another example, if not all of the data items are present, *e.g.*, the secure cookie is not present but the flash cookie is present, a second determination or score is returned. EX1004, 19:58-65 (“[i]f no data item is present or if all data items do not match, a score of ‘10’ (a score indicated a high likelihood of fraud) is returned ... [i]n case where some data items are present and match while other data items are absent or do not match, this table invokes further checks ... [i]f the retrieved data tokens that were previously stored on a device by this invention, *e.g.*, a secure cookie, a Flash cookie, or Flash data, are not present, a further pattern check is performed.”). If the secure cookie is not present but the flash cookie is present, a POSITA would have understood this to mean that the network resource request associated with the secure cookie was not available but that the network resource request associated with the flash cookie was available.

TABLE 8

Primary device decision table					
Data item					
Secure cookie	Flash cookie	Flash data	Browser characteristics	Operating system characteristics	Score
*	*	*	*	*	0
X	*	*	*	*	PATTERN CHECK
M	*	*	*	*	SECONDARY CHECK
X/M	X	*	*	*	PATTERN CHECK
X/M	M	*	*	*	SECONDARY CHECK
X/M	X/M	X	*	*	PATTERN CHECK
X/M	X/M	M	*	*	SECONDARY CHECK
X/M	X/M	X/M	M	*	SECONDARY CHECK
X/M	X/M	X/M	X/M	M	10

Key:
 X = missing;
 M = present and mismatched;
 * = present and matched

EX1004, Table 8 [annotation added].

Secure cookie is enabled
 and flash is disabled
 Secure cookie is disabled
 and flash is enabled

Only secure cookie came
 through successfully.
 Only flash cookie came
 through successfully.

EX1004, Appendix A (excerpted). Thus, the server makes a pattern check determination based on receipt of a network resource request that includes a flash cookie but not a secure cookie. EX1002, ¶181.

Varghese also expressly shows that a scenario where a secure cookie (first cookie) is “out of sync” and the flash cookie (second cookie) “is in sync.” EX1004, Appendix A. A POSITA would have understood this to mean that a network

resource request that includes an out of sync secure cookie is not a valid resource request and that, therefore, the server does not receive a network resource request with a valid or in sync secure cookie. Similarly, a POSITA would have understood that a network resource request that includes an in sync flash cookie is a valid resource request and that, therefore, the server identifies a presence of the flash cookie.

Secure cookie out of
sync and flash is in
sync.
Flash cookie out of
sync and secure cookie
is sync.

Id (excerpted). Thus, Varghese discloses another determination in which a valid network resource requests for one of the secure cookie and the flash cookie is present and the other is not present. EX1002, ¶182.

A POSITA would have understood and found it obvious that Varghese discloses the authentication server's processor is configured to perform this function. EX1002, ¶183.

12. [1.g]

Varghese discloses *a memory coupled to the one or more processors and configured to provide the one or more processors with instructions*. Varghese's systems server 1306 "includes a CPU, RAM memory, disc or other database

memory 1308, communication interfaces, optional user interface equipment, and the like.” EX1004, 8:51-53. Further, a POSITA would have understood that system server’s 1306 memory would need to include instructions that are provided to the authentication server’s processor otherwise the processor would not know how to perform the generation and validation steps described above. EX1002, ¶¶184-185.

D. Claim 3

Varghese discloses and renders claim 3 obvious. *See* §V(C) (claim 1). The client device is identifiable by a first cookie in a first client device browser storage area. §V(C)(7), (9) (limitations 1.b.v, 1.d.i), The client device is also identifiable by a second cookie in a second client device browser storage area. §V(C)(7), (11) (limitations 1.b.v, 1.e). EX1002, ¶¶186-188.

E. Claim 4

Varghese renders claim 4 obvious. *See* §V(C) (claim 1). A POSITA would have been familiar with the problem of domain spoofing, or pharming (EX1015), and been motivated to add detection of pharming to Varghese. A POSITA would have been motivated by server load balancing to implement the Flash cookie at the client device to include a server identifier, such as an IP address, for the authentication server in order to prevent possible domain spoofing. *See e.g.*, EX1013. In this manner, when the user attempts to use the Flash cookie to authenticate themselves at the authentication server, the authentication server

verifies whether the cookie includes both the URL of the authentication server and the IP address of the authentication server, as an additional check to determine whether a malicious actor spoofed its URL, *i.e.*, perform a detection of pharming, based on the Flash cookie. If the authentication server determines that the URL contained in the cookie is correct but the IP address contained in the cookie is incorrect, then it has detected pharming. EX1002, ¶¶189-191.

F. Claim 5

Varghese discloses and renders claim 5 obvious. *See* §V(C) (claim 1). In the case where the determination is performed responsive to a subsequent login request, Varghese discloses that one of the secure token or the flash token is replaced by the authentication server upon each login. EX1004, 26:13-14 (“[a] feature of the invention relates to the replacement of the cookie on the user’s machine upon each login.”). Thus, the replacement token is a third cookie of one of the first or second type. EX1002, ¶¶192-194.

G. Claim 6

Varghese discloses and renders claim 6 obvious. *See* §V(C) (claim 1). EX1002, ¶¶195-205.

1. [6.pre]

Varghese discloses and renders obvious the preamble. *See* §V(C) (claim 1). EX1002, ¶196.

2. [6.a.i]

Varghese discloses and renders obvious this limitation. *See* §V(C)(3)
(limitation 1.b.i). EX1002, ¶197.

3. [6.a.ii]

Varghese discloses and renders obvious this limitation. *See* §V(C)(4)
(limitation 1.b.ii). EX1002, ¶198.

4. [6.a.iii]

Varghese discloses and renders obvious this limitation. *See* §V(C)(5)
(limitation 1.b.iii). EX1002, ¶199.

5. [6.a.iv]

Varghese discloses and renders obvious this limitation. *See* §V(C)(6)
(limitation 1.b.iv). EX1002, ¶200.

6. [6.a.v]

Varghese discloses and renders obvious this limitation. *See* §V(C)(7)
(limitation 1.b.v). EX1002, ¶201.

7. [6.b]

Varghese discloses and renders obvious this limitation. *See* §V(C)(8)
(limitation 1.c). EX1002, ¶202.

8. [6.c.i] and [6.c.ii]

Varghese discloses and renders obvious this limitation. *See* §V(C)(9)
(limitations 1.d.i-1.d.ii). EX1002, ¶203.

9. [6.d]

Varghese discloses and renders obvious this limitation. *See* §V(C)(10) (limitation 1.e). EX1002, ¶204.

10. [6.e]

Varghese discloses and renders obvious this limitation. *See* §V(C)(11) (limitation 1.f). EX1002, ¶205.

H. Claim 8

Varghese discloses and renders obvious this claim. *See* §§V(D), (G) (claims 3, 6). EX1002, ¶206.

I. Claim 9

Varghese discloses and renders obvious this claim. *See* §§V(E), (G) (claims 4, 6). EX1002, ¶207.

J. Claim 10

Varghese discloses and renders obvious this claim. *See* §§V(F)-(G) (claims 5-6). EX1002, ¶208.

If Patent Owner argues the portions of Varghese cited above relate to different, incompatible embodiments (which they do not), it would have been obvious to a POSITA to combine such embodiments into a single system for client-server communications at least because such embodiments are described in

the same prior art reference, are fully compatible with each other, and could be combined with minimal effort to achieve predictable results. EX1002, ¶209.

If the Board finds that Varghese does not disclose any limitation of the challenged claims, such limitation would have nonetheless been obvious to a POSITA in light of Varghese and a POSITA's knowledge. Practicing any limitation of the challenged claims in light of Varghese would have been within the knowledge and skill of a POSITA, would have required minimal effort, would have yielded predictable results, would have been fully compatible with the Varghese system, and would have been a mere design choice. Motivation to do so arises from at least common sense and the disclosures of Varghese set forth above. EX1002, ¶210.

VI. GROUND 3: VARGHESE IN VIEW OF HINTON RENDERS OBVIOUS CLAIMS 2 AND 7

A. Overview of Varghese

§V(A). EX1002, ¶211.

B. Overview of Hinton

§IV(A). EX1002, ¶212.

C. Motivation to Combine

A POSITA would have been motivated to modify the teachings of Varghese with Hinton because Varghese teaches an authentication system for providing protection against identity theft over a computer network and Hinton teaches a

means for authenticating users across systems situated in different computer domains. *See* §§IV(A), V(A). EX1002, ¶213.

A POSITA would have been motivated to modify Varghese's computing environment such that Varghese's systems (e.g., systems 1302, 1304, 1306) can be distributed across different domains. Such a combination involves a combination of prior art elements (e.g., combining Hinton's distributed system with Varghese's authentication system) according to known methods to yield predictable results (e.g., such as in the situation where Varghese's authentication process it utilized in a business-to-business environment). EX1005, ¶10. A POSITA would have been motivated by the teachings of Hinton to modify Varghese such that system 1302 is in a different domain than system 1306, such as, for example, in a business-to-business use case. Moreover, in the timeframe, a POSITA would have known that a single organization was known to use multiple domains with multiple servers for providing content. EX1012. Thus, it was well-known that a business would need to support multiple domains, so a POSITA would have been motivated to modify Varghese to do so (as taught by Hinton). Therefore, it was obvious to a POSITA that the provision of cookies in a single domain would be used by another domain. Therefore, a POSITA would have been motivated to make this combination. EX1002, ¶214.

A POSITA would have had a reasonable expectation of success in making the proposed combination given the teachings of Varghese and Hinton. Such a combination would have required minimal modifications to Varghese, as the use of cookies to identify users when interacting with servers (regardless of the domain) is well known in the art and would have been well within the skillset of a POSITA. The proposed combination would not have required undue experimentation and would have yielded the predictable result. EX1002, ¶215.

D. Claim 2

Varghese in view of Hinton discloses and renders this claim obvious. *See* §V(C) (claim 1), §IV(D) (claim 2). EX1002, ¶¶216-218.

Varghese discloses an authentication system in which systems 1302, 1304, and 1306 receive network resource requests from a client device. §V(C)(3) (limitation 1.b.i); EX1004, Fig. 13A, (“[t]he present invention enables application service providers score risk for each online login and transaction and to increase authentication security in real time”). As part of the process, server 1306, for example, creates cookies—secure cookie, flash cookie—that can be used to authenticate individuals. §§V(C)(4), (6) (limitations 1.b.ii, 1.b.iv). Server 1302 may utilize the cookies in authenticating the user (“system 1302 does not itself perform pre-authentication processing, but does performs[sic] ‘post-authentication services’”). Although Varghese does not explicitly disclose that server 1302 and

server 1306 are part of the same domain, Hinton discloses a computing environment in which cookies can be used to authenticate users at servers located in different domains. §IV(D) (claim 2). EX1002, ¶217.

A POSITA would have been motivated by the teachings of Hinton to modify Varghese such that system 1302 is in a different domain than system 1306 and had a reasonable expectation of success. §V(B) (motivation to combine). Accordingly, Varghese in view of Hinton yields a computing environment in which the user is provided with cookies generated by a first system located in a first domain (e.g., system 1306) and can use those cookies to verify themselves at a second system in a second domain (e.g., system 1304). EX1002, ¶218.

E. Claim 7

Varghese in view of Hinton discloses and renders this claim obvious. *See* §V(G) (claim 6), §VI(D) (claim 2). EX1002, ¶219.

Discretionary Denial is not Warranted Pursuant to Acting Director Coke M. Stewart's March 26, 2025, Memorandum regarding Interim Processes for PTAB Workload Management, Petitioner understands that discretionary denial issues if any will be raised in a separate brief to be filed by Patent Owner. If Patent Owner files such a brief, Petitioner intends to respond in an opposition brief consistent with Acting Director Coke M. Stewart's March 26, 2025, Memorandum regarding Interim Processes for PTAB Workload Management. Accordingly, Petitioner will

not address discretionary denial issues in this Petition other than to note that Petitioner is filing concurrently with this Petition a Motion for Joinder with Home Depot's now-instituted IPR2024-01316 for the '823 patent, and if joined, Petitioner would be taking an understudy role and the Board's finite resources would not be impacted.

VII. COMPLIANCE WITH FORMAL REQUIREMENTS

A. Mandatory Notices Under 37 C.F.R. §§42.8(b)(1)-(4)

1. Real Party-In-Interest

Petitioner Walmart Inc. and Walmart Stores Texas, LLC are the real parties-in-interest.

2. Related Matters

The '823 patent is subject to the following actions:

- *RavenWhite Licensing LLC v. Walmart, Inc. et al.*, 2:24-cv-00689 (EDTX)
- *RavenWhite Licensing LLC v. The Home Depot, Inc. et al.*, 2:24-cv-00688 (EDTX)

The '823 patent was previously subject to the following actions which have been dismissed without prejudice:

- *RavenWhite Licensing LLC v. The Home Depot, Inc. et al.*, 2:23-cv-00423 (EDTX) (Dismissed)

- *RavenWhite Licensing LLC v. Walmart, Inc. et al.*, 2:23-cv-00418

(EDTX) (Dismissed)

3. Lead and Backup Counsel

Lead Counsel	Backup Counsel
James M. Heintz Reg. No. 41,828 DLA Piper LLP (US) 11911 Freedom Dr., Suite 300 Reston VA 20190-5602 Phone: 1-703-773-4000 Fax: 1-703-773-5000 jim.heintz@us.dlapiper.com	Clayton W. Thompson Reg. No. 58,463 DLA Piper LLP (US) 1201 W. Peachtree St. NW, Suite 2900 Atlanta, GA 30309-3800 Tel: 404-736-7800 Fax: 404-6872-7800 Clayton.thompson@us.dlapiper.com

4. Service Information

Service information for lead and backup counsel is provided in the designation of lead and backup counsel above. Petitioner consents to electronic service to lead and backup counsel and to: DLA-Walmart-RavenWhite@us.dlapiper.com.

B. Proof of Service on the Patent Owner

In accordance with 37 C.F.R. §§42.6(e) and 42.105, as identified in the attached Certificate of Service, a copy of this Petition in its entirety is being served electronically to the Patent Owner's attorney of record, as well as on counsel for Patent Owner in the District Court Litigation.

C. Power of Attorney

Powers of attorney are being filed with designation of counsel in accordance with 37 C.F.R. §§41.10(b).

D. Standing

In accordance with 37 C.F.R. §42.104(a), Petitioner certifies that the '823 patent is available for *inter partes* review and that Petitioner is not barred or estopped from requesting an *inter partes* review challenging the patent claims on the grounds identified in this Petition.

E. Fees

The undersigned authorizes the Director to charge the fee specified by 37 C.F.R. §§42.15(a) and any additional fees that might be due in connection with this Petition to Deposit Account No. DA 503266.

VIII. CONCLUSION

All challenged claims should be found unpatentable for the reasons discussed in this Petition.

Respectfully submitted,

/s/ James M. Heintz

James M. Heintz
Reg. No. 41,828
DLA Piper LLP (US)
11911 Freedom Dr., Suite 300
Reston VA 20190-5602
Phone: 1-703-773-4000
Fax: 1-703-773-5000

U.S. Patent No. 10,594,823
Petition for *Inter Partes* Review
jim.heintz@us.dlapiper.com

***Attorney for Petitioner, Walmart Inc.
and Walmart Stores Texas, LLC***

CERTIFICATE OF WORD COUNT

Pursuant to 37 C.F.R. §§42.24(d), Petitioner certifies that this petition includes 13,519 words, as measured by Microsoft Word, exclusive of the table of contents, mandatory notices under §§42.8, certificates of service, word count, claim listing, and exhibits.

Date: March 28, 2025

/s/ James M. Heintz

James M. Heintz

Reg. No. 41,828

DLA Piper LLP

11911 Freedom Dr., Suite 300

Reston VA 20190-5602

Phone: 1-703-773-4000

Fax: 1-703-773-5000

jim.heintz@us.dlapiper.com

***Attorney for Petitioner, Walmart, Inc
and Walmart Stores Texas, LLC.***

CERTIFICATE OF SERVICE

The undersigned certifies pursuant to 37 C.F.R. §§42.6(e) and 42.105 that on March 28, 2025, a true and correct copy of the Petition for *Inter Partes* Review of U.S. Patent No. 10,594,823 was served by emailing a copy of same by agreement to the following attorneys, who have agreed to accept service on behalf of the Patent Owner:

Robert Kramer
Kramer Alberti Lim & Tonkovich LLP
950 Tower Lane, Suite 1725
Foster City, CA 94404
Office: 650-514-2747
Mobile: 415-419-1895
Krameralberti-ravenwhite@krameralberti.com
rkramer@krameralberti.com

A courtesy copy was sent to the below counsel via electronic mail:

Robert F. Kramer
David Alberti
Robert C. Mattson
Russell Steven Tonkovich
Sal Lim
Kramer Alberti Lim & Tonkovich LLP
577 Airport Blvd., Suite 250
Burlingame, CA 94010
Rkramer@krameralberti.com
Dalberti@krameralberti.com
Rmattson@krameralberti.com
Rtonkovich@krameralberti.com
Slim@krameralberti.com

Andrea Fair
Ward, Smith & Hill
Andrea@wsfirm.com

Respectfully submitted,

/s/ James M. Heintz
James M. Heintz