

UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE PATENT TRIAL AND APPEAL BOARD

APPLE INC.
Petitioner

v.

HEADWATER RESEARCH LLC
Patent Owner

Inter Partes Review Case No. IPR2025-01572
U.S. Patent No. 10,064,055

**PETITION FOR *INTER PARTES* REVIEW
OF U.S. PATENT NO. 10,064,055**

TABLE OF CONTENTS

I.	INTRODUCTION.....	1
II.	SUMMARY OF THE '055 PATENT.....	1
A.	DESCRIPTION OF THE '055 PATENT	1
B.	SUMMARY OF THE PROSECUTION HISTORY	2
C.	LEVEL OF ORDINARY SKILL IN THE ART	3
III.	REQUIREMENTS UNDER 37 C.F.R. § 42.104	3
A.	GROUND FOR STANDING UNDER 37 C.F.R. § 42.104(A).....	3
B.	IDENTIFICATION OF CHALLENGE UNDER 37 C.F.R. § 42.104(B)	3
C.	CLAIM CONSTRUCTION UNDER 37 C.F.R. § 42.104(B)(3).....	6
IV.	ANALOGOUS PRIOR ART.....	6
V.	GROUND 1: CLAIMS 1-2, 6-11, 13-15, AND 18-19 ARE OBVIOUS OVER LUNDBLADE, JOBST, AND HARDJONO	8
A.	CLAIM 1	8
1.	<i>Claim 1[Pre]</i>	8
2.	<i>1[a]</i>	10
3.	<i>1[b]</i>	14
a)	Jobst's Teachings	15
b)	Combination of Lundblade and Jobst.....	16
c)	Motivation to Combine Lundblade and Jobst	16
4.	<i>1[c]</i>	18
5.	<i>1[d]</i>	20
a)	Jobst's Teachings	21
b)	Combination of Lundblade and Jobst.....	22
c)	Motivation to Combine Lundblade and Jobst	22
6.	<i>1[e]</i>	24
7.	<i>1[f]</i>	26
8.	<i>1[g]</i>	30
a)	Hardjono's Teachings	32
b)	Combination of Lundblade-Jobst and Hardjono	37
c)	Motivation to combine Lundblade and Hardjono	39
9.	<i>1[h]</i>	42
10.	<i>1[i]</i>	43
B.	CLAIM 2	44
C.	CLAIM 6	44
D.	CLAIM 7	46

E.	CLAIM 8	47
F.	CLAIM 9	49
G.	CLAIM 10	50
H.	CLAIM 11	52
I.	CLAIM 13	53
J.	CLAIM 14	54
K.	CLAIM 15	55
L.	CLAIM 18	55
M.	CLAIM 19	59
VI.	GROUND 2: CLAIM 3 IS OBVIOUS OVER LUNDBLADE, JOBST, HARDJONO, AND ZUCCHERATO	59
A.	CLAIM 3	59
a)	Zuccherato's Teachings.....	60
b)	Combination of Lundblade, Jobst, Hardjono, and Zuccherato	61
c)	Motivation to Combine with Zuccherato	62
VII.	GROUND 3: CLAIM 4 IS OBVIOUS OVER LUNDBLADE, JOBST, HARDJONO, AND CHEN	63
A.	CLAIM 4:.....	63
a)	Chen's Teachings	64
b)	Combination of Lundblade, Jobst, Hardjono and Chen	65
c)	Motivation to Combine with Chen.....	66
VIII.	GROUND 4: CLAIMS 5 AND 16 ARE OBVIOUS OVER LUNDBLADE, JOBST, HARDJONO, AND LUNDBLADE '330	67
A.	CLAIM 5	67
a)	Lundblade '330's Teachings	68
b)	Combination of Lundblade, Jobst, Hardjono, and Lundblade '330	69
c)	Motivation to Combine with Lundblade '330.....	70
B.	CLAIM 16	71
IX.	GROUND 5: CLAIM 12 IS OBVIOUS OVER LUNDBLADE, JOBST, HARDJONO, AND FOSBACK	71
A.	CLAIM 12	71
a)	Fosback's Teachings	72
b)	Combination of Lundblade, Jobst, Hardjono, and Fosback.....	73

c)	Motivation to Combine with Fosback	74
X.	GROUND 6: CLAIM 17 IS OBVIOUS OVER LUNDBLADE, JOBST, HARDJONO, AND SCARLATA.....	75
A.	CLAIM 17	75
a)	Scarlata’s Teachings.....	76
b)	Combination of Lundblade, Jobst, Hardjono, and Scarlata	77
c)	Motivation to Combine with Scarlata	78
XI.	CONCLUSION.....	80
XII.	MANDATORY NOTICES UNDER 37 C.F.R. § 42.8(A)(1)	82
A.	REAL PARTY-IN-INTEREST	82
B.	RELATED MATTERS	82
C.	LEAD AND BACK-UP COUNSEL	82
D.	37 C.F.R. § 42.8(B)(4) – SERVICE INFORMATION	83

TABLE OF AUTHORITIES

Cases:

<i>KSR Int’l Co. v. Teleflex Inc.</i> , 550 U.S. 398 (2007)	17, 22
---	--------

Statutes:

35 U.S.C. § 103	2, 5
35 U.S.C. § 112	2

Regulations:

37 C.F.R. § 42.6(e)	92
37 C.F.R. § 42.8	91
37 C.F.R. § 42.8(a)(1)	82
37 C.F.R. § 42.8(b)(1)	82
37 C.F.R. § 42.8(b)(3)	82
37 C.F.R. § 42.8(b)(4)	82, 83
37 C.F.R. § 42.42.24	91
37 C.F.R. § 42.104	3
37 C.F.R. § 42.104(a)	3
37 C.F.R. § 42.104(b)	3
37 C.F.R. § 42.104(b)(3)	6
37 C.F.R. § 42.104(b)(2)	4
37 C.F.R. § 42.104(b)(4)	6
37 C.F.R. § 42.104(b)(5)	6
37 C.F.R. § 42.105	92
37 C.F.R. § 42.105(b)	92

I. INTRODUCTION

Petitioner Apple Inc. requests *inter partes* review (“IPR”) of Claims 1-19 (the “Challenged Claims”) of U.S. Patent No. 10,064,055 (“EX1001”) (“the ’055 Patent”).

II. SUMMARY OF THE ’055 PATENT

A. Description of the ’055 Patent

The ’055 Patent purports to describe improvements to the security of mobile devices on a network. EX1001, Abstract, 1:53-55. The ’055 Patent describes a method of providing a device credential to an end-user device 100. EX1001, 46:47-47:10. The method includes sending a request that includes “a device identifier” from the end-user device 100. EX1001, 46:47-58, 38:25-28. The network verifies the device identifier, then generates a device credential for the end-user device, storing the credential as “associated with end-user device 100.” EX1001, 46:60-65, 47:2-8, 38:36-40. The network sends the device credential to device 100. EX1001, 47:8-10. The device credential may be a “certificate, a key, a shared secret, a password, a hash, a shared algorithm, or any other item of information that allows” the network to confirm the device’s identity. EX1001, 45:57-62, 56:59-62. A device identifier is “(e.g., any identifier associated with the end-user device).” EX1001, 38:28-31.

The ’055 Patent also purportedly improves network and device security by validating an “application installed on an end-user device is authentic.” EX1001,

56:54-56, 57:40-42. Some methods of validating the application include “generat[ing] a signed application credential using the application credential and a device credential.” EX1001, 57:50-53. The network then sends the signed application credential to the end-user device. EX1001, 57:56-57.

B. Summary of the Prosecution History

The '055 Patent issued from Application No. 15/287,597, filed October 6, 2016. EX1001, (21)-(22). The earliest claimed priority for the '055 Patent is U.S. Application No. 61/206,354, filed January 28, 2009. For purposes of this proceeding only, Petitioner applies January 28, 2009, as the priority date.

The '055 Patent received a single rejection from the Office during prosecution, rejecting claims under 35 U.S.C. § 112 for lacking written description, and 35 U.S.C. § 103 as obvious over US 8,146,142 to Lortz et al. EX1002, 563-567. The Examiner determined that the specification did not have written disclosure for the terms “secure device credential,” “device identifiers of the first device,” “second request,” “secured application credential,” “general application credential for the first application,” and “first secured application credential unique to the first device” as recited in the claims. EX1002, 564-565.

In response, Applicant pointed to disclosure purportedly supporting these terms. EX1002, 727-730 (citing description of FIGs. 22, 35, 37, and 38 as exemplary support). Additionally, Applicant argued that Lortz failed to teach “associating the

secure device credential with one or more device identifiers” because Lortz focuses on enrollee identification. EX1002, 730-731. Applicant also argued that Lortz fails to teach generating the secured application credential and receiving a second request because Lortz’s “application merely seeks and receives configuration data and keys for its own secure sessions.” EX1002, 730-731. None of the claims were amended, and a Notice of Allowance was issued. EX1002, 990.

C. Level of Ordinary Skill in the Art

A POSITA at the time of the ’055 Patent would have had a master’s degree in computer engineering, computer science, electrical engineering, or a related field, with at least two years of experience in the field of data communications, or a bachelor’s degree in the same field with at least four years of experience in the field of data communications. Additional education or experience might substitute for the above requirements.

III. REQUIREMENTS UNDER 37 C.F.R. § 42.104

A. Grounds for Standing Under 37 C.F.R. § 42.104(a)

Petitioner certifies the ’055 Patent is eligible for IPR and that Petitioner is not barred or estopped from requesting an IPR challenging the claims of the ’055 Patent.

B. Identification of Challenge Under 37 C.F.R. § 42.104(b)

In view of the prior art and evidence presented, the Challenged Claims of the ’055 Patent are unpatentable and should be cancelled. 37 C.F.R. § 42.104(b)(1).

Further, based on the prior art references identified below, IPR of the Challenged

Claims should be granted. 37 C.F.R. §42.104(b)(2).

Claim(s)	Basis ¹	Reference(s)
1-2, 6-11, 13-15, 18-19	§ 103	Lundblade (EX1004), Jobst (EX1005), Hardjono (EX1006)
3	§ 103	Lundblade (EX1004), Jobst (EX1005), Hardjono (EX1006), Zuccherato (EX1007)
4	§ 103	Lundblade (EX1004), Jobst (EX1005), Hardjono (EX1006), Chen (EX1008)
5, 16	§ 103	Lundblade (EX1004), Jobst (EX1005), Hardjono (EX1006), Lundblade '330 (EX1009)
12	§ 103	Lundblade (EX1004), Jobst (EX1005), Hardjono (EX1006), Fosback (EX1010)
17	§ 103	Lundblade (EX1004), Jobst (EX1005), Hardjono (EX1006), Scarlata (EX1011)

¹The pre-AIA version of 35 U.S.C. § 103 applies.

Sections **Error! Reference source not found.**-X identifies where each element of the Challenged Claims is found in the prior art. 37 C.F.R. §42.104(b)(4). The exhibit numbers of the evidence relied upon to support the challenges are provided above, and the relevance of the evidence to the challenges raised is provided in Sections V-X. 37 C.F.R. § 42.104(b)(5). Exhibits 1001-1061 are also attached.

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

For purposes of this IPR only, Petitioners do not believe claim construction is necessary. Under any reasonable construction, the Challenged Claims are unpatentable as explained herein.²

IV. ANALOGOUS PRIOR ART

The prior art is analogous to the claimed invention of the '055 Patent. The prior art is from the same field of endeavor as the '055 Patent, namely, security of devices and/or software communicating data via a network. EX1001, 1:53-55, Abstract, 56:54-56, 57:40-42, 46:47-5; EX1004, Abstract, 9:59-67, 10:19-50, FIGs. 1-2, 10:60-11:4, 11:37-39, 16:49-7:6, 7:10-17, FIG. 5, 17:46-18:30, FIG. 8; EX1005, Abstract, 1:41-49, 6:31-33, 6:57-7:32, 7:37-65, 8:1-25, 8:45-49, 9:5-10,

² Petitioner's application of prior art to the Challenged Claims is not a waiver of any argument in litigation regarding a claim term's construction or indefiniteness.

9:11-14, FIG. 3; EX1006, Abstract, [0009], [0031], FIG. 1, [0095], [0038], [0101]-[0104], [0068], [0041], FIGs. 15-16; EX1007, Abstract, [0023],-[0024], [0031]-[0033], FIG. 2; EX1008, Abstract, 8:16-41, 8:45-66, 9:20-24; EX1009, Abstract, [0028], [0031], [0034]-[0035], [0042]-[0043], [0047]-[0048], [0056], [0058]-[0060]; EX1010, Abstract, [0040], [0043], [0045]-[0046], [0053]-[0054], [0067], [0069]; EX1011, Abstract, 1:53-63, 7:31-57, 8:40-52; EX1003, ¶¶31-32, ¶65, ¶73, ¶81, ¶86, ¶90, ¶97, ¶102, ¶106.

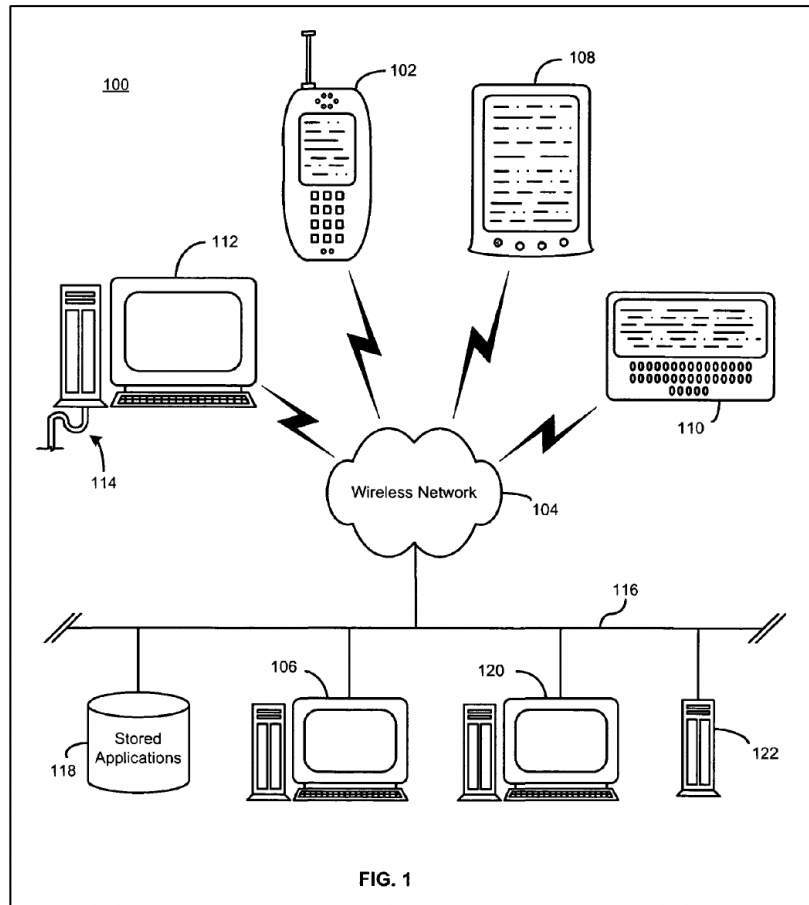
The prior art is also reasonably pertinent to at least one problem facing the '055 Patent's inventors, namely the problem of decreasing fraudulence at a device, application, and/or on a network. EX1001, 1:45-51, 1:55-57, 25:39-52, 56-54-56, 57:40-4; EX1004, Abstract, 2:13-15, 9:59-67, 10:65-11:4, 17:61-18:5; EX1005, Abstract, 1:41-49, 2:1-13, 4:35-46, 7:15-24, 8:17-25; EX1006, Abstract, [0006]-[0009], [0041], [0031], [0101]-[0104]; EX1007, Abstract, [0003], [0005], [0009]-[0010], [0033]; EX1008, Abstract, 1:57-58, 1:12-42, 1:43-53, 8:16-41, 8:45-66; EX1009, Abstract, [0010], [0009], [0006], [0034], [0035], [0058]-[0060]; EX1010, Abstract, [0040], [0043], [0045], [0067], [0069]; EX1011, Abstract, 1:31-39, 1:53-2:3, 7:31-57; EX1003, ¶¶33-34, ¶66, ¶74, ¶82, ¶87, ¶91, ¶98, ¶103, ¶107.

V. GROUND 1: CLAIMS 1-2, 6-11, 13-15, AND 18-19 ARE OBVIOUS OVER LUNDBLADE, JOBST, AND HARDJONO

A. Claim 1

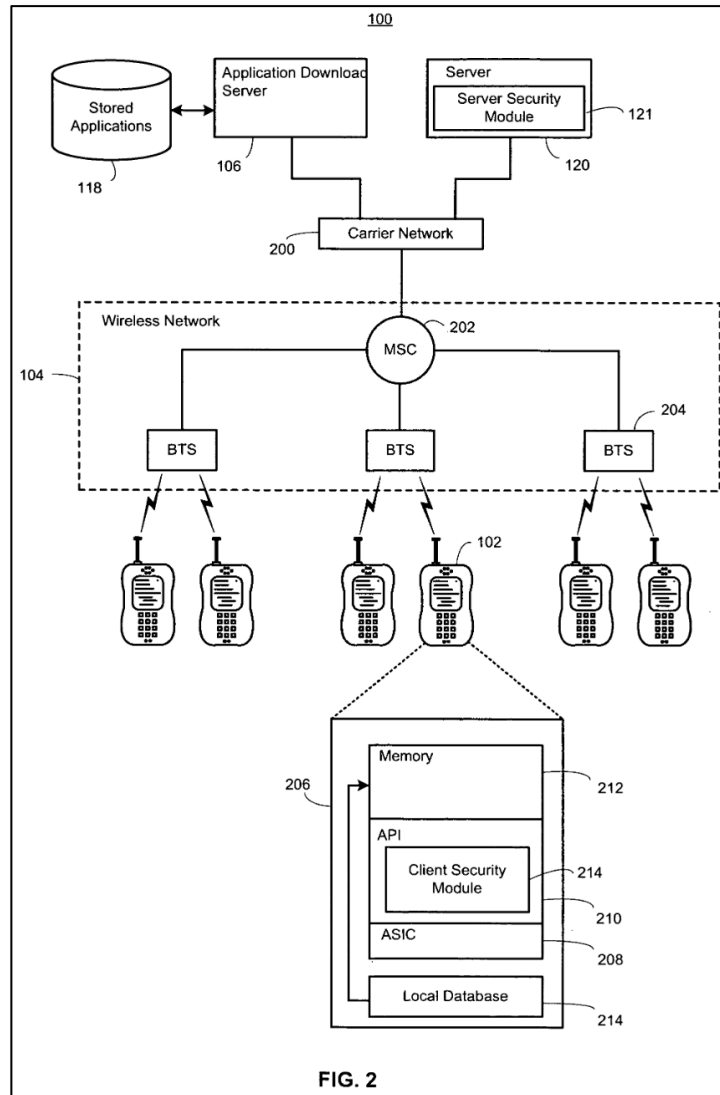
1. Claim 1[Pre]

To the extent the preamble is deemed limiting, Lundblade teaches *a method of operating a network system*; namely, a method for providing authentication via credentials on a network system (i.e., servers 106 and 120, wireless network 104, and carrier network 200) to provide secure communications between remote computing devices and servers. EX1004, Abstract. FIG. 1 shows a system 100 including wireless devices in communication with servers “across a wireless network 104.”



EX1004, FIG. 1, 10:19-23, 10:38-43; EX1003, ¶109.

FIG. 2 “more fully illustrates” network system 100, comprising a wireless network 104, a carrier network 200, and servers 106 and 120.

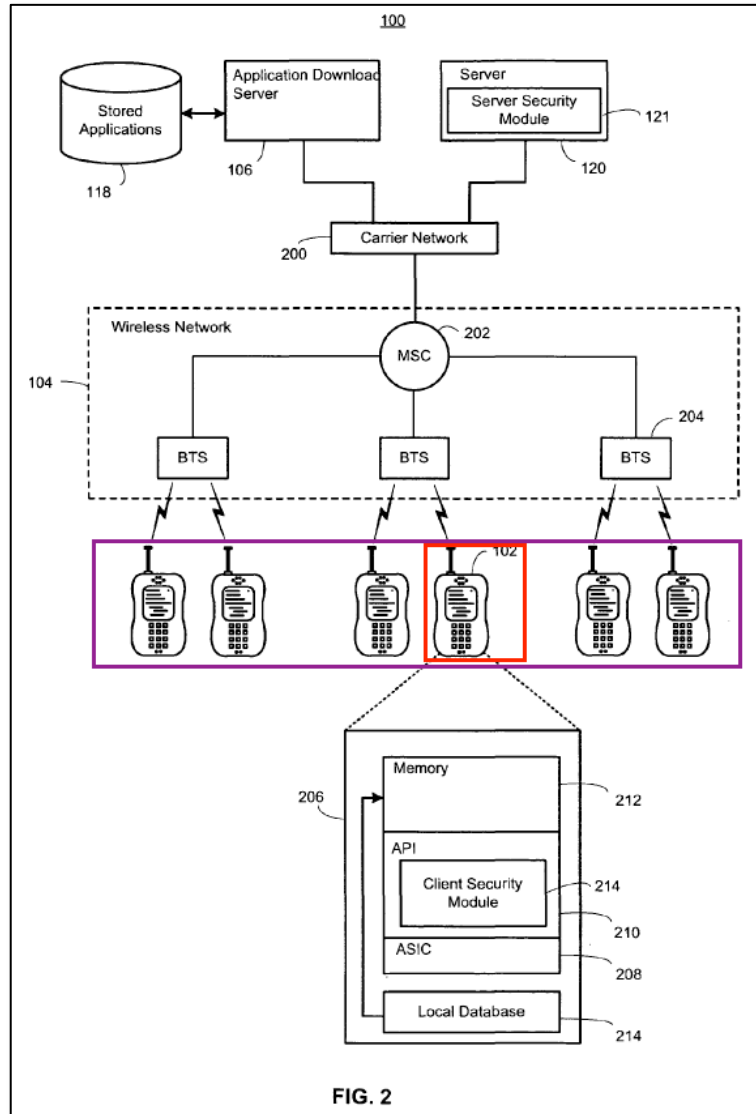


EX1004, FIG. 2; EX1003, ¶110.

2. 1[a]

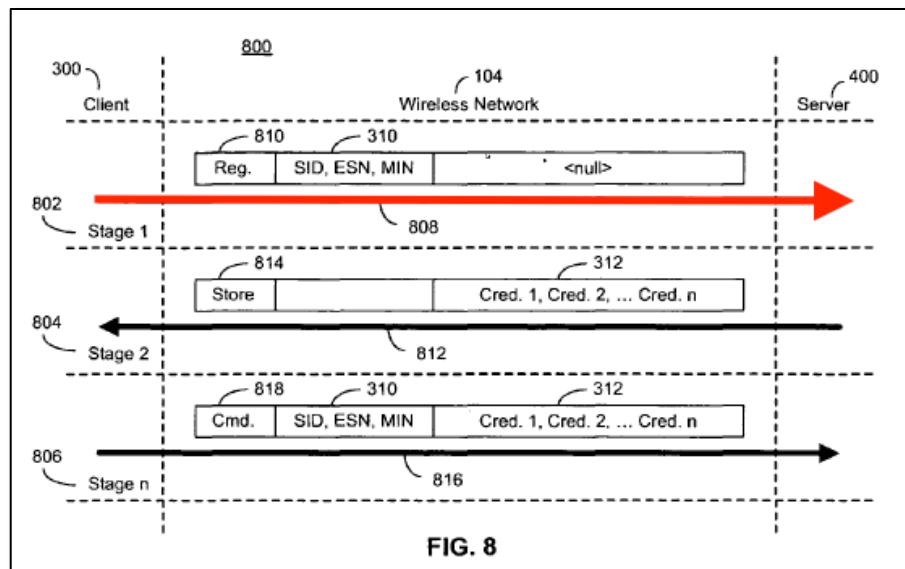
Lundblade teaches *receiving a first request from any first device in a plurality of similar wireless end-user devices, the first request for a secure device credential;* namely, by server 400 (e.g., server 120) receiving a registration command signal from a client 300 (e.g., mobile phone 102) for the purpose of getting credentials 312.

The client 300 (e.g., mobile phone 102 – in red below) is a *first device in a plurality of similar wireless end-user devices*, shown in purple below:



EX1004, FIG. 2. Additionally, Lundblade teaches client 300 may be a personal computer, PDA, cell phone, etc., and these wireless devices are similar because they all comprise a “remote module including wireless communication portal[.]” EX1004, 15:63-65, 10:26-37; EX1003, ¶113.

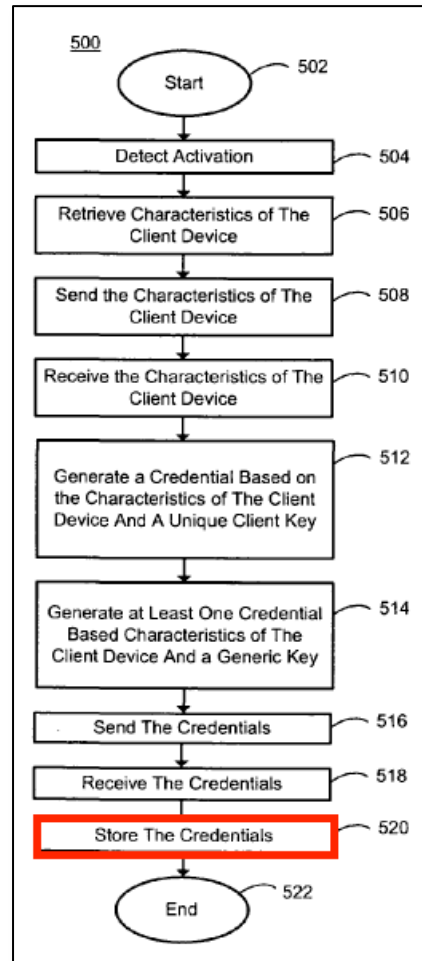
Lundblade FIG. 5 illustrates a method for obtaining credentials 312 for secure communications between a client 300 and the server 400. EX1004, FIG. 5, 16:49-59; EX1003, ¶114. In FIG. 8, the process 800 (which implements method 500) begins when “the client computing device 300 transmits a signal 808 containing a registration command 810 and characteristics 310 of the client computing device 300.” EX1004, 18:14-17 (step 802); EX1003, ¶114 (opining a POSITA would understand process 800 as showing signal transmissions for carrying out method 500).



EX1004, FIG. 8 (annotated). A POSITA would have understood the signal 808 is a *request* because the client is originating a communication for registration to server 400, thus *requesting* registration, and the request is *for a secure device credential* because it includes a registration command that informs the server that the client 300 desires credentials 312 to perform secure communications. EX1003, ¶¶115-116

(analogizing Lundblade's Reg 800 and SID 310 to the '055 Patent's end-user device information (38:24-49), and opining a device registration request was known to serve as a request for credentials, particularly where credentials are sent in response to the request). Additionally, the server generates credentials 312 "in response to" receiving the registration command, indicating the registration command is a request that is responded to by server 400. EX1004, 14:59-63; EX1003, ¶116.

Included in signal 808 are characteristics 310 of the client. The server generates multiple credentials 312 based on characteristics 310 and a unique client key 409 (to form credential 320) and a generic key 410 (to form credential 322). EX1004, 16:59-17:3. The server uses any of various cryptographic algorithms to produce each credential. EX1004, 12:51-58. Thus, each credential (*i.e.*, credential 320 and credential 322, collectively (referred to as credentials 312) or individually) is a secure device credential because they are generated through cryptographic algorithms, such as a one-way hash function. EX1003, ¶117. Once credentials 312 are generated, the server sends credentials 312 to client 300 for storing in system memory 302. EX1004, 17:4-10.



EX1004, FIG. 5 (annotated).

3. *1[b]*

Lundblade in view of Jobst teaches performing a verification step of characteristics 310 provided by client 300, prior to generating any credentials 312.

Lundblade teaches client 300 transmits its device characteristics 310 to server 400 alongside the registration command 810 within signal 808. *See* Claim 1[a]. The characteristics 310 include one or more device identifiers, such as a hardware ID (electronic serial number (ESN), system identification number (SID), etc.). EX1004,

12:29-42. The device characteristics 310 are “useful in identifying the client computing device 300[.]” EX1004, 12:43-46. A POSITA would have understood characteristics such as the SID and ESN are unique to client 300 and constitute *one or more device identifiers supplied by the first device*. EX1003, ¶119.

a) Jobst’s Teachings

Jobst teaches verifying a device identifier (i.e., an International Mobile Equipment Identity (IMEI)) at a server by comparing a received IMEI from the phone to a database table of IMEIs stored at a certification server to determine if the requesting device has an open account. EX1005, 7:15-22. This is much like the ’055 Patent describing verifying the device identifier from a table of allowed device IDs on the server. EX1001, 46:62-65; EX1003, ¶120. When the IMEI is verified, Jobst’s certification server calculates and outputs a phone password (i.e., device credential) based on the IMEI and a master password. EX1005, 7:23-32.

Jobst teaches a mobile phone that downloads an application, application program segment, or instruction for activating an already-downloaded application from a software provider. EX1005, 6:57-7:4, 8:46-49. The software provider 34 uses a certification center 35 to check the validity of the request by comparing the IMEI to a database of IMEI codes for authorized phones. EX1005, 7:15-22. The IMEI “is a universal code unambiguously identifying the phone.” EX1005, 7:34-35. If the

request is deemed valid, the certification center produces a phone password 40 used to download code from the software provider. EX1005, 7:23-8:27.

b) Combination of Lundblade and Jobst

The process of Lundblade is modified to include the verification step taught by Jobst. Jobst teaches *performing a verification step of one or more device identifiers supplied by the first device* (an IMEI). In the combination, any of the device-identifying characteristics contemplated by Lundblade (e.g., SID or ESN) is compared to database entries on the server, as described by Jobst, to determine if the device has permission to use the network. If the received characteristic matches a characteristic stored in the database, the process of Lundblade continues, and the device is provided with credentials to perform secure communications using the network. A POSITA would have understood that credentials for the client are only generated *upon a successful verification in the verification step*, in the same manner that Jobst only calculates a phone password if the IMEI is valid. EX1005, 7:23-25; EX1003, ¶122.

c) Motivation to Combine Lundblade and Jobst

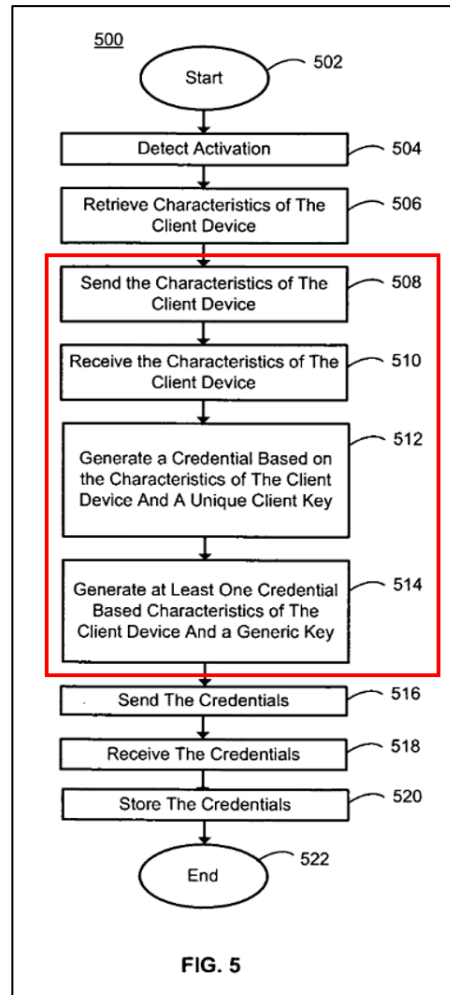
A POSITA would have been motivated and found it obvious to modify Lundblade's secure communication method to verify a characteristic 310 of the device at server 400 by determining if the characteristic matches a value stored in a database, as taught by Jobst. EX1005, 7:15-22. Such a modification constitutes use

of a known technique (verifying that a device has permission to access a network and/or server by verifying the device identifier against stored identifiers at a server) to improve a similar method (providing credentials for wireless devices) in the same way; *i.e.*, by ensuring only authorized individuals are able to set up secure communications with the server. *KSR Int'l Co. v. Teleflex Inc.*, 550 U.S. 398, 417-418 (2007). A POSITA would have appreciated verifying the client is permitted to gain access to the network system prior to providing credentials, preventing credentials from being provided to unauthorized devices. EX1003, ¶¶123-124. Including a verification step would have ensured unauthorized devices do not receive credentials for performing secure communications over the network. *Id.*

Further, a POSITA would have had a reasonable expectation of success in making the modification, given Lundblade teaches server 400 stores device characteristics 310 and that client 300 sends characteristics to server 400. Thus, the modification requires only programming server 400 with additional logic to check the received characteristics (e.g., ESN, SID, etc.) against characteristics stored in the database to verify the identity of the client. EX1003, ¶125. This modification would have been simple to a POSITA, requiring no undue experimentation, as verifying the identity of wireless devices trying to communicate over a network was a conventional server/database function. EX1005, 2:27-33; EX1003, ¶125.

4. *1[c]*

Lundblade teaches that server 400 generates credentials for client 300 (i.e., *for the first device*). After client 300 sends characteristics 310 to server 400 via the network, “the server 400 generates a credential 320 based on the characteristics 310 of the client computing device 300 and a unique client key 409[,]” and “at least one credential 322 based on the characteristics 310 of the client computing device 300 and a generic key 410.” EX1004, 12:66-13:5. Credential 320 and credential 322 constitute credentials 312, and each, alone or in combination is a *secure* credential. *See* Claim 1[a]. FIG. 5 illustrates this process.



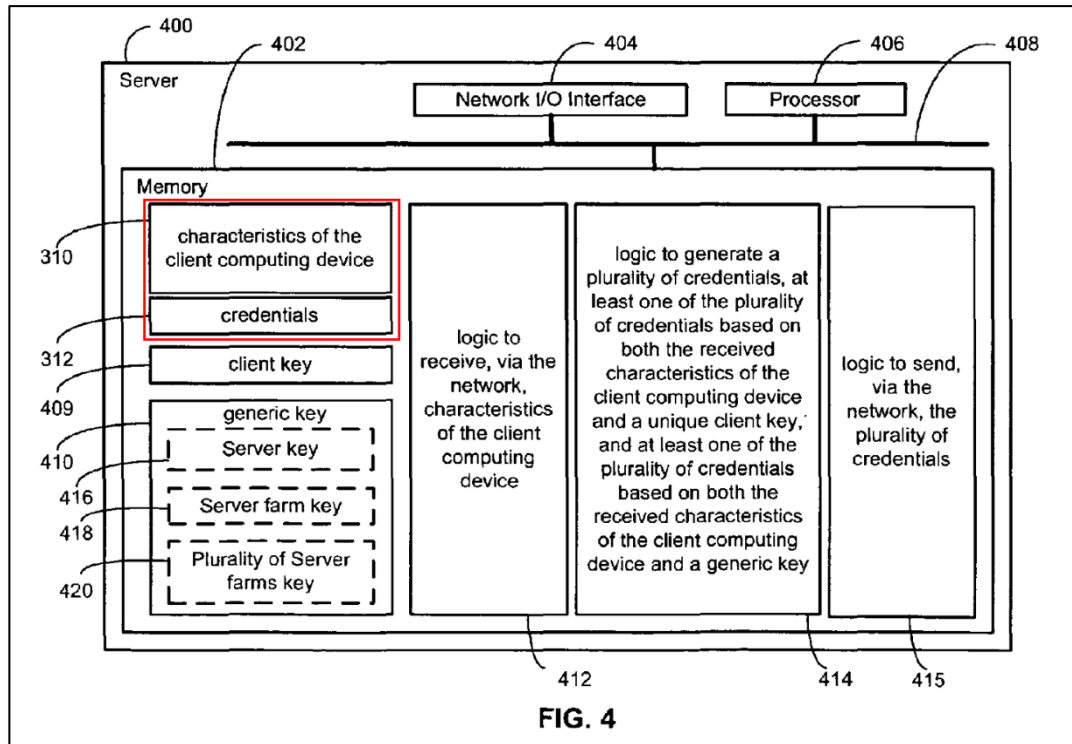
EX1004, FIG. 5 (annotated). Lundblade teaches one or more credentials being “formulated by performing a one-way cryptographic operation on characteristics 310 of the client computing device 300 and the client key.” EX1004, 12:51-56. Since credentials 312 are created for the device using device characteristics and are cryptographically generated, Lundblade teaches *generating a secure device credential for the first device*. EX1003, ¶¶126-127 (noting that “a one-way cryptographic operation” would be a hash, consistent with disclosure of the ’055

Patent at 45:58-62 and 38:33-36 and opining cryptography is one way of securing data).

5. *1[d]*

Lundblade in view of Jobst teaches *associating the secure device credential with one or more device identifiers of the first device*. Lundblade teaches server 400 includes memory 402, and “located in memory 402 is characteristics 310 of the client computing device 300, credentials 312.” EX1004, 15:11-22; *cf.* ’055 Patent, 47:5-8 (“*service controller 122 stores the device credential* in database cluster 116 *as associated with* the subscriber identifier and *the device identifier* associated with end-user device 100.”).³

³ All emphasis is added unless noted otherwise.



EX1004, FIG. 4 (annotated).

a) Jobst's Teachings

Jobst teaches storing a device-specific password in a server record and addressing the record with the device's IMEI code so a new password does not need to be calculated every time the device requests software:

Alternatively the certification center may calculate the phone password when the account is established and *store the password in the record that may be addressed by the IMEI code*. Then there will be no need for calculating the phone password every time new software is requested.

EX1005, 9:11-15. A POSITA would have understood, in addressing the phone password via the IMEI when stored, the phone password is *associat[ed]* with the

IMEI, much like the '055 Patent stores the device credential at the server “as associated with” the device identifier. EX1001, 47:5-8; EX1003, ¶129.

b) Combination of Lundblade and Jobst

The Lundblade-Jobst combination associates the credentials 312 with the device characteristics 310 in server memory (i.e., *associating the secure device credential with one or more device identifiers of the first device*) in the manner described by Jobst—as addressed by the characteristics. This provides the benefit of Jobst, in that the server does not need to generate and provide new credentials every time the client asks to establish secure communications with the server; rather, the credentials may be addressed knowing only the device characteristics.

c) Motivation to Combine Lundblade and Jobst

A POSITA would have been motivated and found it obvious to modify Lundblade to store the credentials in a record addressed by the characteristics in the manner discussed by Jobst, so credentials 312 are stored in association with the characteristics 310. EX1005, 9:11-15. Such a modification constitutes the use of a known technique (i.e., storing device credentials in association with device identifier on server side) to improve similar devices (network systems) in the same way by creating an organized, streamlined storage system and reducing data required to be communicated from the client. *KSR*, 550 U.S. at 417-18; EX1003, ¶131. Additionally, this association provides a simplified method to identify compromised

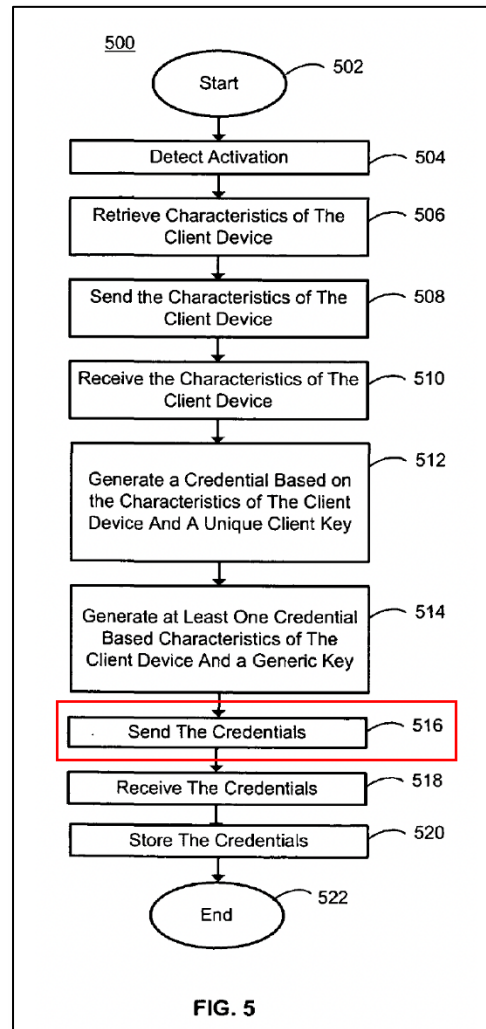
devices. Given that both credentials and devices becoming compromised were known issues, a POSITA would have appreciated that associating credentials 312 with the corresponding characteristics 310 in the server memory 402 allows a network operator to identify the device associated with compromised credentials from server 400. EX1004, 16:29-33, 3:30-33; EX1003, ¶132. Additionally, a POSITA would have appreciated storing credentials 312 in association with characteristics 310 at server 400 removes the need to send characteristics with each signal, thereby reducing the risk of the characteristics being intercepted by a bad actor and replicated by an unauthorized user, while reducing message lengths. EX1004, 18:30-32; EX1003, ¶133.

A POSITA would have had a reasonable expectation of success in making this modification, given Lundblade already teaches server 400 stores characteristics 310 and credentials 312 in the same memory 402. EX1004, 15:11-22. Thus, the modification requires only reprogramming server 400 to associate credentials 312 with corresponding characteristics 310 in memory 402, such as, e.g., by generating a pointer that points from credentials to characteristics. EX1003, ¶134. This modification would have been within the skill of a POSITA without undue experimentation, as networks were known to store device credentials with device identifiers. *Id.*

6. *1[e]*

Lundblade teaches *securely sending the secure device credential to the wireless end-user device*. Namely, Lundblade teaches, in step 516 of FIG. 5, server 400 sends credentials 312 to client 300 “under a secure mode” to “prevent unwanted snooping of data by unauthorized programs and unauthorized entities.” EX1004, 17:4-20. Lundblade discusses that vulnerabilities arise in systems “when both the cryptographic algorithm and the data used by such algorithms are freely available[.]” EX1004, 2:34-36; EX1003, ¶135 (explaining how prior systems allowed credential forgery). To prevent such vulnerabilities, Lundblade teaches the step of transmitting the device credential is performed in a “secure mode.” EX1004, 17:4-15. The secure mode is accomplished in “any number of currently known techniques known in the art,” including the use of a “secure file system.” EX1004, 17:13-20. A POSITA would understand that this “secure mode” thus includes performing the cryptographic algorithm on data protected by the secure file system to prevent the vulnerabilities. EX1003, ¶135. Furthermore, since step 516 (“send the credentials”) is included in the list of steps performed in the secure mode (EX1004, 17:13-15), a POSITA would have understood the sending is performed via the secure mode, such as by “secure communications” enabled by the “cryptographic (encryption) algorithms” Lundblade discusses in this context. EX1003, ¶136 (opining that a

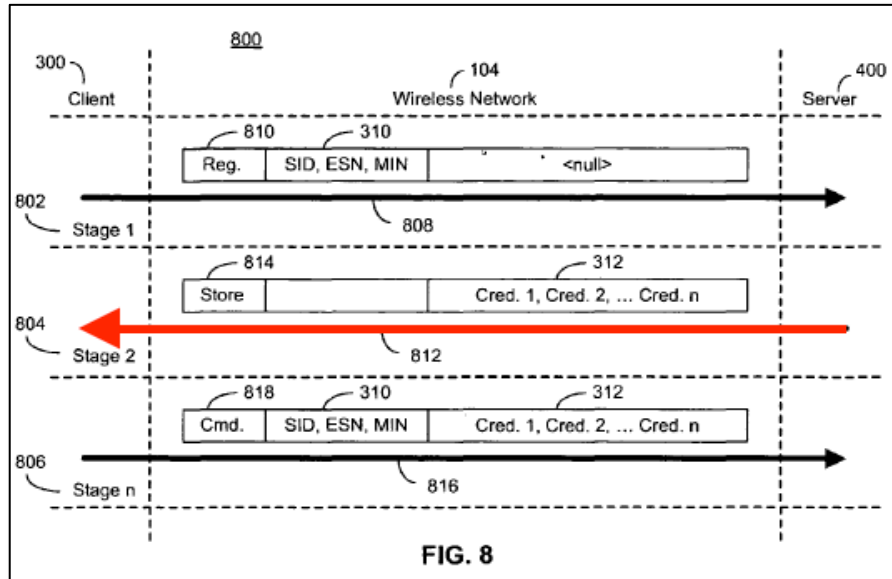
POSITA would have understood Lundblade's disclosure of sending "under a secure mode" to teach transmitting data in encrypted form); EX1004, 2:13-17, 2:30-41.



EX1004, FIG. 5 (annotated).

As illustrated in FIG. 8, a signal 812 including credentials 312 is transmitted to client 300. EX1004, 18:17-21. A POSITA would have understood that FIGs. 5, 7, and 8 depict the same embodiment at different levels of granularity. EX1003, ¶¶137-

138 (opining this is the case at least because the signals of FIG. 8 are those sent in FIGs. 5 and 7).



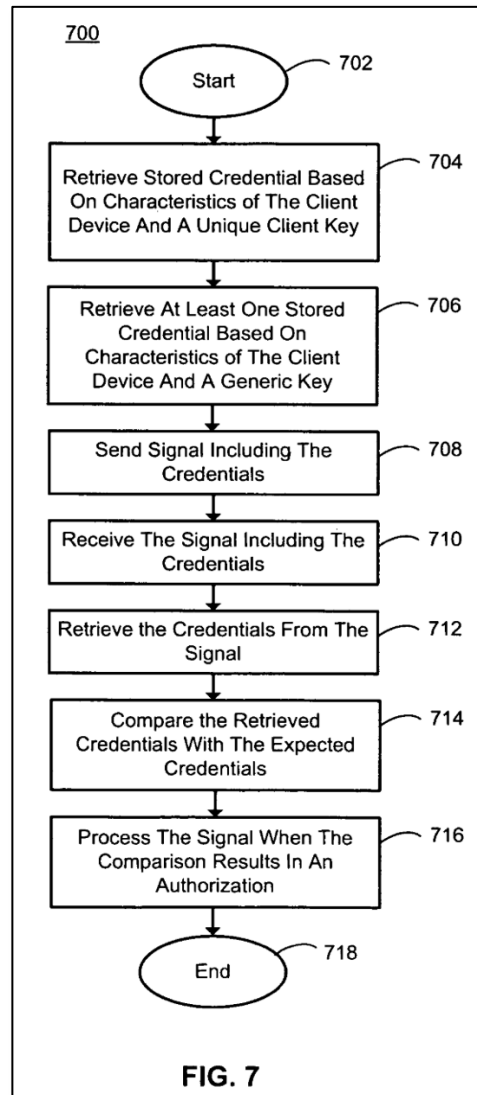
EX1004, FIG. 8 (annotated). A POSITA would have understood transmitting credentials 312 to client 300 using a secure mode constitutes *securely sending the secure device credential to a wireless end-user device*. EX1003, ¶139. A POSITA would have been aware of and capable of implementing common secure modes of transmission known in the art (e.g., encryption). EX1003, ¶140.

7. 1[f]

Lundblade teaches using the credentials to negotiate a secure message link between server 400 and the client security module.

Lundblade teaches after receiving credentials 312, subsequent signals sent between client 300 and server 400 include the credentials for secure

communications. FIG. 7 depicts using the stored credentials to secure communications from client 300 to server 400. EX1003, ¶¶142-143.

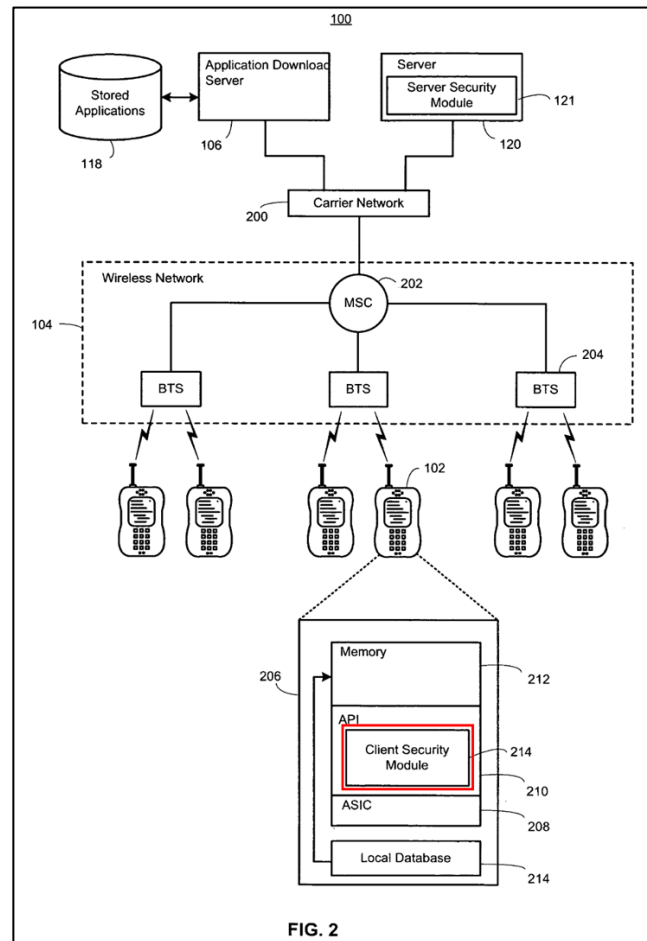


EX1004, FIG. 7. Credentials 312 stored on client 300 are retrieved and included in the signal sent to server 400. EX1004, 17:56-61. Once the signal is received by server 400, the server compares the retrieved credentials with the expected credentials and authorizes the signal upon determining a match. EX1004, 17:61-18:5.

Furthermore, Lundblade describes the method of FIG. 8 as “using signals to initiate secure communications” and “transmissions ... that establish and begin secure communications.” EX1004, 9:17, 18:12-14. A POSITA would have understood “initiat[ing]” or “establish[ing] and “begin[ning]” communications to be an example of *negotiating a secure message link*. EX1003, ¶144. Similarly, a POSITA would have understood sending a signal including credentials that are authorized at a server in the manner taught by Lundblade to be in line with this example of *negotiating a message link between the client and the server based...on the secure device credential*. EX1003, ¶145. By requiring a match between the credentials received from the client and stored credentials, the server may or may not authorize the signal. *Id.* A POSITA would have understood that if the signal is authorized in the manner taught by Lundblade, secured communications are provided, and the link between the server and the client is secured. EX1003, ¶146 (explaining FIG. 7 is a method of providing secure communications, and a POSITA understands the transmission of secure messages between two devices to be a secure link).

Lundblade also teaches client 300 (cellular phone 102 of FIG. 2) includes a client security module 214 containing “logic configured to provide for secure communications over the carrier network 200. Such client security module 214 operates in conjunction with a [server] security module 121 to provide secure

communications.”⁴ EX1004, 11:34-39. “Further, the communications over the wireless network 104 are performed in a secure manner, at least in part, because of the interaction and operation between the client security module 214 and the server security module 121.” EX1004, 11:52-55.



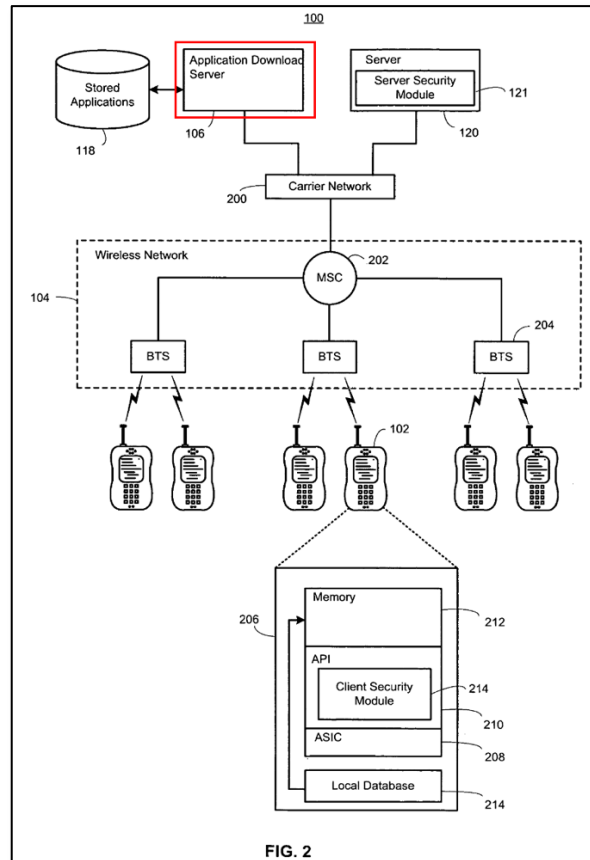
⁴ A POSITA would have understood that 121 refers to the server security module as shown in FIG. 2 and is only mistakenly called the “client security module 121,” in the passage quoted. EX1003, ¶147.

EX1004, FIG. 2 (annotated). A POSITA would have understood the secured communications of FIG. 7 are established between client security module 214 and server security module 121 shown in FIG. 2. EX1003, ¶148. Thus, the *secure message link* is negotiated between server security module 121 (part of a *message link server in the network system*) and client security module 214, a *device link agent in the first device*. *Id.*

8. I[g]

Lundblade in view of Hardjono teaches transmitting a request for access to a network resource that includes a digest (*general application credential*) to obtain a trust report (*secured application credential*) that allows access to network resources.

Lundblade teaches that the client may “receive and execute software applications transmitted from the application download server 106[.]” and that “[i]n at least one embodiment the server 400 can be any of the servers 106, 120, 122 shown and described in relation to FIG. 1.” EX1004, 11:20-23, 14:63-65. Thus, a POSITA would have understood application download servers are one type of server capable of performing secure communications with client 300 over the network. EX1003, ¶150.



EX1004, FIG. 2 (annotated). Further, Lundblade's teachings are applicable to any communications between a wireless device and any other component connected via a wireless network. EX1004, 10:54-60. Thus, a POSITA would have understood that Lundblade is not limited to the server functionality described. EX1003, ¶¶151-152. Further, a POSITA would have understood signals containing various types of content may be sent through the secure communications between client 300 and server 400 once the credentials are generated and stored. EX1003, ¶153.

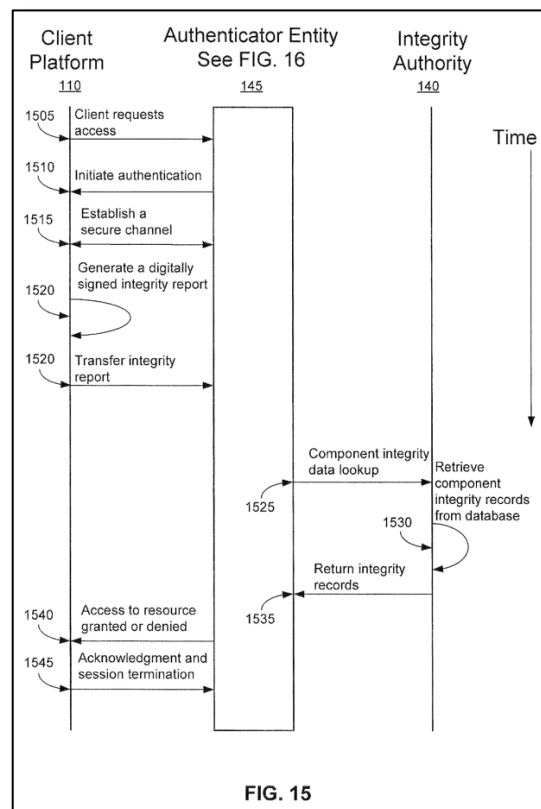
a) Hardjono's Teachings

Hardjono teaches a method to ensure a client and its components (e.g., application software) are in a proper state (*i.e.*, trustworthy and not tainted with malware) prior to providing access to network services and resources. EX1006, [0008], [0006], Abstract, [0036]-[0037]. A mobile device (client platform 110) communicates with a server to obtain access to a network resource by transmitting an integrity report that must be verified by the network prior to the device gaining access to the resource. EX1006, [0009], [0030] (client platform 110 may be a laptop, mobile phone, etc.). An authentication entity 145 and integrity authority 140 work in cooperation to validate the client 110 components and, according to the validation, grant or deny access to the requested network resource or service. EX1006, [0031]. The authentication entity 145 and integrity authority 140 may be “combined and operated by the same physical machine.” *Id.*; EX1003, ¶156.

Hardjono teaches a multi-step process whereby a device requests access to a resource from a server. A POSITA would have recognized that the multi-step requests for resources disclosed in Hardjono are common, particularly where the resource being requested is access-controlled, such as by a password. EX1003, ¶161 (analogizing Hardjono's multi-step request with the HTTP authorization framework). In Hardjono, the request process additionally includes a client integrity report:

FIGS. 15 and 16 show an operation flow of a procedure used to verify the integrity of components 160 of client platform 110 (of FIG. 1). At step 1505, *client platform 110 initiates a handshake with authentication server 210 of authenticator entity 145 of FIG. 2 by requesting access to resource 150*. At step 1510, authentication server 210 responds to the request from client platform 110 by initiating the platform integrity verification of client platform 110.

EX1006, [0101].



EX1006, FIG. 15.

Client 110 transmits an integrity report 170 to authentication entity 145.

EX1006, [0101]-[0102]. Authentication entity 145 is “located in a single place,” and,

in one embodiment, may include “a single server with the functions of authentication server 210 and policy server 205 [sic] performed by verification server 205 alone.” EX1006, [0035]. Digests are generated for hardware and software components of client 110 and are “collected into an integrity report.” EX1006, Abstract, [0026], [0036]-[0038]. For example, digests may be for individual applications on the device. EX1006, [0037]. The digests include hashes of the components and are used to generate integrity report 170:

Digests can include hashes of components 160. Hashes of components can be generated using one-way cryptographic hash functions such as MD5, SHA-1, SHA-256; a person skilled in the art will recognize other possible hash functions that can be used. *Integrity report generator 165 collects all of the digests (including the hashes and other derived information) and generates integrity report 170.* Integrity report 170 can be digitally signed, as further discussed in the description of FIG. 6, to verify the identity of the client platform from which integrity report 170 is received.

EX1006, [0038]; *see also*, [0077]. A digest is a *general application credential* because it uniquely identifies and verifies the application and includes a hash. EX1001, 57:45-48; EX1003, ¶158.

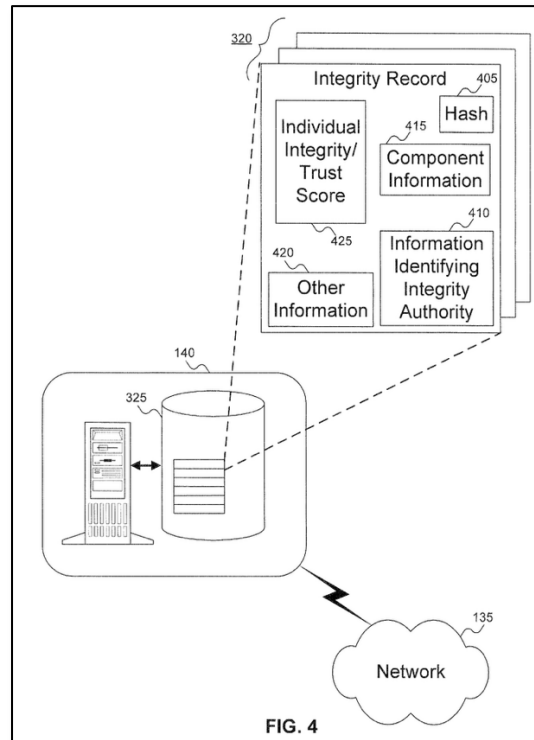
The authentication entity receives integrity report 170 and compares the digests in the integrity report with integrity/trust records to determine a level of trustworthiness for the device. EX1006, [0104], [0045], [0048], [0027], [0077]-

[0079]; EX1003, ¶159. Authentication entity 145 then generates a trust report 220 and grants access to the requested resource, if the digest matches one in records 320. EX1006, [0104], [0085]; EX1003, ¶160 (opining if the digest matches, the access is granted because the component is trustworthy). Hardjono teaches providing the trust report to client, which allows the device to access other network resources or services. EX1006, [0041]. The trust report may include information relating to the components 160, a signature, and a reference to the certificates of the device:

Trust report 220 can also include other information 520, *such as: one or more references to TPM certificates of the client platform 110*, a date/time of the evaluation, and *a digital signature of verification server 205*, and a reference to the database against which the digests collected from the client platform were compared. ... Additionally, other information 520 can include *a list of one or more components 160*, each with a reference to corresponding integrity/trust record 320 in GSDB 325 (shown in FIG. 3). *Verification server 205 generates trust report 220 according to the evaluation of client platform 110.*

EX1006, [0068]; EX1001, 57:50-53 (describing the server generating a “signed application credential” using the application credential and device credential); EX1003, ¶162. The information included in the integrity record 320 is shown in FIG.

4.



EX1006, FIG. 4.

A POSITA would have understood the multi-step request process of steps 1505-1520 (excluding step 1515) forms a *second request from the first device for a secured application credential*, because the authentication entity generates a trust report 220 (i.e., *secured application credential*) in response to receiving a resource request and integrity report. *Supra* Claim [1(f)]; *see also* EX1003, ¶¶163-164 (opining the combination would omit Hardjono’s step 1515 because the secure channel would have already been established as taught by Lundblade). The second request comprises a *general application credential for the first application*, because the integrity report 170 includes a digest containing a hash of “at least one software component.” EX1006, [0095]; EX1003, ¶165.

b) Combination of Lundblade-Jobst and Hardjono

The Lundblade-Jobst-Hardjono combination incorporates the process for validating a software component of the client taught by Hardjono into the secure communication between client 300 and server 400 taught by Lundblade. Specifically, as taught by Hardjono, the client has multiple applications capable of requesting resources from the network (*a plurality of device applications registered to use wireless network communications*). See EX1006, [0037], [0029]. A POSITA would have understood, when accessing a network resource via one of the applications (i.e., accessing email or accessing storage for certain network data), Hardjono would verify the application that is requesting the network resource, and thus the *second request* is for any first application identified from a plurality of device applications registered to use wireless network communications. EX1006, [0037]; EX1003, ¶166. Server 400 of Lundblade incorporates all the functionalities of the authentication entity of Hardjono, allowing server 400 to grant or deny access to a resource. EX1006, Abstract. This includes the functionality of Hardjono's authentication server 210, policy server 215, and verification server 205.

As explained in Claim 1[Pre]-[f], the client and server establish credentials to use for secure communications (that generate a secure link) for future transmissions between the client and the server. See Claim 1[pre]-1[f]. In the combination, when one of the applications requires access to a resource (i.e., *second request*), for

example, “an email package,” the device sends a message to the server including credentials 312 and the request for access to resources for the application (*receiving, over the secure message link, a second request from the device, the second request for a secured application credential*), including steps 1505-1520 (excluding 1515) illustrated in Hardjono FIG. 15, specifying the desired resource and sending an integrity report to the server, along the same link established via secure communication. EX1006, [0101]. The integrity report includes the digest containing a hash (*the second request comprising a general application credential*) for the specific application (e.g., an email package or banking application) (*a first application*) that requires the resource. EX1006, [0037], [0044].

Server 400 performs the integrity verification of the selected application by comparing the digest received with the digests stored on server 400. EX1006, [0102]-[0104]. If the server determines the request is trustworthy (i.e., the digest matches the digest stored on the server), the server decides to grant the request and provide a trust report (*a secured application credential*) to the client that may be used to access other related resources or services. EX1006, [0104], [0041]. The trust report includes information such as the digest, the relevant application, a digital signature of the verification server 205, and a TPM certificate associated with the device. EX1006, [0068], [0048]. In the combination, credentials 312 of Lundblade are used in combination with the TPM certificate when generating the trust report.

EX1003, ¶169. Additionally, the trust report includes the signature of server 400, which has the capabilities of verification server 205. Because the trust report includes the digital signature of the server, the trust report is *secured*. *Id.*

Therefore, the combination of Lundblade-Jobst with Hardjono teaches using secured communications to send a request to a network system for a trust report to access a resource (*a second request from the first device*), the request including a digest of a specific application (*a general application credential of [a] first application*). Further, the application of the digest is one application of a plurality that is *registered to use wireless network communications*. EX1003, ¶170.

c) Motivation to combine Lundblade and Hardjono

A POSITA would have been motivated and found it obvious to modify Lundblade's network system to perform Hardjono's software component verification. EX1006, [0101]-[0102]. A POSITA would have understood this modification is nothing more than combining prior art elements (*i.e.*, Lundblade's secure network communication and Hardjono's software verification) according to known methods to yield predictable results of increasing the security of the network by verifying the integrity of software applications loaded onto client 300 prior to granting said applications access to resources. *KSR*, 550 U.S. at 416; EX1003, ¶171. Hardjono provides an express teaching, suggestion, or motivation for implementing the resource verification process: to "identify and authenticate components of a

client platform—both hardware and software—that are in a potentially improper state before the client platform is given access to network services or resources.” EX1006, [0008]. Given Lundblade’s client 300 contains downloaded applications, a POSITA would appreciate performing Hardjono’s resource verification procedure to ensure a downloaded application is not corrupted, malicious, or otherwise non-secure before providing the client access to a specific network resource requested by the application. EX1004, 11:20-23, 10:43-46; EX1003, ¶172. Further, using credentials 312 taught by Lundblade to create the trust report constitutes combining prior art elements to yield a predictable result (a trust report containing a device credential). EX1003, ¶172.

Additionally, a POSITA would have been motivated to use the credential-based, secure communication link taught by Lundblade to secure all future communications between the server and client, including for signals used to request a network resource, and for the trust report sent back to the device. EX1003, ¶173. Using the same link reduces resource requirements when a client registers to network 104 and wants to access a network resource once registered, increasing efficiency over creating a new link each time. *Id.*

Further, the modification would have had a reasonable expectation of success, given Hardjono teaches authentication entity 145 components and integrity authority 140 being implemented on the same server. EX1006, [0031]. Additionally, as

discussed at Claim 1[f], Lundblade teaches a method of authenticating device signals for secure communication to establish a secure message link. Thus, the modification only requires implementing authentication entity 145 and integrity authority 140 on server 400 of Lundblade, which would include programming client 300 to generate and send an integrity report 170 (including a digest for at least one software application) to server 400 over the secure message link when requesting access to a network resource. EX1003, ¶174. With authentication entity 145 and integrity authority 140 being implemented on server 400 of Lundblade, the modified server would be programmed to compare the digest (which includes a hash) of the at least one software application in the integrity report 170 to digests in the integrity records 320 of the integrity authority 140, generate a trust report 220 based on the digest and device credentials 312, and then send the trust report to the client 300 over the secure message link for subsequent use in accessing network resources. EX1003, ¶175. Similarly, generating the trust report using credentials 312 would have been a basic software modification to a POSITA. It would have been well within the skill of a POSITA to implement this functionality on the server 400 of Lundblade. *Id.*

A POSITA would have understood implementing the software verification teachings of Hardjono would not interfere with implementing the device identifier verification or server memory organization taught by Jobst, at least because the modifications are independent and occur at different parts of Lundblade's process.

EX1003, ¶176. A POSITA would have the knowledge required to implement the teachings into a single system without undue experimentation. *Id.*

9. 1[h]

Lundblade in view of Hardjono discloses this element. *See* Claim 1[g]. Hardjono discloses generating trust report 220 based on the component (the application), the digest, and the TPM certificate in response to client 110's network resource request that includes the integrity report (*i.e., in response to the second request*). *See* Claim 1[g]; EX1003, ¶177. In the same way as the '055 Patent describes generating a signed application credential using the application credential and device credential, Hardjono's trust report 220 is based on the TPM certificate and digest (hash) of one or more applications. EX1001, 57:43-56; EX1003, ¶178. As discussed in Claim 1[g], in the combination, credentials 312 are used in generating the trust report, in addition to the TPM certificate. *See* Claim 1[g] (providing the Motivation to Combine for this modification). As taught by Hardjono, it would have been known that other credentials, like the credentials of Lundblade, could be utilized and placed in the integrity report 170.

Further, a person skilled in the art will also recognize that integrity report 170 can be signed by a certificate derived from a TPM-bound certificate *or signed by other certificates, not necessarily TPM certificates.*

EX1006, [0076]; EX1003, ¶179. Thus, Lundblade-Hardjono teaches generating a trust report (*generating...a first secured application credential unique to the first device*), including the credentials 312 of the client, in response to the request. The trust report 200 disclosed in the Lundblade-Hardjono combination is *unique to the first device* because the trust report 220 is generated using credentials 312 unique to the client 110. EX1003, ¶179 (opining that trust report 220, being based on a credential of client 110, is thus unique to client 110). Thus, the trust report (*secured application credential*) is *based on* (1) the hash of an application (*the identified first application*) and (2) credentials provided in the request (*the secure device credential*).

10. 1[i]

Lundblade in view of Hardjono discloses this element. *See* Claim 1[g]. As discussed in Claim 1[g], the Lundblade-Hardjono combination sends trust report 220 to client 300 to be used to access other resources or services.

In another embodiment, authenticator entity 145 can provide trust report 220 to client platform 110. This can allow client platform 110 to use trust report 220 as a sort of “trust ticket” to access other network resources or services. The other network resources or services can also have an authenticator entity to check the validity of trust report 220. A person skilled in the art will recognize that any number of resources or services can be accessed by client platform 110 using trust report 220.

EX1006, [0041]. In the combination, the trust report 220 is transmitted over the link established via Lundblade's secure communications used for all communications between client 300 and server 400. *See* Claim 1[g] (providing the Motivation to Combine); EX1003, ¶180. Therefore, the Lundblade-Hardjono combination teaches sending the trust report (*sending the first secured application credential*) to the client (*first device*) over secure communications (*the secure message link*).

B. Claim 2

As discussed in Claim 1[g], Lundblade's server 400 is modified by Hardjono to implement authentication entity 145 and integrity authority 140 processes, which includes generating trust report 220 (*secured application credential*). In the modified system, because the trust report 220 includes a reference to credentials 312 (*secure device credential*) and a reference to the software component digest (containing hash) (i.e., *general application credential*), generating the trust report in the manner disclosed by the Lundblade-Hardjono combination comprises creating a combination of the digest and the credentials. EX1003, ¶181 (opining a reference to the device credential and the digest being together within the trust report creates a combination of the information).

C. Claim 6

Lundblade-Jobst-Hardjono teaches generating a trust report (*secured application credential*). *See* Claim 1[g]-[h]. The trust report 220 is generated in

response to client 300 requesting access to a network resource for a software application on the client. *See* Claims 1[g]-1[h]. The software application (i.e., software component 160) is already installed on client 110 and executable. EX1006, [0037], [0095]. Hardjono discloses resource 150 may be a web service or application. EX1006, [0031]. For example, tiered access to bank account information is granted to the client depending on the trust score 225 included in trust report 220. EX1006, [0044]. A POSITA would have understood, in the banking scenario, the resource is access to bank account information for banking software on the device. EX1003, ¶182.

Additionally, Hardjono discloses trust report 220 being a “trust ticket” that provides access to “network resources or services.” EX1006, [0041]. A POSITA would have understood in using the “trust ticket” (i.e., trust report 220) to access network resources including, *e.g.*, accessing a printer to print something or accessing account information in a banking application, the trust report disclosed by the Lundblade-Hardjono combination is a *run-time application credential* because it is generated for an application attempting to obtain a resource when it is executing. EX1003, ¶183 (*citing* EX1001, 54:43-47 explaining a run-time application credential in the ’055 Patent is “for an application that intends to initiate execution and obtain access network service.”).

D. Claim 7

Lundblade-Jobst-Hardjono teaches obtaining a trust report at a client. *See* Claim 1[h]. Trust report 220 is a run-time application credential. *See* Claim 6. Hardjono teaches the trust report is a “‘trust ticket’ to access other network resources or services” when sent to the client after being generated. EX1006, [0041]. The Lundblade-Jobst-Hardjono combination teaches that every communication made by the client over the network is a secure communication via the secure link (*secure message link*) between the client security module (*device link agent*) of the client and a network server. *See* Claim 1[f], [g], [i]. A POSITA would have understood that to access a network resource or service with the trust report stored on the client, the trust report is sent to server 400 of the combination using secure communications (i.e., *receiving, from the device link agent over the secure message link, the run-time application credential*). EX1003, ¶184.

Further, trust report 220 is evaluated by a network server. Hardjono teaches when using the trust report as a “trust ticket,” “[t]he other network resources or services can also have an authenticator entity to check the validity of trust report 220.” EX1006, [0041]. In the combination, the authenticator entity is incorporated into server 400. *See* Claim 1[g]. A POSITA would have understood that the server 400 thus receives the trust report and determines if the trust report is valid

(evaluating the run-time application credential at an element of the network system)

prior to granting access to requested network resources. EX1003, ¶185.

E. Claim 8

Hardjono teaches, upon evaluation of Hardjono's trust report 220 (i.e., *in response to evaluating the run-time application credential*), authentication entity 145 determines, for example, what banking privileges are allowed for a banking application requesting network resource access. Trust score 225 contained within the trust report is evaluated against thresholds. EX1006, [0044]. Each threshold allows the device access to particular resources or information for the application. EX1006, [0044], [0100]; EX1003, ¶186 (opining Hardjono's threshold comparison being used for a banking application, as well as other applications). In the banking application scenario, when a lower threshold is met, the client is only enabled to view account balances. EX1006, [0044], [0100]. For example, when a higher threshold is met, the client is enabled to view account balances and perform money transfers. EX1006, [0044], [0100]. The thresholds are driven by the policy server 215 and are specific to a given institution. EX1006, [0044]. Hardjono teaches policy server 215 (incorporated into server 400) determining whether to grant or deny access to a resource based on the predetermined policy. EX1006, [0104]. The grant/deny result is forwarded to the client. *Id.* A grant or deny result is, itself, a *network policy setting* because it “identif[ies] conditions in which a network access

policy should be implemented”; for Hardjono’s grant and deny responses, the network access policy would never be implemented (*i.e.*, the policy enforcement agent would never be invoked to restrict access to the resource) or always be implemented (*i.e.*, the policy enforcement agent would always be invoked to restrict access to the resource), respectively. EX1001, 13:39-45; EX1003, ¶187. Furthermore, Hardjono discloses the forwarded result may inform client 110 that a “required remediation” be performed, such as re-installing the applications or providing “reduced” access to the network resource or service. EX1006, [0099]. For Hardjono’s “required remediation” response, the network policy setting specifies the conditions under which the network access policy would be implemented (*i.e.*, the policy enforcement agent would be invoked to restrict access to the resource until the required remediation has taken place). EX1003, ¶188.

Because the grant/deny result comes from a policy server 215 and specifies the level of access to a network resource or service, the grant/deny result disclosed by the Lundblade-Jobst-Hardjono combination is a *network policy setting*. EX1003, ¶¶189-190. In the modified system, server 400 contains authentication entity 145 functionality, which includes policy server 215. *See* Claim 1[g] (providing Motivation to Combine). Thus, server 400 of Lundblade-Jobst-Hardjono *return[s] a network policy setting over the secure link established via Lundblade’s secure communication (i.e., over the secure message link to the device link agent)* in

response to evaluating the trust score 225 in the trust report 220 (i.e., *in response to evaluating the run-time credential*).

A POSITA would have understood the grant/deny result is *applicable to the first application* (e.g., the banking application) because the trust report (i.e., *secure application credential/run-time application credential*) and, by extension, the trust score is created from the digest of the application. *See* Claim 1[g]-[h]; EX1003, ¶191. Further, a POSITA would have understood that the grant/deny result (*network policy settings*) would only be applicable to the specific application requesting the resource. EX1003, ¶192. For example, a POSITA would have understood banking thresholds would not apply to other applications, such as an email package. *Id.*

F. Claim 9

The Lundblade-Jobst-Hardjono system receives and evaluates a run-time application credential. *See* Claim 7. The evaluation is done by server 400 of Lundblade, which incorporates the functionality of authenticator entity 145 and integrity authority 140 of Hardjono, including profiles database 235, verification server 205, authentication server 210, and policy server 215. EX1006, [0035], FIG. 16; *see* Claim 1[g]. Thus, server 400 of the Lundblade-Jobst-Hardjono combination is an authentication server (*wherein the element of the network system is an authentication server*). EX1003, ¶193. Hardjono teaches the authentication entity is coupled to the profiles database 235, which stores trust reports 220 (*storing secured*

application credentials, including the first secured application credential) along with other information. EX1006, [0043], FIG. 2. Therefore, server 400 of the combination is likewise coupled to the profiles database (an *application credential database coupled to the authentication server*). EX1003, ¶¶194-195.

G. Claim 10

The Lundblade-Jobst-Hardjono client 300 receives a trust report 220 from server 400 to access network resources for an application. *See* Claim 1[g]-[i]; EX1006, [0041]. Hardjono discloses the trust report does not provide access to resources for an unlimited amount of time. Specifically, periodic reevaluation may be performed “according to a corporate policy.” EX1006, [0079]. A POSITA would have understood a company policy, disclosed by the Lundblade-Jobst-Hardjono combination, outlines events that require the system to perform reevaluation before providing access to resources. EX1003, ¶196. For example, when a laptop computer connects to a network external to a corporate network during a business trip, it may become tainted by malware, which may subsequently be introduced to the network. EX1006, [0006]. Thus, a POSITA would have recognized the benefit of a corporate policy requiring reevaluation after a device connects to an external network. EX1003, ¶196. A POSITA would have understood obtaining a new trust report 220 at the occurrence of certain events described in a company policy (*device event[s]*) would better protect the system from malware. *Id.* A POSITA would have

understood a company policy would include requiring reevaluation after a set period of time, device connection to an external network, or application updates. EX1003, ¶197 (discussing trust score includes verification date/time and contemplates software versions resulting in different integrity scores).

Hardjono discloses reevaluation is performed in the same manner as the initial evaluation (device requested), or the authenticator entity may request an integrity report for the client. EX1006, [0079]. For the reasons discussed above (i.e., increasing network system security by verifying components to keep malware from affecting the network), the Lundblade-Jobst-Hardjono system would reevaluate based on events outlined in a company policy (i.e., *based on a first device event*) and generate a new trust report. *See* Claim 1[g] (providing Motivation to Combine). A POSITA would have understood a new trust report is returned to the device after the reevaluation (*refreshing the first secured application credential to create a different secured application credential unique to the first device, and sending the different secured application credential to the first device*). EX1003, ¶198. Further, it would have been understood the old trust report is no longer valid, and the new trust report is used to access resources (*the different credential superseding the first secured application credential*). EX1003, ¶199.

H. Claim 11

Lundblade is modified by Hardjono to implement the resource verification procedure via authentication entity 145 and integrity authority 140 functionality on server 400 to request a network resource (*second request*) and generate a trust report (*secured application credential*). See Claim 1[g]-1[h]. As part of generating the trust report 220 for the software component, the digest (*general application credential*) of the component (application) in the integrity report is compared to a digest of that application in the integrity records 320 in a global signature database 325 (*a known-application credential for the first application*). EX1006, [0045].

More specifically, integrity authority 140 harvests raw information 305 about components 310 from their sources (e.g., manufacturers, software vendors, and hardware vendors, among other possibilities). ***As previously mentioned, the digests can be obtained from off-the-shelf products by harvesting records from a purchased version, directly from the manufacturer by uploading or otherwise obtaining the information through a direct means, or using other techniques.***

EX1006, [0045]. The digest in record 320 is a *known-application credential for the first application* because it is harvested from the software source and includes a hash.

EX1001, 58:11-14 (describing the known-application credential as a “hash for the known application program”); EX1003, ¶200. Given the digest received from the client is compared to the known digest stored at the integrity authority (either locally or in a global database), a POSITA would have understood the comparison resulting

in a match verifies the digest included in integrity report 170 matches the known digest (*verifying that the general application credential matches a known-application credential for the first application*). EX1003, ¶201; EX1006, [0038], [0070], [0085]. Thus, the Lundblade-Jobst-Hardjono system verifies the integrity report 170 digest against a known digest for the first application, in *response to the second request*.

I. Claim 13

The Lundblade-Jobst-Hardjono combination performs the resource verification procedure by comparing a digest within integrity report 170 to integrity records 320 while generating trust report 220. *See* Claim 11. Hardjono teaches the digests stored in integrity records 320 are formed from raw information 305 of software components coming “from their sources (e.g., manufacturers, software vendors, and hardware vendors, among other possibilities).” EX1006, [0045]. A POSITA would have understood a “software vendor[]” is an app store, because a software vendor sells various software applications (“apps”) in a marketplace (the “app store”). EX1003, ¶202 (citing EX1006, [0045] to explain a POSITA would understand Hardjono’s desire for “the ‘chain of custody’ of the component to be as secure as possible” would teach obtaining the credential directly from the app store for those apps purchased from the app store). Thus, Lundblade-Jobst-Hardjono’s

system *obtain[s] the known-application credential* (i.e., hash of record 320 generated from digests of harvest information 305) *from an app store*.

J. Claim 14

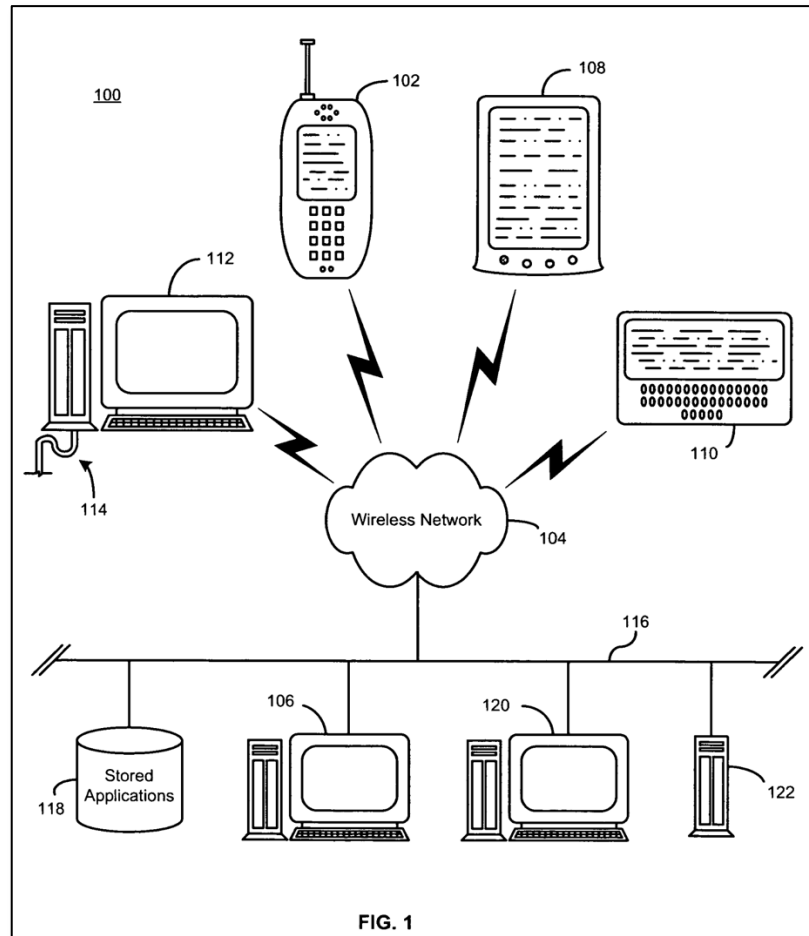
Lundblade-Jobst-Hardjono teaches performing verification of characteristics 310 against expected characteristics stored at server 400. *See* Claim 1[b]. Lundblade's characteristics 310 include device identifiers as discussed *supra*, and subscriber identifiers (e.g., mobile directory number (MDN), IMSI, etc.). EX1004, 12:29-42. The MDN is the 10-digit phone number associated with a mobile device user. EX1003, ¶203. Jobst teaches, when validating the device ID (i.e., IMEI), the software provider 33 "can find the phone number of the requesting phone 1 based on the IMEI code." EX1005, 7:15-22. Given the IMEI is validated and used to find the phone number, a POSITA would have understood the verification of the Lundblade-Jobst-Hardjono combination verifies both a device ID (e.g., SID) and subscriber ID (i.e., MDN) (*the verification step further comprising evaluating a subscriber identifier associated with one or more wireless end-user devices, including the first device*). EX1003, ¶203. A POSITA would have been motivated to apply Jobst's verification step to Lundblade's characteristics 310 for the same reasons provided at Claim 1[b], including verifying both the device identifier and a subscriber identifier.

K. Claim 15

Lundblade teaches generating credentials 312 based on “any one or more of” a list of characteristics 310, including both device identifiers (e.g., ESN, SID) and subscriber identifiers (e.g., removable user identity module (RUIM), MDN, subscriber billing account, etc.). EX1004, 12:29-42, 15:53-60; EX1003, ¶204 (opining these are exemplary subscriber identifiers). Thus, Lundblade teaches credentials 312 generated at least in part from characteristics 310 that include both device identifiers and subscriber identifiers (*generating the secure device credential based...on or more of the device identifiers and also on the subscriber identifier*).

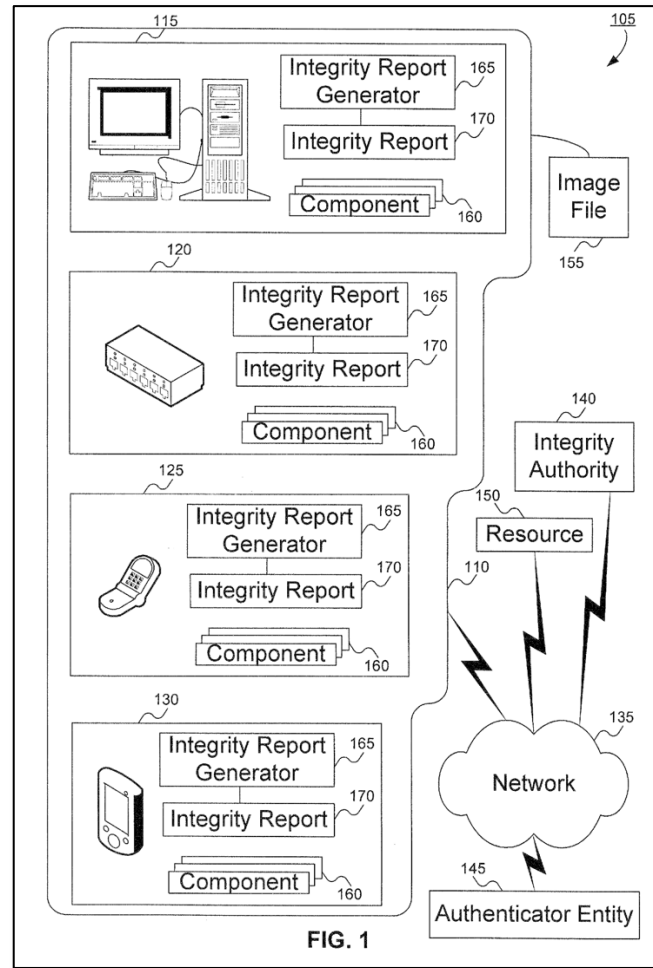
L. Claim 18

Lundblade teaches a plurality of wireless devices being connected to the network system, including mobile phones 102, personal computer 112, personal digital assistant 108, etc., and any of these devices being used as client 300. EX1004, 10:19-37, 15:63-65, FIG. 1.



EX1004, FIG. 1. Lundblade also teaches characteristics 310 used in generating the credentials 312 include subscriber identifiers (e.g., IMSI, user ID, etc.). EX1004, 12:19-42.

Hardjono teaches performing the resource verification on various types of wireless devices connected to network 135, such as personal computer 115, wireless phone 125, personal digital assistant 130, etc. EX1006, [0030], FIG. 1.



EX1006, FIG. 1. Thus, a variety of devices, such as a personal computer, wireless phone, or PDA utilize the invention of Hardjono. EX1006, [0030]. A POSITA would have understood in a situation where a user has both a company laptop and company phone, the same application is downloaded onto both, and a unique trust report would be generated for each device (*a first subscriber is associated with both the first device and with a second wireless end-user device*). EX1006, [0095], [0033]; EX1003, ¶206. Additionally, the trust report is unique to the device requesting the

network resource, as the trust report is based on the device certificate. EX1006, [0068].

Thus, Lundblade-Jobst-Hardjono teaches the device credentialing procedure may be applied to multiple different devices (mobile phone, personal computer, personal digital assistant) for a user, and the resource verification procedure being performed for an application on any of those devices. Further, a POSITA would have understood an individual often owns and uses multiple devices with the same applications. EX1003, ¶207 (explaining for many applications, such as the email package discussed by Hardjono, a user would commonly have the email package installed on multiple devices, such as a personal computer and mobile phone).

Given (1) Lundblade-Jobst-Hardjono's network contains multiple devices, (2) performs resource verification on multiple devices, and (3) associates devices with a subscriber (by including subscriber ID in characteristics 310), a POSITA would have understood that the system disclosed by the Lundblade-Jobst-Hardjono combination generates a new device credential for a second wireless device associated with the same user and performs the resource verification process for the same application (e.g., email package) downloaded onto the second device to generate a trust report (*second secured application credential*) unique to the second device. EX1003, ¶208. Thus, in the Lundblade-Jobst-Hardjono system, a second trust report (*second secured application credential unique to the second device*) is

generated for the same user (*first subscriber*) for a second device (*second wireless end-user device*), based on the same application (*first application*) requesting a resource and the credentials of the second device (*second secure device credential associated with the second device*).

M. Claim 19

Lundblade-Jobst-Hardjono generates trust report 220, which includes a combination of credentials 312 and the component digest. *See* Claim 2. The digest includes a hash of the component. *See* Claim 1[g]. Because the digest included in trust report 220 contains a hash, the *combination comprises a hash*. EX1003, ¶209.

VI. GROUND 2: CLAIM 3 IS OBVIOUS OVER LUNDBLADE, JOBST, HARDJONO, AND ZUCCHERATO

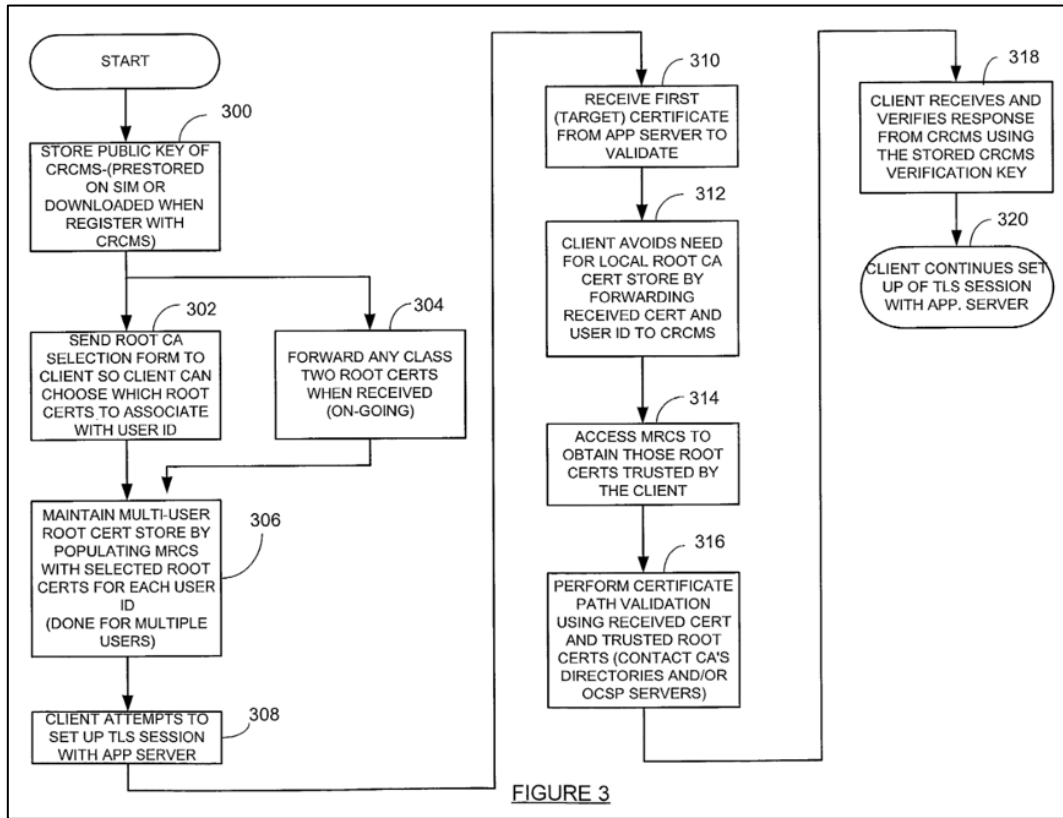
A. Claim 3

Lundblade-Jobst-Hardjono teaches server 400 receiving a request for secure communications from client 300, verifying the characteristics 310 within the request, and returning credentials 312 to the client. *See* Claim 1[Pre]-[e], *supra*. While Lundblade-Jobst-Hardjono does not expressly discuss establishing the authenticity of the network system as a part of the communication between the first device and the network system, a POSITA would have understood that such connections are commonly performed over protocols such as SSL/TLS, and that (as a part of establishing a connection using such a protocol), a digital certificate is commonly used to establish the authenticity of the server side of the connection. EX1003, ¶210

(explaining that such is common practice to prevent man-in-the-middle attacks). Indeed, during the verification process, Hardjono contemplates the server (as well as the client) both “exchanging” certificates over, e.g., SSL/TLS protocols to establish a secure channel connection. EX1006, [0101]. A POSITA would thus be motivated to seek out a reference like Zuccherato that provides details on how to set up a secure link. EX1003, ¶210.

a) Zuccherato’s Teachings

Zuccherato teaches a method of setting up a transport layer security session (TLS session) between a mobile device 202 and a server 216. EX1007, [0031]. Zuccherato teaches that when a wireless mobile device 202 attempts to set up a TLS session with a server, the server 216 sends a certificate 252 to the device 202 (*returning to the first device a network system credential*). EX1007, [0032]. The device forwards this certificate to a certificate management server 204 that checks the validation of the certificate 252 by constructing a certificate chain (*verifiable through a trusted certificate authority*). EX1007, [0032]-[0033]. If the certificate 252 is valid, the server 204 sends the device 202 a validation response, and the device continues setting up the TLS session with the server 216. EX1007, [0033].



EX1007, FIG. 3.

b) Combination of Lundblade, Jobst, Hardjono, and Zuccherato

The Lundblade-Jobst-Hardjono-Zuccherato combination modifies the secure communication process taught by Lundblade-Jobst to include an additional step of sending a certificate to the client 300 for validation at a certificate management server 252, which is also part of the network system. If the certificate is determined to be valid, the secure communication establishment process continues. Specifically, the registration request is sent by client 300, the server certificate is validated, and then the verification process taught by Jobst is performed. If the characteristics 310 are verified, the credentials 312 are generated and sent back to the client 300 to

establish secure communications as taught by Lundblade. *See* Claims 1[a]-1[e]; EX1003, ¶212. Thus, Zuccherato's validation process is performed *in response to receiving the first request and prior to performing the verification step*.

c) Motivation to Combine with Zuccherato

A POSITA would have found it obvious and been motivated to modify Lundblade's network system and process to include Zuccherato's verifiable server certificate. EX1003, ¶213. A POSITA would have been motivated to modify the network system of Lundblade-Jobst-Hardjono to send client 300 a certificate after receiving a registration request and before completing device verification to allow a client to confirm that the network server is legitimate (and not, for example, a man-in-the-middle). EX1003, ¶214. Indeed, a POSITA would appreciate that ensuring the server is a trusted server is important to the security of the entire process. Otherwise, any credential generated and sent to the device may be compromised itself or cause the device to become compromised if used. EX1003, ¶215. A POSITA would have understood that this modification is nothing more than combining prior art elements (*i.e.*, Lundblade's network system and Zuccherato's verifiable server certificate) according to known methods to yield predictable results of increasing the security of the network. *KSR*, 550 U.S. at 416.

Further, a POSITA would have had a reasonable expectation of success in modifying Lundblade's system, given that networks were known to send a credential

to a device when a device attempts connection in protocols like TLS and that certificate authorities were well-known. EX1003, ¶216. Thus, all that is required is making simple programming changes to server 400 and client 300 to implement the validation process using the well-known certificate architecture of the Internet at the time, which would not require undue experimentation. *Id.*

VII. GROUND 3: CLAIM 4 IS OBVIOUS OVER LUNDBLADE, JOBST, HARDJONO, AND CHEN

A. Claim 4:

Lundblade teaches that generating credential 312 for client 300 includes applying cryptographic algorithms, such as a one-way cryptographic algorithm, or “different cryptographic algorithms.” EX1004, 12:51-60. Further, Lundblade discloses that credentials 312 are formulated through a one-way cryptographic operation on characteristics 310 and a client key.

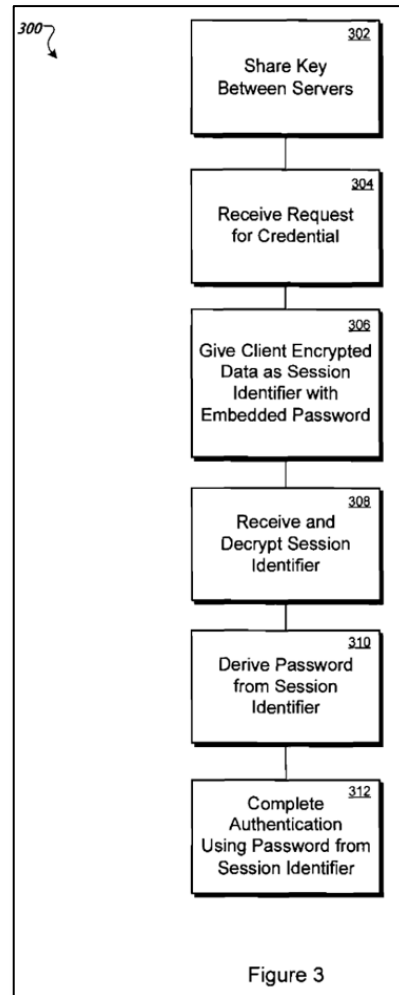
Further, in at least one embodiment the at least one credential based on both the received characteristics 310 of the client computing device and the unique client key (320) *is formulated by performing a one-way cryptographic operation on characteristics 310 of the client computing device 300 and the client key.* In some embodiments other credentials are formulated using the same or different cryptographic algorithms, while in other embodiments certain credentials are not formulated using a cryptographic algorithm.

EX1004, 12:51-60. Based on Lundblade’s disclosure of “different cryptographic algorithms,” a POSITA would be motivated to seek out a reference teaching ways

of encrypting data, such as public-key algorithms, shared-secret-key algorithms, and keyed hashes. EX1003, ¶217. Chen is one such reference.

a) Chen's Teachings

Chen teaches a credential server for generating a password and encrypting a password using a key for establishing secure communication (*generating a secure device credential including encrypting the credential with a key known by at least one element of the network system*). EX1008, 8:16-59, FIG. 3. The credential server receives a request for a credential from a client. EX1008, 8:35-36. In response, the credential server generates a password. EX1008, 8:38-51. The password is encrypted using a 128-bit key to generate a 16-byte binary ciphertext. EX1008, 8:52-58. This text is transmitted to the client for use in secure communications with servers that possess the key, allowing for secure communications. EX1008, 8:58-62.



EX1008, FIG. 3.

b) Combination of Lundblade, Jobst, Hardjono and Chen

The combination of Lundblade and Chen uses the process of generating credentials 312 as taught by Lundblade, except that the generated credentials are encrypted with a known key before being sent to the device as taught by Chen. Specifically, the unique client key 409 and generic key 410 of Lundblade are modified to be 128-bit keys, and the server 400 of Lundblade uses these keys to encrypt the credentials before they are sent to the client. EX1003, ¶219.

c) Motivation to Combine with Chen

A POSITA would have found it obvious and been motivated to modify Lundblade's server 400 to encrypt credentials 312 using the unique client key 409 and generic key 410 prior to sending them to the client as taught by Chen. EX1008, 8:35-57; EX1003, ¶¶220-222. Such a modification is nothing more than the use of a known technique (encryption) to improve methods (establishing secure communication) in the same way, by increasing the security of the credentials 312 sent to client 300 and saved in memory 402 of server 400. *KSR*, 550 U.S. at 417-18; EX1003, ¶220; EX1004, 15:11-13, 17:4-6. Lundblade already focuses on securing the credentials 312 by using a "secure" mode when sending credentials to the client 300 and by generating the credentials using cryptographic algorithms. EX1004, 17:4-6, 12:51-60. Furthermore, though Lundblade teaches using one-way cryptographic algorithms for generating credentials 312, a POSITA would appreciate that additionally encrypting the credentials increases the security of the credentials because only devices with access to the relevant key can decrypt them. EX1003, ¶221. For example, encrypting the credentials using a key known to the network (but not the client) would prevent a compromised client from leaking the individual components making up credentials 312, a problem Lundblade was concerned with. *See* EX1004, 2:13-41. Encryption was well-known to provide the advantage of securing data. EX1003, ¶221.

Further, a POSITA would have found a reasonable expectation of success in modifying Lundblade's network system, as server 400 already uses cryptographic algorithms and keys stored at the server to generate the credentials 312. EX1003, ¶222. Thus, all that is necessary to implement the teachings of Chen is to utilize those keys to encrypt the credentials 312 before sending them to the client, and this would not have required undue experimentation. *Id.*

VIII. GROUND 4: CLAIMS 5 AND 16 ARE OBVIOUS OVER LUNDBLADE, JOBST, HARDJONO, AND LUNDBLADE '330

A. Claim 5

Lundblade teaches negotiating a secure message link using credentials 312 between server 400 and client 300. *See* Claim 1[f]. Lundblade-Jobst-Hardjono also teaches generating and sending a trust report 220 (i.e., secured application credential) to client 300 from server 400, with the trust report being for a first identified application. *See* Claim 1[g]-1[i]. Hardjono teaches using the trust report as a “trust ticket” when the application on client 300 requests another network resource, and that the trust report may be verified by an authenticator when sent from client 300 to provide access to the other network resource. EX1006, [0041]. The server stores the trust report. EX1006, [0043]. Hardjono also teaches that such resources are used by applications on the device and may be stored on a server. EX1006, [0031]. However, Lundblade-Jobst-Hardjono does not expressly teach how this “trust ticket” is subsequently used to route messages. Accordingly, a POSITA

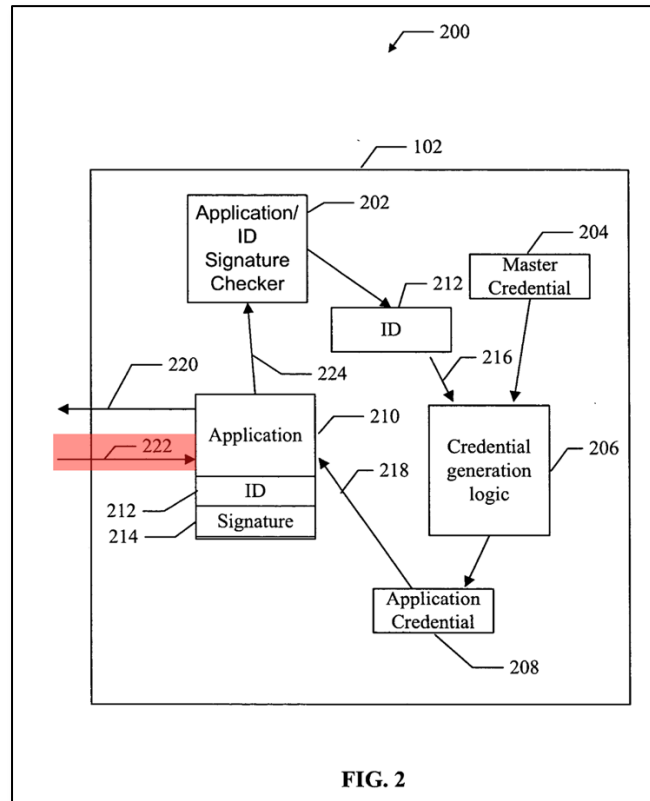
would be motivated to seek out a reference teaching the use of application credentials to access resources. EX1003, ¶223. Lundblade '330 is one such reference.

a) Lundblade '330's Teachings

Lundblade '330 teaches using an application credential to request resources, such as data, from a server. EX1009, [0048]-[0060]. The application credential is generated from an application's ID 212. EX1009, [0048]. The application credential is used to request data for use by the application on the device. EX1009, [0050]-[0051]. Specifically, Lundblade '330 teaches transmitting a secured application credential to a data server when requesting data, and if the application credential is authenticated, sending the data back to the application for which the application credential was generated. EX1009, [0051], [0059]-[0060].

At block 310, *assuming the application's data request is authenticated; the requested data is transmitted to the application 210 from the data server*, as shown by path 222. For example, the data server 110 receives the request from the application 210 and operates to authenticate the application before transmitting the requested data.

EX1009, [0051]. Because the data comes from a data server within network 104, it is a *network message*. EX1003, ¶224.



EX1009, FIG. 2 (annotated to highlight the data path to the application in a mobile device).

b) Combination of Lundblade, Jobst, Hardjono, and Lundblade '330

The combination uses the trust report of Hardjono in the manner specified by Lundblade '330 to obtain a resource from the server. Specifically, after the trust ticket is obtained from server 400 as discussed in claim 1[Pre]-[i], the trust ticket is sent to server 400 to obtain an additional resource. Upon receiving the trust ticket, the server 400 compares it to the trust ticket stored on the server (as taught by both Hardjono and Lundblade '330). If the trust ticket is verified, the resource (data) is sent to the requesting application associated with the trust ticket in the mobile device

in the manner taught by Lundblade '330. As taught by Lundblade, every communication between the client and the server is done using secure communications. Thus, the combination teaches routing a resource to the application linked to the trust report over secure communications (*routing a network message to the identified first application over the secure message link*). Further, the trust report is generated for the application requesting the resource (the digest of the application is used to generate the trust report, *see* Claim 1[g]-1[h]). Thus, a POSITA would have understood that the trust report indicates the application to receive the resource (*using the first secured application credential to indicate that the identified first application is to receive the network message*). EX1003, ¶225.

c) Motivation to Combine with Lundblade '330

A POSITA would have been motivated and found it obvious to modify Lundblade-Jobst-Hardjono's network system to route a network message (such as data requested through the network) to the identified software component in client 300 per Lundblade '330. EX1003, ¶226. Such a modification is nothing more than the use of a known technique (routing network messages to an application based on the application credential) to improve similar methods (providing software components (applications) access to network resources) in the same way by ensuring the application requesting the network resource is trusted and authentic prior to providing the requested information. *KSR*, 550 U.S. at 417-18; EX1003, ¶226. A

POSITA would have been motivated to route a network message to the device using the trust report 220 to ensure the correct (and authenticated) requesting software receives the network resource access. EX1003, ¶227.

Further, A POSITA would have had a reasonable expectation of success when making the modification, given that the Lundblade-Jobst-Hardjono system already verifies an application before granting access to a network resource (e.g., before transmitting data for that application to a device). *See* Claim 1[g]; EX1003, ¶228. Thus, all that is required to make the combination is simple programming changes to the software of server 400 and client 300 to route a requested network resource to the application specified in the trust report 220, which would not require undue experimentation. EX1003, ¶228.

B. Claim 16

See Claim 5.

IX. GROUND 5: CLAIM 12 IS OBVIOUS OVER LUNDBLADE, JOBST, HARDJONO, AND FOSBACK

A. Claim 12

The Lundblade-Jobst-Hardjono combination teaches performing a verification procedure that compares a digest within an integrity report 170 sent to server 400 with a digest in integrity records 320 stored on the server 400 as part of generating a trust report 220. *See* Claim 11. Hardjono teaches that the digests are formed from raw information 305 of software components that is harvested “from

their sources (e.g., manufacturers, software vendors, and hardware vendors, among other possibilities).” EX1006, [0045]. Additionally, Hardjono teaches that such raw information (e.g., digests) may be obtained “directly from the manufacturer by uploading or otherwise obtaining the information through a direct means, or using other techniques.” EX1006, [0045]. As discussed above with respect to Claim 13, a POSITA would understand Hardjono to teach that the credential should come from the application developer for chain-of-custody purposes. EX1003, ¶230 (citing EX1006, [0045] to explain that a POSITA would understand Hardjono’s desire for “the ‘chain of custody’ of the component to be as secure as possible” to motivate obtaining the credential from the app developer). Based on those teachings, a POSITA would be motivated to seek out a reference that teaches how an app developer may make known-good application credentials available. EX1003, ¶230.

a) Fosback’s Teachings

Fosback teaches uploading applications from a product submission and management system 104. EX1010, [0040]. The product submission and management system 104 allows an application developer (i.e., “digital product submitter”) to log in and submit new products (i.e., upload new software), edit previous submissions, review and submit contracts, monitor distribution of their digital products, and manage access permissions, email settings, and account information to be associated with the digital products. EX1010, [0054]. These

products are stored in products store 106. *See* EX1010, FIG. 1. A user may then download software from the product distribution site 102. EX1010, [0043]. A POSITA would have understood that the product submission and management site 104 is an *application developer service design center*, because the product submission and distribution site allows a developer to track the usage of the application and view access permissions. '055 Patent, 23:39-43; EX1003, ¶231.

b) Combination of Lundblade, Jobst, Hardjono, and Fosback

The combination modifies the Lundblade-Jobst-Hardjono system to incorporate the product submission and distribution site 102 of Fosback to process the submission of digital products and authorize distribution. The Lundblade-Jobst-Hardjono system harvests digests of the applications from the product submission and distribution site 102 (*the known-application credential is uploaded...via an application developer service design center*), in the same manner that the digests would be received by a manufacturer. In order to verify such applications, a POSITA would be motivated to harvest the record 320 digests from the source of download, i.e., from the product distribution site the developer used to submit the software. EX1003, ¶232. In this way, record 320 comes from as close to the source (i.e., the developer) as possible, thus making the comparison process more accurate. EX1006, [0045].

c) Motivation to Combine with Fosback

A POSITA would have been motivated and found it obvious to incorporate the production submission and distribution site 102 into the system of the Lundblade-Jobst-Hardjono combination and to receive digests from site 102. EX1003, ¶233. A POSITA would appreciate that a product submission and management site provides a convenient location to upload and manage information related to applications produced from a multitude of independent software developers (so that it may be, for example, provided to users via an app store such as product distribution site 102). *Id.* Thus, a POSITA would have been motivated to use such a site to access a large database of digests without reaching out to individual developers. *Id.* Such a modification is nothing more than applying a known technique (i.e., uploading software from a product distribution site) to a known method (i.e., resource verification via authenticator 145 and integrity authority 140), ready for improvement to yield predictable results of making software applications more easily and readily available for authentication via the resource verification procedure. *KSR*, 550 U.S. at 416; EX1003, ¶233.

A POSITA would similarly recognize that the user of client 300 would appreciate access to the variety of applications located on the product distribution site 102 while still implementing the authentication entity 145/integrity authority

140 to ensure the application has not been compromised when attempting to access network resources. EX1006, [0006], [0008]; EX1003, ¶234.

Further, a POSITA would have had a reasonable success in making such a modification, given that the modified system already harvests digests from software vendors and directly from the manufacturer, and Hardjono expressly states “other possibilities” of software sources. EX1006, [0045]; EX1003, ¶235. Additionally, the network system contains application servers 106 from which applications may be downloaded. EX1004, 10:19-26. Given these teachings, a POSITA would have found it simple to modify the system to upload software using product submission and management system 104 for distribution via product distribution site 102 (corresponding to the application servers 106). EX1003, ¶235. All that is required is connecting a product submission and management system into the network system and programming the integrity authority 140 to harvest digests from the applications available on the distribution site 102, and this modification would not require undue experimentation. *Id.*

X. GROUND 6: CLAIM 17 IS OBVIOUS OVER LUNDBLADE, JOBST, HARDJONO, AND SCARLATA

A. Claim 17

Lundblade teaches generating credentials 312 specific to client 300 for secure communication using characteristics of the device and/subscriber. *See* Claim 1[c]. Lundblade also teaches providing secure communication between a client security

module 214 of the client 300 and a server 400. *See* Claim 1[f]; EX1004, FIG. 2, 11:1-4, 11:34-39. Lundblade does not provide details about the hardware used to implement the secure mode using the client security module 214, and a POSITA would have been motivated to look to existing hardware to support Lundblade's secure file system and secure communications. EX1003, ¶236. Scarlata is one such reference.

a) Scarlata's Teachings

Scarlata provides further detail regarding a secured device credential. Namely, Scarlata teaches a processing system (such as a computer, tablet, telephone, or PDA) with a Trusted Platform Module (TPM) 30. EX1011, 3:20-32, 4:12-22. The TPM 30 includes a program code segment, a processor, non-volatile memory, and an active memory. EX1011, 8:48-59. The TPM 30 "provides security services such as attesting to the identity and/or integrity of the platform, based on characteristics of the platform." EX1011, 1:53-55. The TPM 30 receives a certificate from a privacy certification authority (CA) 76 to make a trustworthiness assertion to a remote processing system. EX1011, 7:51-57. Because the TPM contains a processor and helps provide communication across a network, the TPM is a *service processor*. EX1003, ¶237 (comparing to EX1001, 8:10-20). Furthermore, the identity credential sent back from the certification authority is used to confirm the TPM is trustworthy, indicating to a POSITA the remote processing system in a network confirms the

identity of the TPM (i.e., *service processor*) based at least in part on the identity credential. EX1003, ¶238. A third-party may rely on the identity credential “as cryptographic proof of the current state of the platform.” EX1011, 7:47-51.

b) Combination of Lundblade, Jobst, Hardjono, and Scarlata

In the combination, the process of establishing secure combinations as taught by Lundblade-Jobst (*see* Claim 1[Pre]-[f]) is modified to utilize the TPM of Scarlata. Specifically, the client security module is implemented using Scarlata’s TPM, which is responsible for secure communications “under the secure mode,” taught by Lundblade. EX1004, 17:15-17; *see also* EX1003, ¶239 (explaining how a TPM can provide for enhanced security, such as Lundblade’s secure file system). A POSITA would have understood that a TPM is a type of hardware security module that was commonly used for secure processing and communications. EX1003, ¶240 (discussing the use of TPMs as security modules in prior art references). Prior to the process taught by the Lundblade-Jobst-Hardjono combination, the TPM receives an identity credential from a privacy certification authority. EX1011, 7:40-44.

During the process taught by the Lundblade-Jobst-Hardjono-Scarlata combination, the identity credential of the TPM is provided to the server 400 in the request for credentials, along with other characteristics of the device, and used to generate credentials 312 (*wherein the secure device credential is a credential associated with a service processor on the first device*). *See* Claim 1[b]-[c]. All

secure communications between the client 300 and server 400 compare received credentials 312 (generated from the identity credential of the TPM) with stored credentials (*comprising confirming the identity of the service processor based at least in part on the secure device credential*). *See* Claim 1[f]. Subsequently, the TPM may be used to securely generate the *general application credential* (i.e., the digest (hash) within integrity report 170), with the integrity report 170 also including (device) credentials 312. *See* Claim 1[g]. The TPM identity credential (certificate) is used to confirm that the general application is authentic prior to generating the trust report (i.e., *secure application credential*). *See* Claim 1[g]-1[h]; *see also*, EX1003, ¶241 (outlining this process and describing how it prevents compromised clients from sending falsified general application credentials in requests for a secured application credential).

c) Motivation to Combine with Scarlata

A POSITA would have found it obvious and been motivated to incorporate the TPM and its functionality taught by Scarlata into the Lundblade-Jobst-Hardjono combination. EX1003, ¶242. Such a modification is nothing more than using a known technique (implementing a TPM and using an identity credential of the TPM to confirm the trustworthiness of the client) to improve similar systems (i.e., network systems providing device authentication for security) in the same way by allowing the server 400 to confirm the client 300 is trustworthy based on the identity of the

TPM. *KSR*, 550 U.S. at 417-418; EX1003, ¶242. A POSITA would have understood that this provides added security over only verifying a device identifier by later ensuring that the client-provided general application credentials are authentic (i.e., come from the actual applications on the actual device, and are not spoofed by a hacker). EX1003, ¶243. Namely, an application credential supplied by the client is generally simple and can be spoofed by malicious actors, while a credential supplied by a TPM is difficult to spoof, providing extra insurance that the device that generated the application credential (i.e., the digest) is trustworthy. EX1003, ¶243; EX1011, 7:47-57. Additionally, given that Lundblade teaches using a security module 214 in providing secure communications, a POSITA would have been motivated to include a TPM for performing the logic of security module 214 to provide “various facilities and services for enhancing the security of” the client 300. EX1011, 1:31-39; EX1003, ¶244.

Further, a POSITA would have had a reasonable expectation of success in making the required modifications. A POSITA would have found the modification simple, given that the credentials 312 are associated with other characteristics of the device and because security module 214 includes logic for providing secure communications using the credentials 312. EX1004, 12:29-46, 11:34-37; EX1003, ¶245.

Additionally, Lundblade already teaches server 400 using credential 312 to authenticate the identity of the device. EX1004, 17:56-18:5. Thus, all that is required to modify the system in operation is the inclusion of the identity credential of the TPM as a characteristic when generating the credentials 312 and signing the generated software digest within integrity report 170 with the TPM identity credential. EX1003, ¶245. Accordingly, when the server 400 compares the credentials 312 to stored credentials to authenticate signals from client 300, the server would confirm the identity of the TPM based on credentials 312. *Id.*

Further, these modifications would not have been challenging and not require undue experimentation, as processing systems, such as PCs, mobile phones, etc., were known to include TPMs, TPMs were often referred to as security modules themselves, and generating TPM-signed data to validate was known. EX1003, ¶245.

XI. CONCLUSION

For the foregoing reasons, Petitioner respectfully requests *inter partes* review of the Challenged Claims.

Respectfully submitted,

ERISE IP, P.A.

BY: /s/ Jennifer C. Bailey
Jennifer C. Bailey, Reg. No. 52,583
jennifer.bailey@eriseip.com
7015 College Boulevard, Suite 700
Overland Park, Kansas 66211
(913) 777-5600 Telephone

IPR2025-01572
U.S. Patent No. 10,064,055
(913) 777-5601 Facsimile

COUNSEL FOR PETITIONER

XII. MANDATORY NOTICES UNDER 37 C.F.R. § 42.8(A)(1)

A. Real Party-In-Interest

Petitioner is the real party-in-interest. 37 C.F.R. § 42.8(b)(1).

B. Related Matters

Patent Owner has filed approximately 75 lawsuits involving patents related to the Challenged Patent, most of which remain pending. Defendants in those lawsuits have filed a number of IPRs challenging claims for asserted patents. Petitioner submits with this Petition an Appendix of Related Matters identifying (1) a subset of the litigations involving patents related to the '055 Patent (i.e., in the same family and claiming priority to at least one common priority document) and asserted against Petitioner; and (2) a list of the litigations filed by Patent Owner involving family members of the Challenged Patent and IPRs challenging asserted patents. The Appendix of Related Matters includes the list of all IPRs and non-dismissed lawsuits.

C. Lead and Back-Up Counsel

Petitioners provide the following designation and service information for lead and back-up counsel. 37 C.F.R. § 42.8(b)(3) and (b)(4).

Lead Counsel	Back-Up Counsel
Jennifer C. Bailey (Reg. No. 52,583) jennifer.bailey@eriseip.com PTAB@eriseip.com	Robin A. Snader (Reg. No. 66,085) robin.snader@eriseip.com
<u>Postal and Hand-Delivery Address:</u> ERISE IP, P.A.	Adam Sandwell (Reg. No. 72,484) adam.sandwell@eriseip.com

7015 College Blvd., Suite 700 Overland Park, Kansas 66211 Telephone: (913) 777-5600 Fax: (913) 777-5601	Virginia A. Brown (Reg. No. 80,538) virginia.brown@eriseip.com <u>Postal and Hand-Delivery Address:</u> ERISE IP, P.A. 7015 College Blvd., Suite 700 Overland Park, Kansas 66211 Telephone: (913) 777-5600 Fax: (913) 777-5601
	Sten Larson (Reg. No. 81,518) sten.larson@eriseip.com Christina Canino (Reg. No. 83,366) christina.canino@eriseip.com <u>Postal and Hand-Delivery Address:</u> ERISE IP, P.A. 717 17th Street, Suite 1400 Denver, Colorado 80202 Telephone: (913) 777-5600 Fax: (913) 777-5601

D. 37 C.F.R. § 42.8(b)(4) – Service Information

Please address all correspondence to the lead and back-up counsel as shown above. Petitioner consents to electronic service by e-mail at the e-mail addresses provided above.

CLAIM LISTING APPENDIX
U.S. Patent No. 10,064,055 for Claims 1-19

Claim Designation	Claim Language
Claim 1[Pre]	1. A method of operating a network system, the method comprising:
Claim 1[a]	receiving a first request from any first device in a plurality of similar wireless end-user devices, the first request for a secure device credential,
Claim 1[b]	performing a verification step of one or more device identifiers supplied by the first device, and, upon a successful verification in the verification step,
Claim 1[c]	generating a secure device credential for the first device,
Claim 1[d]	associating the secure device credential with one or more device identifiers of the first device, and
Claim 1[e]	securely sending the secure device credential to the wireless end-user device;
Claim 1[f]	based at least in part on the secure device credential, negotiating a secure message link between a message link server in the network system and a device link agent in the first device;
Claim 1[g]	receiving, over the secure message link, a second request from the first device, the second request for a secured application credential, for any first application identified from a plurality of device applications registered to use wireless network communications, the second request comprising a general application credential for the first application; and
Claim 1[h]	in response to the second request, generating, based on the identified first application and the secure device credential, a first secured application credential unique to the first device, and
Claim 1[i]	sending the first secured application credential to the first device over the secure message link.
Claim 2	2. The method of claim 1, wherein generating the first secured application credential comprises creating a combination of the general application credential and the secure device credential.

Claim Designation	Claim Language
Claim 3	3. The method of claim 1, further comprising, in response to receiving the first request and prior to performing the verification step, returning to the first device a network system credential that is verifiable through a trusted certificate authority.
Claim 4	4. The method of claim 1, wherein generating a secure device credential for the first device comprises encrypting the credential with a key known by at least one element of the network system.
Claim 5	5. The method of claim 1, further comprising routing a network message to the identified first application over the secure message link, using the first secured application credential to indicate that the identified first application is to receive the network message.
Claim 6	6. The method of claim 1, wherein the first secured application credential is a run-time application credential.
Claim 7	7. The method of claim 6, further comprising receiving, from the device link agent over the secure message link, the run-time application credential, and evaluating the run-time application credential at an element of the network system.
Claim 8	8. The method of claim 7, further comprising, in response to evaluating the run-time application credential, returning a network policy setting over the secure message link to the device link agent, the network policy setting applicable to the first application.
Claim 9	9. The method of claim 7, wherein the element of the network system is an authentication server, the network system further comprising an application credential database coupled to the authentication server and storing secured application credentials, including the first secured application credential.
Claim 10	10. The method of claim 1, further comprising based on a first device event, refreshing the first secured application credential to create a different secured application credential unique to the first device, and sending the different secured application credential to the first device, the different credential superseding the first secured application credential.

Claim Designation	Claim Language
Claim 11	11. The method of claim 1, further comprising, in response to the second request, verifying that the general application credential matches a known-application credential for the first application.
Claim 12	12. The method of claim 11, wherein the known-application credential is uploaded to the network system via an application developer service design center.
Claim 13	13. The method of claim 11, further comprising obtaining the known-application credential from an app store.
Claim 14	14. The method of claim 1, the verification step further comprising evaluating a subscriber identifier associated with one or more wireless end-user devices, including the first device.
Claim 15	15. The method of claim 14, further comprising generating the secure device credential based at least in part on one or more of the device identifiers and also on the subscriber identifier.
Claim 16	16. The method of claim 1, further comprising forwarding network messages to the first application on the first device over the secure message link, based on the first secured application credential.
Claim 17	17. The method of claim 1, wherein the secure device credential is a credential associated with a service processor on the first device, the method further comprising confirming the identity of the service processor based at least in part on the secure device credential.
Claim 18	18. The method of claim 1, wherein a first subscriber is associated with both the first device and with a second wireless end-user device, the method further comprising the network system generating, based on the identified first application and a second secure device credential associated with the second device, a second secured application credential unique to the second device.
Claim 19	19. The method of claim 2, wherein the combination comprises a hash.

APPENDIX OF EXHIBITS

Exhibit 1001	U.S. Patent No. 10,064,055 (“’055 Patent”)
Exhibit 1002	File History of the ’055 Patent (the “’055 Patent File History”)
Exhibit 1003	Declaration of Charles D. Creusere
Exhibit 1004	U.S. Patent No. 7,784,089 to Lundblade et al. (“Lundblade”)
Exhibit 1005	U.S. Patent No. 6,707,915 to Jobst et al. (“Jobst”)
Exhibit 1006	U.S. Patent Application Publication No. 2007/0143629 to Hardjono et al. (“Hardjono”)
Exhibit 1007	U.S. Patent Application Publication No. 2003/0014629 to Zuccherato (“Zuccherato”)
Exhibit 1008	U.S. Patent No. 7,958,544 to Chen et al. (“Chen”)
Exhibit 1009	U.S. Patent Application Publication No. 2005/097330 to Lundblade (“Lundblade ’330”)
Exhibit 1010	U.S. Patent Application Publication No. 2009/276433 to Fosback et al. (“Fosback”)
Exhibit 1011	U.S. Patent No. 7,587,595 to Scarlata et al. (“Scarlata”)
Exhibit 1012	U.S. Patent No. 7,203,967 to Chmaytelli et al. (“Chmaytelli”)
Exhibit 1013	U.S. Patent No. 6,980,660 to Hind et al. (“Hind”)
Exhibit 1014	U.S. Patent Application Publication No. 2004/0039924 to Baldwin et al. (“Baldwin”)
Exhibit 1015	U.S. Patent Application Publication No. 2006/0002556 to Paul (“Paul”)
Exhibit 1016	U.S. Patent No. 9,143,494 to Giles et al. (“Giles”)
Exhibit 1017	U.S. Patent Application Publication No. 2008/0022103 to Brown et al. (“Brown”)
Exhibit 1018	U.S. Patent No. 8,166,296 to Buer et al. (“Buer”)
Exhibit 1019	Trusted Computing Group: Controlling Network Access and Endpoints (Nov. 2007) (“Trusted Computing Group”)
Exhibit 1020	U.S. Patent Application Publication No. 2008/0060066 to Wynn et al. (“Wynn”)
Exhibit 1021	U.S. Patent Application Publication No. 2008/0301776 to Weatherford (“Weatherford”)
Exhibit 1022	3GPP Technical Report, Version 6.2.0, Release 6 (March, 2005), (“3GPP”)
Exhibit 1023	International Patent Publication No. 01/82037 to Moore et al. (“Moore”)
Exhibit 1024	U.S. Patent Application Publication No. 2007/0198834 to Ksontini et al. (“Ksontini”)

Exhibit 1025	U.S. Patent Application Publication No. 2009/0328207 to Patel (“Patel”)
Exhibit 1026	U.S. Patent No. 8,201,251 to Ngalle (“Ngalle”)
Exhibit 1027	U.S. Patent No. 8,522,025 to Lakshmeshwar et al. (“Lakshmeshwar”)
Exhibit 1028	U.S. Patent Application Publication No. 2006/0101507 to Camenisch (“Camenisch”)
Exhibit 1029	U.S. Patent No. 5,978,833 to Pashley et al. (“Pashley”)
Exhibit 1030	U.S. Patent No. 5,935,247 to Pai et al. (“Pai”)
Exhibit 1031	U.S. Patent No. 7,984,487 to Katsube et al. (“Katsube”)
Exhibit 1032	International Patent Publication No. 2007/128134 to Matta et al. (“Matta”)
Exhibit 1033	U.S. Patent Application Publication No. 2002/0071567 to Kurn et al. (“Kurn”)
Exhibit 1034	3GPP Technical Specification, Version 5.7.0 (Release 5) (December, 2003), (“3GPP TS 24.229”)
Exhibit 1035	U.S. Patent Application Publication No. 2007/0255957 to Arnone et al. (“Arnone”)
Exhibit 1036	An Introduction to Computer Security – The NIST Handbook, https://csrc.nist.gov/publications/nistpubs/800-12/800-12-html/chapter19-printable.html (July, 2007), (“An Introduction to Computer Security – The NIST Handbook”)
Exhibit 1037	Secure Programming for Linux and Unix HOWTO, <i>David A. Wheeler</i> , Version 3.50 (2004) (“Secure Programming”)
Exhibit 1038	U.S. Patent No. 7,979,912 to Roka (“Roka”)
Exhibit 1039	U.S. Patent Application Publication No. 2007/0214370 to Sato et al. (“Sato”)
Exhibit 1040	Introduction to Programming Using Java, Version 3.1, https://math.hws.edu/eck/cs124/javanotes3/java2cpp/s2.html (June 2004), (“Introduction to Programming”)
Exhibit 1041	Cell-Phone Security Far From Airtight, Wired, https://www.wired.com/1998/04/cell-phone-security-far-from-airtight/ (April 13, 1998), (“Cellphone Security Far From Airtight”)
Exhibit 1042	U.S. patent No. 7,925,214 to Weinans et al. (“Weinans”)
Exhibit 1043	U.S. Patent No. 5,579,394 to Waldron Jr. et al. (“Waldron”)
Exhibit 1044	Telecommunications Law in the Internet Age, <i>Sharon K. Black</i> , Academic Press (2002) (“Telecommunications”)

Exhibit 1045	The TLS Protocol Version 1.0, Network Working Group (January 1999), (“TLS Protocol Version 1.0”)
Exhibit 1046	Hypertext Transfer Protocol – HTTP/1.1, Network Working Group (June 1999), (“Hypertext Transfer Protocol”)
Exhibit 1047	Digital Signatures, Gareth Branwyn, https://www.wired.com/1994/06/digital-signatures/ (1994) (“Digital Signatures”)
Exhibit 1048	TCG Specification Architecture Overview, Revision 1.4 (August 2, 2007), (“TCG Specification Architecture Overview”)
Exhibit 1049	Banking on Mobiles: Why, How, for Whom?, CGAP, No. 48, Mas et al. (June 2008), (“Banking on Mobiles”)
Exhibit 1050	SMS Banking Services: A 21 st Century Innovation in Banking Technology, Rotimi et al., Issues in Informing Science and Information Technology, Volume 4 (2007), (“SMS Banking Services”)
Exhibit 1051	Form 10-K Netscout Systems INC, 10kWizard SEC Power Search (June 5, 2006), (“Netscout Systems Inc”)
Exhibit 1052	U.S. Patent Application Publication No. 2007/0287433 to Wollersheim et al. (“Wollersheim”)
Exhibit 1053	Internet and Email Use for Work, Madden et al., Pew Research Center, https://www.pewresearch.org/internet/2008/09/24/internet-and-email-use-for-work/ (Sept. 24, 2008), (“Internet and Email Use for Work”)
Exhibit 1054	Microsoft Reveals New Windows Mobile 6 Smartphone Software, Improves World’s Fastest-Growing Mobile Operating System, Microsoft Source, https://news.microsoft.com/source/2007/02/11/microsoft-reveals-new-windows-mobile-6-smartphone-software-improves-worlds-fastest-growing-mobile-operating-sy%E2%80%A6 (Feb. 12, 2007), (“Microsoft Reveals New Windows 6 Smartphone Software”)
Exhibit 1055	International Patent Publication No. 2005/098630 to Wilson (“Wilson”)
Exhibit 1056	An Initial Security Analysis of the IEEE 802.1X Standard, Mishra et al. (Feb. 6, 2002), (“An Initial Security Analysis of the IEEE 802.1X Standard”)
Exhibit 1057	Direct Anonymous Attestation Explained, Jan Camenisch (July 2007) (“Attestation Explained”)

Exhibit 1058	Enterprise Security: Putting the TPM to Work, Trusted Computing Group (2008), (“Putting the TPM to Work”)
Exhibit 1059	U.S. Patent No. 9,112,681 to Kotani et al. (“Kotani”)
Exhibit 1060	A Security Assessment of Trusted Platform Modules, Evan Sparks, Computer Science Technical Report TR2007-597 (June 28, 2007), (“Security Assessment”)
Exhibit 1061	The Role of TPM in Enterprise Security, IBM Research Report, Sailer et al. (Oct. 6, 2004), (“The Role of TPM in Enterprise Security”)

CERTIFICATION OF WORD COUNT

The undersigned certifies pursuant to 37 C.F.R. §42.24 that the foregoing Petition for *Inter Partes* Review, excluding any table of contents, mandatory notices under 37 C.F.R. §42.8, certificates of service or word count, or appendix of exhibits, contains 13,981 words according to the word-processing program used to prepare this document (Microsoft Word).

Dated: October 3, 2025

ERISE IP, P.A.

BY: /s/ Jennifer C. Bailey

Jennifer C. Bailey (Reg. No. 52,583)
jennifer.bailey@eriseip.com
7015 College Blvd., Suite 700
Overland Park, KS 66211
P: (913) 777-5600
F: (913) 777-5601

COUNSEL FOR PETITIONER

**CERTIFICATE OF SERVICE ON PATENT OWNER
UNDER 37 C.F.R. § 42.105**

Pursuant to 37 C.F.R. §§ 42.6(e) and 42.105(b), the undersigned certifies that on March 21, 2024, a complete and entire copy of this Petition for *Inter Partes* Review and Exhibits were provided via Federal Express to the Patent Owner by serving the correspondence address of record for the '055 Patent:

Headwater Research LLC
C/O Farjami & Farjami LLP
26522 La Alameda Ave., Suite 360
Mission Viejo, CA 92691

Further, a courtesy copy of this Petition for *Inter Partes* Review was sent via email to Patent Owner's litigation counsel:

Marc Fenster (mfenster@raklaw.com)
Reza Mirzaie (rmirzaie@raklaw.com)
Brian Ledahl (bledahl@raklaw.com)
Dale Chang (dchang@raklaw.com)
Kristopher Davis (kdavis@raklaw.com)
James S. Tsuei (jtsuei@raklaw.com)
Jason M. Wietholter (jwietholter@raklaw.com)

Respectfully submitted,

ERISE IP, P.A.

BY: /s/ Jennifer C. Bailey

Jennifer C. Bailey (Reg. No. 52,583)
jennifer.bailey@eriseip.com
7015 College Blvd., Suite 700
Overland Park, KS 66211

P: (913) 777-5600
F: (913) 777-5601

COUNSEL FOR PETITIONERS

APPENDIX OF RELATED MATTERS

RELATED APPLE MATTERS

Litigation Matter	Patents-In-Suit	Related Patents
<i>Headwater Research LLC v. Apple Inc.</i> , U.S. Dist. Ct. W.TX No. 7:25-cv-00318	10,064,055 8,667,571	10,064,055 – Patent at Issue 8,667,571 – Family Member
<i>Headwater Research LLC v. Apple Inc.</i> , U.S. Dist. Ct. W.TX, No. 7:25-cv-00370	9,973,930 11,096,055 11,405,429 11,966,464 11,985,155 9,609,510	9,973,930 – Family Member 11,096,055 – Family Member 11,405,429 – Family Member 11,966,464 – Family Member 11,985,155 – Family Member 9,609,510 – Family Member
<i>Headwater Research LLC v. Apple Inc.</i> , U.S. Dist. Ct. W.TX., No. 7:25-cv-00371	9,615,192 10,321,320	9,615,192 – Family Member 10,321,320 – Family Member
<i>Headwater Research LLC v. Apple Inc.</i> , U.S. Dist. Ct., W.TX., No. 7:25-cv-00373	9,154,428 9,277,445 9,392,462 9,866,642	9,154,428 – Family Member 9,277,445 – Family Member 9,392,462 – Family Member 9,866,642 – Family Member
<i>Headwater Research LLC v. Apple Inc.</i> , U.S. Dist. Ct., W.TX, Case No. 7:25-cv-00375	8,631,102 8,799,451 8,023,425	8,631,102 – Family Member 8,799,451 – Family Member 8,023,425 – Family Member
<i>Headwater Research LLC v. Apple Inc.</i> , U.S. Dist. Ct., W.TX, Case No. 7:25-cv-00377	8,635,335 10,237,757 10,791,471	8,635,335 – Family Member 10,237,757 – Family Member 10,791,471 – Family Member
<i>Headwater Research LLC v. Apple Inc.</i> , U.S. Dist. Ct., W.TX., Case No. 7:25-cv-00379	9,143,976 9,179,359 9,647,918 8,666,364	9,143,976 – Family Member 9,179,359 – Family Member 9,647,918 – Family Member 8,666,364 – Family Member
<i>Headwater Research LLC v. Apple Inc.</i> , U.S. Dist. Ct., W.TX., Case No. 7:25-cv-00407	9,232,403 9,491,564 8,639,935	9,232,403 – Family Member 9,491,564 – Family Member 8,639,935 – Family Member

<i>Apple Inc. v. Headwater Research LLC</i> , U.S. Dist. Ct., N.D.Cal., Case No. 3:25-cv-07591	9,232,403 9,491,564 8,639,935	9,232,403 – Family Member 9,491,564 – Family Member 8,639,935 – Family Member
<i>Headwater Research LLC v. Apple Inc. et al.</i> , Unified Patent Ct., Case No. ACT 23132/2025	EP3107243	

RELATED NON-APPLE MATTERS

Litigation Matter	Patents-In-Suit
<i>Headwater Research LLC v. AT&T Inc., et al.</i> , U.S. Dist. Ct. E.TX., Case No. 2:23-cv-00397	8,589,541 9,215,613
<i>Headwater Research LLC v. Charter Communications Inc., et al.</i> , U.S. Dist. Ct. E.TX., Case No. 2:25-cv-00901	9,973,930 9,609,510 8,639,935 11,985,155 11,966,464 11,405,429 11,096,055
<i>Headwater Research LLC v. Charter Communications Inc., et al.</i> , U.S. Dist. Ct. E.TX., Case No. 2:25-cv-00904	9,179,359 9,277,445 9,609,544
<i>Headwater Research LLC v. Charter Communications Inc., et al.</i> , U.S. Dist. Ct. E.TX., Case No. 2:25-cv-00906	8,666,364 9,143,976 9,647,918
<i>Headwater Research LLC v. Charter Communications Inc., et al.</i> , U.S. Dist. Ct. E.TX., Case No. 2:25-cv-00916	8,635,335 10,791,471 10,237,757
<i>Headwater Research LLC v. Charter Communications Inc., et al.</i> , U.S. Dist. Ct. E.TX., Case No. 2:25-cv-00919	8,023,425 8,631,102 8,799,451
<i>Headwater Research LLC v. Comcast Cable Communications, LLC, d/b/a Xfinity, et al.</i> , U.S. Dist. Ct. E.TX., Case No. 2:25-cv-00903	9,973,930 9,609,510 8,639,935 11,985,155 11,966,464 11,405,429 11,096,055
<i>Headwater Research LLC v. Comcast Cable Communications, LLC, d/b/a Xfinity, et al.</i> , U.S. Dist. Ct. E.TX., Case No. 2:25-cv-00908	8,666,364 9,143,976 9,647,918
<i>Headwater Research LLC v. Comcast Cable Communications, LLC, d/b/a Xfinity, et al.</i> , U.S. Dist. Ct. E.TX., Case No. 2:25-cv-00909	9,179,359 9,277,445 9,609,544

<i>Headwater Research LLC v. Comcast Cable Communications, LLC, d/b/a Xfinity, et al.</i> , U.S. Dist. Ct. E.TX., Case No. 2:25-cv-00914	8,023,425 8,631,102 8,799,451
<i>Headwater Research LLC v. Comcast Cable Communications, LLC, d/b/a Xfinity, et al.</i> , U.S. Dist. Ct. E.TX., Case No. 2:25-cv-00917	8,635,335 10,791,471 10,237,757
<i>Headwater Research LLC v. Dish Network, Corp, et al.</i> , U.S. Dist. Ct. E.TX., Case No. 2:25-cv-00902	11,096,055 11,405,429 11,966,464 11,985,155 8,639,935 9,609,510 9,973,930
<i>Headwater Research LLC v. Dish Network, Corp, et al.</i> , U.S. Dist. Ct. E.TX., Case No. 2:25-cv-00905	9,179,359 9,277,445 9,609,544
<i>Headwater Research LLC v. Dish Network, Corp, et al.</i> , U.S. Dist. Ct. E.TX., Case No. 2:25-cv-00907	8,666,364 9,143,976 9,647,918
<i>Headwater Research LLC v. Dish Network, Corp, et al.</i> , U.S. Dist. Ct. E.TX., Case No. 2:25-cv-00920	8,635,335 10,791,471 10,237,757
<i>Headwater Research LLC v. Dish Network, Corp, et al.</i> , U.S. Dist. Ct. E.TX., Case No. 2:25-cv-00921	8,023,425 8,631,102 8,799,451
<i>Headwater Research LLC v. Google LLC</i> , U.S. Dist. Ct. W.TX., Case No. 7:25-cv-00231	10,321,320 9,615,192
<i>Headwater Research LLC v. Google LLC</i> , U.S. Dist. Ct. W.TX., Case No. 7:25-cv-00367	9,198,117
<i>Headwater Research LLC v. Google LLC</i> , U.S. Dist. Ct. W.TX., Case No. 7:25-cv-00369	11,096,055 11,405,429 11,966,464 11,985,155 8,639,935 9,609,510 9,973,930
<i>Headwater Research LLC v. Google LLC</i> , U.S. Dist. Ct. W.TX., Case No. 7:25-cv-00372	9,179,359 9,277,445 9,609,544

<i>Headwater Research LLC v. Google LLC</i> , U.S. Dist. Ct. W.TX., Case No. 7:25-cv-00374	8,666,364 9,647,918
<i>Headwater Research LLC v. Google LLC</i> , U.S. Dist. Ct. W.TX., Case No. 7:25-cv-00376	8,023,425 8,631,102 8,799,451
<i>Headwater Research LLC v. Google LLC</i> , U.S. Dist. Ct. W.TX., Case No. 7:25-cv-00378	10,237,757 10,791,471 8,635,335
<i>Headwater Research LLC v. Google LLC</i> , U.S. Dist. Ct. W.TX., Case No. 7:25-cv-00380	9,143,976
<i>Google LLC v. Headwater Research LLC</i> , U.S. Dist. Ct. N.D. Ca., Case No. 3:25-cv-07453	9,232,403 9,491,564
<i>Samsung Electronics Co., Ltd. et al. v. Headwater Research LLC</i> , PTAB Case No. IPR2024-01337	8,588,110
<i>Headwater Research LLC v. T-Mobile US Inc., et al.</i> , U.S. Dist. Ct. E.TX., Case No. 2:23-cv-00377	8,589,541 9,215,613
<i>Headwater Research LLC v. T-Mobile US Inc., et al.</i> , U.S. Dist. Ct. E.TX., Case No. 2:23-cv-00379	8,924,543 9,198,042
<i>Headwater Research LLC v. Verizon Communications Inc., et al.</i> , U.S. Dist. Ct. E.TX., Case No. 2:23-cv-00352	8,589,541 8,924,543 9,198,042 9,215,613
<i>Verizon Corporate Services Group Inc. et al. v. Headwater Research LLC</i> , PTAB Case No. IPR2024-00809	9,198,042
<i>Verizon Corporate Services Group Inc. et al. v. Headwater Research LLC</i> , PTAB Case No. IPR2024-01041	8,924,543
<i>Verizon Corporate Services Group Inc. et al. v. Headwater Research LLC</i> , PTAB Case No. IPR2024-01042	8,924,543
<i>Headwater Research LLC v. Walmart, Inc.</i> , U.S. Dist. Ct. E.TX., Case No. 2:25-cv-00961	9,198,117 10,321,320 9,615,192
<i>Headwater Research LLC v. Uber Technologies, Inc. et al.</i> , U.S. Dist. Ct. E.TX., Case No. 2:25-cv-00962	9,198,117 10,321,320 9,615,192
<i>Headwater Research LLC v. Target Corp.</i> , U.S. Dist. Ct. E.TX., Case No. 2:25-cv-00963	9,198,117 10,321,320 9,615,192
<i>Headwater Research LLC v. Supercell Oy</i> , U.S. Dist. Ct. E.TX., Case No. 2:25-cv-00964	9,198,117 10,321,320

	9,615,192
<i>Headwater Research LLC v. Amazon.com, Inc. et al.</i> , U.S. Dist. Ct. W.TX., Case No. 7:25-cv-00286	9,615,192 10,321,320
<i>Headwater Research LLC v. Amazon.com, Inc. et al.</i> , U.S. Dist. Ct. E.TX., Case No. 2:25-cv-00897	8,667,571 9,198,117
<i>Headwater Research LLC v. Tencent Holdings Ltd.</i> , U.S. Dist. Ct. E.TX., Case No. 2:25-cv-00965	9,198,117 10,321,320 9,615,192
<i>Headwater Research LLC v. Samsung Elecs. France S.A.S.</i> , Unified Patent Ct., Case No. UPC CFI 822/2025	EP2391947
<i>Headwater Research LLC v. Samsung Elecs. France S.a.i.</i> , Unified Patent Ct., Case No. UPC CFI 823/2025	EP2391947
<i>Headwater Research LLC v. Samsung Elecs. France S.A.S.</i> , Unified Patent Ct., Case No. UPC CFI 833/2025	EP3110069
<i>Headwater Research LLC v. Samsung Elecs. GmbH</i> , Unified Patent Ct., Case No. UPC CFI 820/2025	EP3110069
<i>Headwater Research LLC v. Flextronics Int'l Europe B.V.</i> , Unified Patent Ct., Case No. UPC CFI 29/2025	EP3110072
<i>Headwater Research LLC v. Motorola Mobility Germany GmbH</i> , Unified Patent Ct., Case No. UPC CFI 482/2024	EP3110072
<i>Headwater Research LLC v. Motorola Mobility LLC</i> , Unified Patent Ct., Case No. UPC CFI 127/2024	EP3110072
<i>Headwater Research LLC v. Flextronics Int'l Europe B.V.</i> , Unified Patent Ct., Case No. UPC CFI 28/2025	EP3110069
<i>Headwater Research LLC v. Motorola Mobility Germany GmbH</i> , Unified Patent Ct., Case No. UPC CFI 494/2024	EP3110069
<i>Headwater Research LLC v. Motorola Mobility LLC</i> , Unified Patent Ct., Case No. UPC CFI 149/2024	EP3110069
<i>Headwater Research LLC v. Samsung Elecs. Co.</i> , Unified Patent Ct., Case No. UPC CFI 496/2025	EP3110072