

UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE PATENT TRIAL AND APPEAL BOARD

APPLE INC,
Petitioner,

v.

CARBYNE BIOMETRICS, LLC,
Patent Owner

IPR2024-00332
U.S. Patent No. 10,929,512

**DECLARATION OF PATRICK MCDANIEL,
UNDER 37 C.F.R. § 1.68 IN SUPPORT OF PETITION FOR
INTER PARTES REVIEW**

TABLE OF CONTENTS

I.	Introduction.....	4
II.	Qualifications and Professional Experience	7
III.	Level of Ordinary Skill in the Art	10
IV.	Relevant Legal Standards	12
V.	Overview of the '512 Patent.....	14
VI.	Technology Background.....	17
	A. AUTHENTICATION.....	18
	B. CREDENTIALS AND CREDENTIAL MANAGEMENT.....	19
	C. CRYPTOGRAPHY	22
	D. CRYPTOGRAPHY IN WEB TRANSACTIONS.....	23
	E. TRUSTED PLATFORM MODULES	24
VII.	Claim Construction.....	27
	A. “ <i>restricted interface</i> ”	27
VIII.	Claims 1-6, 10-16 and 20-21 of the '512 patent are Unpatentable	29
	A. Ground 1: Claims 1-6, 10-16 and 20-21 are obvious under 35 U.S.C. § 103 over Kesanupalli, Challenger and Cheng.....	31
	1. Summary of Kesanupalli	31
	2. Summary of Challenger	33
	3. Summary of Cheng.....	38
	4. Summary of the Kesanupalli and Challenger Combination.....	42
	5. Reasons to Combine Kesanupalli and Challenger	46
	6. Reasons to Combine Kesanupalli, Challenger and Cheng.....	50

7.	Claim 1	55
8.	Claim 2	106
9.	Claim 3	107
10.	Claim 4	109
11.	Claim 5	111
12.	Claim 6	113
13.	Claim 10	115
14.	Claims 11 – 16 and 20	116
15.	Claim 21	119
IX.	Conclusion	122

I, Patrick McDaniel, do hereby declare as follows:

I. INTRODUCTION

1. I make this declaration based upon my own personal knowledge and, if called upon to testify, would testify competently to the matters stated herein.

2. I have been asked by Apple Inc. (“Apple”) to provide my expert opinion connection with the *Inter Partes* Review of U.S. Patent No. 10,929,512 (“’512 patent”) concerning whether the ’512 patent is unpatentable over certain prior art. This declaration is a statement of my opinions on issues relating to the patentability of Claims 1-6, 10-16 and 20-21 (“the Challenged Claims”) of the ’512 patent. As I explain more fully below, it is my opinion that all the Challenged Claims would have been obvious to a person of ordinary skill in the art (“POSITA”) at the time of the alleged invention.

3. I am being compensated for my work in this matter at my standard hourly rate. I am also being reimbursed for reasonable and customary expenses associated with my work and testimony in this investigation. My compensation is not contingent on the outcome of this matter or the specifics of my testimony. I have no other financial interest in this proceeding.

4. I have reviewed and considered the following documents in connection with my analysis of the ’512 patent:

a) the ’512 patent, APPL-1001;

- b) the prosecution history of the '512 patent ("’512 File History"),
APPL-1002;
- c) U.S. Patent No. 8,799,666 to Kesanupalli et al. ("Kesanupalli"),
APPL-1006;
- d) U.S. Patent No. 8,099,789 to Challener et al. ("Challener"),
APPL-1007; and
- e) U.S. Patent Publication No. 2004/0117636 to Cheng et al.
("Cheng"), APPL-1008;

5. In forming the opinions expressed below, I have considered:

- a) the documents listed above;
- b) the relevant legal standards, including the standard for obviousness, and any additional authoritative documents as cited in the body of this declaration; and
- c) my own knowledge and experience based upon my work in the field of authentication and distributed computing as described below, as well as the following materials:

APPL-1009	U.S. Patent No. 9,652,629 ("Brown")
APPL-1012	Patrick McDaniel, <i>Authentication</i> , Handbook of Computer Networks, Volume II, Chapter 171, John Wiley and Sons, May, 2007. Eds. Hossein Bidgoli.
APPL-1013	IBM Introducing Fingerprint Reader Into Laptop. Jack Germain. 2004 at https://www.technewsworld.com/story/ibm-

	introducing-fingerprint-reader-into-laptop-37017.html (retrieved Dec. 12, 2023)
APPL-1014	U.S. Patent Publication 2008/0059804 to Shah et al.
APPL-1015	Kerberos: An Authentication Service for Computer Networks. B. Clifford Neuman et al. 1994 at https://gost.isi.edu/publications/kerberos-neuman-tso.html
APPL-1016	ISO/IEC 11889-1:2009 Information technology Trusted Platform Module Part 1: Overview at https://www.iso.org/standard/50970.html
APPL-1017	TPM Main Part 3 Commands Specification Version 1.2 1 March 2011 at https://trustedcomputinggroup.org/wp-content/uploads/TPM-Main-Part-3-Commands_v1.2_rev116_01032011.pdf
APPL-1018	Infineon White Paper, <i>TPM Key Backup and Recovery For Trusted Platforms</i> , https://www.infineon.com/dgdl/Infineon-TPM_Key_Backup_and_Recovery-AP-v01_00-EN.pdf?fileId=db3a304412b407950112b41656d7203a (2006)
APPL-1019	Computer Dictionary, Second Edition, entry “data record” (1994)
APPL-1020	How to Restore Your Data After a Hard Drive Crash, Glenn Derene, Aug 4, 2011 at https://www.popularmechanics.com/technology/gadgets/how-to/a6819/how-to-restore-your-data-after-a-hard-drive-crash/
APPL-1021	U.S. Patent Application No. 2007/0100913 to Sumner et al.
APPL-1023	RFC2818, HTTP Over TLS (May 2000)
APPL-1024	Trusted Computing Group, <i>Enterprise Security: Putting the TPM to Work</i> , https://trustedcomputinggroup.org/wp-content/uploads/TPM-Applications-Whitepaper.pdf (2008)

6. Unless otherwise noted, all emphasis in any quoted material has been added.

II. QUALIFICATIONS AND PROFESSIONAL EXPERIENCE

7. My Curriculum Vitae (CV) is attached as APPL-1004. This exhibit also includes a list of my publications and the cases in which I have testified at deposition, hearing, or trial within the past four years.

8. I earned a Ph.D. in Computer Science from the Computer Science and Engineering Department at University of Michigan, Ann Arbor in 2001. I earned a Bachelor of Science degree in Computer Science from Ohio University in 1989, and a Master of Science degree, also in Computer Science, from Ball State University in 1991.

9. Since 2022, I have been Tsun-Ming Shih Professor of Computer Sciences in the School of Computer, Data and Information Sciences at the University of Wisconsin-Madison. Prior to my move to Wisconsin, I was the William L. Weiss Professor of Information and Communications Technology in the School of Electrical Engineering and Computer Science at the Pennsylvania State University in University Park, PA. I was also the director of the Institute for Network and Security Research, and founder and co-director of the Systems and Internet Infrastructure Security Laboratory, a research laboratory focused on the study of security in diverse network and computer environments. My research efforts primarily involve computer systems (including authentication), network management and authentication, systems security, and technical public policy.

10. Prior to 2017 I was an Assistant Professor (2004-2007), Associate Professor (2007-2011), Full Professor (2011-2015), and Distinguished Professor (2016-2017) of Computer Science and Engineering and the William L. Weiss Professor of Information and Communications Technology (2017-2022) at the Pennsylvania State University. Since 2004, I have taught courses in the field of computer systems, systems programming, mobile device security, networks, and network and computer security at both the undergraduate and graduate level. I created and maintained several of these courses at Penn State and Wisconsin.

11. From 2003-2009, I was also an Adjunct Professor at the Stern School of Business at New York University in New York, NY. At the Stern School of Business, I taught courses in computer and network security and online privacy.

12. I am a Fellow of the Association for Computing Machinery (ACM, the leading professional association for computer science), a Fellow Institute for Electrical and Electronics Engineering (IEEE, the leading professional association for engineering), and Fellow of the American Association for the Advancement of Science (AAAS, the leading professional association for science).

13. I am the director of the NSF Frontier Center for Trustworthy Machine Learning (CTML), a research organization exploring security and privacy of machine learning (and AI in general). Funded by the US National Science Foundation, the CTML is led by the University of Wisconsin-Madison and has

members from Stanford, the University of California, Berkeley, The University of California-San Diego, and the University of Virginia.

14. From 2013 to 2018 I was the Program Manager (PM) and lead scientist for the Cyber Security Collaborative Research Alliance (CRA). The CRA is led by Penn State University and includes faculty and researchers the Army Research Laboratory, Carnegie Mellon University, Indiana University, the University of California-Davis, and the University of California-Riverside. This initiative is a major research project aimed at developing a new science of cyber-security for military networks, computers, and installations.

15. I regularly advise students on research projections covering aspects of computer systems security. In this role, I regularly serve as an advisor to Ph.D. and master's degree candidates, several of whom have gone on to become professors at various institutions such as North Carolina State University, Purdue University, the University of Toronto, the University of Oregon, the Georgia Institute of Technology and the University of Florida.

16. Before joining Pennsylvania State University as a professor, from 1991 - 1996 I was a software developer and project manager for companies in the networking industry including Applied Innovation, Inc., and Primary Access Corporation. I was also a senior researcher at AT&T Research-Labs. As part of my duties in these industrial positions, I informed, reviewed, and formed corporate

policies and practices relating to the security of mobile devices, networks, and software systems.

17. I have published extensively in the field of authentication, mobile security, network and security management, computer systems, authentication, systems security, applied cryptography and network security. In addition to writing several articles for industry journals and conferences, I have authored portions of numerous books related to computer systems (including authentication), applied cryptography and network security. I have served on the editorial boards of several peer-reviewed journals including ACM Transactions on Internet Technology, for which I was the Editor-in-Chief. I was also an Associate Editor for ACM Transactions on Information and System Security and IEEE Transactions of Software Engineering, two highly regarded journals in the field.

III. LEVEL OF ORDINARY SKILL IN THE ART

18. I understand there are multiple factors relevant to determining the level of ordinary skill in the pertinent art, including (1) the levels of education and experience of persons working in the field at the time of the invention; (2) the sophistication of the technology; (3) the types of problems encountered in the field; and (4) the prior art solutions to those problems.

19. I understand that the '512 patent was filed September 21, 2020 and claims priority through various continuations to provisional applications filed

January 17, 2012 and December 9, 2011. My analysis reflects the views of a person of ordinary skill in the art (“POSITA”) in the field of the ’512 patent as of December 9, 2011.

20. Here, a POSITA at the time of the claimed invention would have had a bachelor’s degree in an electrical engineering, computer engineering, computer science, or a related field, and at least two years of experience in the research, design, development, and/or testing of authentication techniques, and related firmware and software, or the equivalent, with additional education substituting for experience and vice versa. A person with less formal education but more experience, or more formal education but less experience, could have also met the relevant standard for a POSITA. For purposes of this Declaration, in general, and unless otherwise noted, my statements and opinions, such as those regarding my experience and the understanding of a POSITA generally (and specifically related to the references I consulted herein), reflect the knowledge that existed in the field as of the earliest proper priority date of the ’512 patent (*i.e.*, December 9, 2011). Unless otherwise stated, when I provide my understanding and analysis below, it is consistent with the level of a POSITA prior to the earliest alleged priority date of the ’512 patent.

21. I met the definition of a POSITA in 2011. I also had a greater knowledge and experience than a POSITA. In my work discussed above,

including as a professor, I worked with graduate students and others who had the knowledge and experience of a POSITA in 2011, and I am able to render opinions from the perspective of a POSITA based on my knowledge and experience.

IV. RELEVANT LEGAL STANDARDS

22. I am not an attorney. In preparing and expressing my opinions and considering the subject matter of the '512 patent, I am relying on certain basic legal principles that counsel have explained to me. These principles are discussed below.

23. I understand that prior art to the '512 patent includes patents and printed publications in the relevant art that predate the proper priority date of the alleged invention recited in the '512 patent.

24. I have been informed that a claimed invention is unpatentable under 35 U.S.C. § 103 if the differences between the invention and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which the subject matter pertains. I have also been informed by counsel that the obviousness analysis takes into account factual inquiries including the level of ordinary skill in the art, the scope and content of the prior art, and the differences between the prior art and the claimed subject matter.

25. I have been informed by counsel that the Supreme Court has

recognized several rationales for combining references or modifying a reference to show obviousness of claimed subject matter. Some of these rationales include the following: (a) combining prior art elements according to known methods to yield predictable results; (b) simple substitution of one known element for another to obtain predictable results; (c) use of a known technique to improve a similar device (method, or product) in the same way; (d) applying a known technique to a known device (method, or product) ready for improvement to yield predictable results; (e) choosing from a finite number of identified, predictable solutions, with a reasonable expectation of success; and (f) some teaching, suggestion, or motivation in the prior art that would have led one of ordinary skill to modify the prior art reference or to combine prior art reference teachings to arrive at the claimed invention.

26. I further understand that certain factors may support or rebut the obviousness of a claim. I understand that such secondary considerations include, among other things, commercial success of the patented invention, skepticism of those having ordinary skill in the art at the time of invention, unexpected results of the invention, any long-felt but unsolved need in the art that was satisfied by the alleged invention, the failure of others to make the alleged invention, praise of the alleged invention by those having ordinary skill in the art, and copying of the alleged invention by others in the field. I understand that there must be a nexus—a

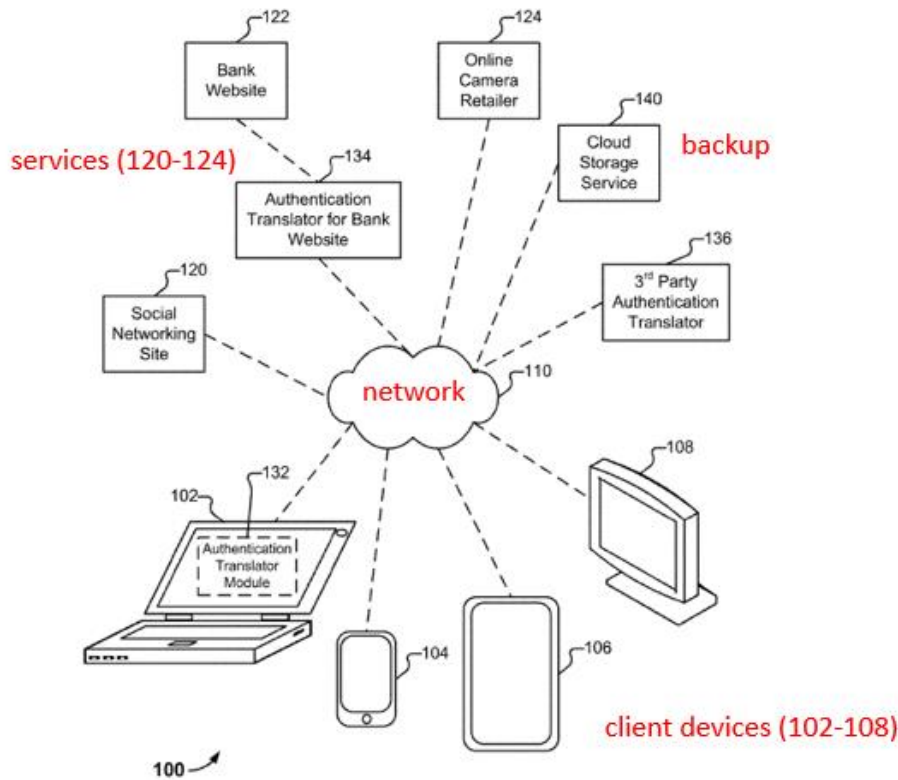
connection—between any such secondary considerations and the alleged invention.

I also understand that contemporaneous and independent invention by others is a secondary consideration tending to show obviousness.

27. I am not aware of any allegations by the named inventor of the '512 patent or any assignee of the '512 patent that any secondary considerations are relevant to the obviousness analysis of any Challenged Claim of the '512 patent.

V. OVERVIEW OF THE '512 PATENT

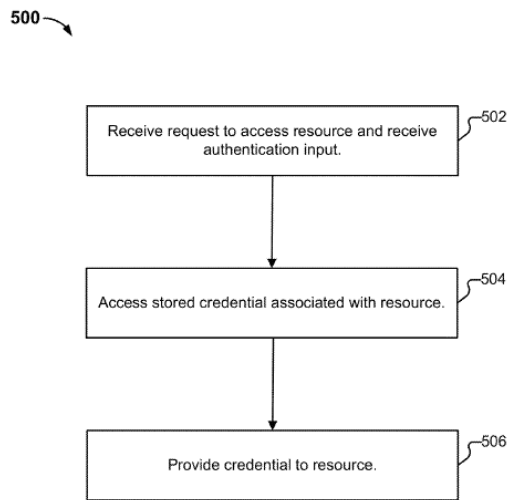
28. The '512 patent is directed to an environment including a client device, such as a notebook computer, smartphone, or tablet, which is connected via a network such as the Internet to a service such as a website. ('512 patent) APPL-1001, 2:37-56. FIG. 1 the '512 patent illustrates such an environment including client devices (102-108) connected to services (120-124).



(’512 patent) APPL-1001, FIG. 1, annotated

29. The ’512 patent describes authentication techniques to provide user credentials (*e.g.*, a password) from a client device to a service with a goal of avoiding reliance on a user’s input of the credential, which is tedious for the user and leads to poor security practices. (’512 patent) APPL-1001, 1:35-44. The ’512 patent states users “need not type such usernames and passwords into their devices whenever required by a service. Instead, users can authenticate themselves to an ‘authentication translator’ ... and the authentication translator will provide the appropriate credentials to the implicated service on the user’s behalf.” APPL-1001, 2:63-3:1. FIG. 5 of the ’512 patent illustrates a process for performing this

authentication translation. ('512 patent) APPL-1001, 1:58-59.



('512 patent) APPL-1001, FIG. 5

30. The '512 patent describes an example of FIG. 5 where a user, Alice, wishes to sign onto a social networking website 120. Alice is prompted by her client device, in this example her laptop, to provide biometric information using an “integrated fingerprint reader on her laptop.” ('512 Patent) APPL-1001, 6:18-31. Alice provides her biometric information in response to the prompt. *Id.* A translation module such as provided in her laptop compares her provided biometric information with a template stored on the computer. ('512 Patent) APPL-1001, 6:39-45. If a match is found, a previously stored credential associated with the resource is provided to the resource. ('512 Patent) APPL-1001, 6:46-52. These techniques for authentication translation were well-known by 2011.

31. I have reviewed the file history of the '512 patent, which I understand

is attached as APPL-1002. There were no office actions or prior art rejections during the prosecution that lead to the '512 patent. In a notice of allowance, the Examiner stated:

However, none of the prior art, alone or in combination teaches ***in response to determining that there is a match between the biometric input and a stored biometric template accessed by the second processor, facilitate a login of the user to the external resource at least in part by transmitting, via the established connection, output based at least in part on the at least one of the password, the cryptographic key, and the cookie retrieved from the record, and wherein the user is logged in to the external resource based at least in part on the output; and perform a secure backup of the record to a storage service, wherein a second device associated with the user is registered with the storage service, and wherein the record is downloaded from the storage service by the second device*** in view of other limitations of the independent claims.

APPL-1002, 93. However, these features, just like the other elements of the claims, were well-known to POSITA before earliest priority date of the '512 patent. As I discuss below, facilitating a login of the user to an external resource is the exact purpose of Kesanupalli, and performing a secure backup as taught in Cheng would have been obvious.

VI. TECHNOLOGY BACKGROUND

32. To provide background for my discussion of the prior art and the element-by-element analysis of the claims that follows, I present an overview of

some of the relevant state of art at existing at the time of the alleged invention of the '512 patent. The background below I discuss relevant technologies and techniques.

A. AUTHENTICATION

33. Authentication is the process of verifying the identity of a user (or system) ensuring that they are (or it is) who they claim to be. This process is essential to protecting systems and data from misuse by non-authorized entities (*e.g.*, hackers, insiders, etc.). The field of authentication for computer systems was well established by at least the 1970s, and methods and tools for its realization are well documented and widely standardized.

34. The process of authentication typically involves the presentation of a credential by a party to some system. If the credential is verified to be correct, the presenter is authenticated and allowed to use the system. For example, a user may be asked present a password to gain access to a computer. The computer then checks that the password is correct (*e.g.*, using protected data stored on that computer—sometimes represented as a password file). In this situation, the password is a credential.

35. Well before the earliest priority date of December 9, 2011, authentication was a well-established practice for users of computers and

information systems.¹ Users of devices and information systems (including websites) interacted with and relied on these authentication systems to establish their identity and ultimately their authorization to access protected information.

B. CREDENTIALS AND CREDENTIAL MANAGEMENT

36. Early authentication methods developed for computerized systems relied upon a password as a credential as described above. These systems prompted users to provide a password credential, which when provided was compared to a list of passwords stored in the system to ensure the user was who they claim to be.

37. Long before the priority date of the '512 patent, authentication methods progressed from use of a simple password to provide additional security. Authentication systems, well before the priority date of the '512 patent, were based on one or more of three kinds of credentials: things you know (*e.g.*, passwords), things you have (*e.g.*, a card), or things you are (*e.g.*, fingerprints). *See also*, (Kesanupalli) APPL-1006, 4:31-35. In all three types of authentication factors, the systems rely on assessing presented credentials to compare the presented credential

¹ APPL-1012, Patrick McDaniel, *Authentication*, Handbook of Computer Networks, Volume II, Chapter 171, John Wiley and Sons, May, 2007. Eds. Hossein Bidgoli.

against some stored information for the purposes of validation.

38. The third type of authentication factor (things you are) is referred to as biometric authentication. Biometrics as a field is well-established, with systems that measure biometrics such as fingerprints, irises, and facial recognition being widely used since the 1990s, and some biometric authentication techniques dating to the 1970s and before. Biometric systems typically use sensors such as touchscreens/surfaces, camera, or other means to measure some aspect of the user's body. Fingerprint readers identify ridges on a finger; cameras capture images that are used to estimate facial dimensions; iris scanners record unique contours and patterns in the user's eye. Prior to the priority date, biometric systems on user devices were widely used and indeed advertised as a more convenient and secure way to protect a user's electronic devices. For example, touchpads serving as biometric sensors were integrated into hardware and sold on commodity systems.²

39. When using biometrics for authentication, a set of biometric features—often called a template—such as features corresponding to a user's fingerprint are stored for example as a data file. This stored template is compared

² APPL-1013, IBM Introducing Fingerprint Reader Into Laptop. Jack Germain. 2004 at <https://www.technewsworld.com/story/ibm-introducing-fingerprint-reader-into-laptop-37017.html>

to a later biometric input (*e.g.*, a reading from a fingerprint sensor) often using a fuzzy matching algorithm. If the later biometric input is sufficiently similar to the template, the identity of the user may be deemed authentic.

40. A security architecture that commonly leveraged biometrics for authentication prior to the priority date of the '512 patent was “single sign-on” or SSO.³ In SSO, users initially login to a client device such as their computer through an authentication service relying upon password, hardware token or biometric input such as discussed above. After having logged in through the single authentication input, the user could access multiple applications or systems. That is, after a successful login, the SSO would retrieve a credential from storage on the client device or create one (*e.g.*, an authentication token) associated with the applications or systems. This credential was then used to gain access to the application or system rather than requiring the user to perform a separate authentication to each. Here, the user only logs in once, which provides access to credentials to all other supported services. Some of the earliest SSO solutions appeared in the late 1990s.⁴

41. As even the earliest authentication systems compared a user’s input

³ See, *e.g.*, APPL-1014; U.S. Patent Publication 2008/0059804 to Shah et al. [0003]-[0044].

⁴ See, *e.g.*, APPL-1015, <https://gost.isi.edu/publications/kerberos-neuman-tso.html>

with a stored credentials (*e.g.*, password lists, biometric templates), the need to protect these credentials from unauthorized access was immediately clear.

Authentication applications such as SSO increased the need to manage security of stored credentials. One typical way to protect of a user's authentication data and credentials was by using cryptography. Here, authentication-related data was (and is) almost always stored in persistent storage (*e.g.*, on hard drives or in databases) in encrypted format—and thus protected from compromised systems.

C. CRYPTOGRAPHY

42. Cryptography is the science of secret writing, which is an incredibly deep and widely studied technology. In computers, it is used to secure data and communication. Specifically—and among other things—cryptography can be used for a goal of protecting confidentiality and/or integrity (protecting against corruption) of data.

43. Perhaps the most widely used cryptographic algorithm is encryption. In its simplistic form, in some implementations, an encryption function E that takes some data (plaintext P) and a cryptographic key K_e produces new data (called ciphertext C) that hides the original data, i.e., $E(K_e, P) = C$. A decryption function D associated with the given encryption function E takes the ciphertext and a decryption key K_d and produces the original plaintext, i.e., $D(K_d, C) = P$. The important part is that the plaintext P cannot feasibly be recovered from the

ciphertext C without K_d ⁵, and therefore the encryption protects the underlying plaintext P from access by users who do not possess the decryption key K_d .

44. Encryption for protection of data such as a passwords relied on secure message authentication codes (MACs), which are functions that take a sequence of bytes as input and output a new sequence of bytes. One type of MAC is a hash-based MAC or HMAC. In HMAC, a key is additionally used as input to the MAC, so that the output of the HMAC depends both on the input bytes as well as the key value. Common (and indeed standardized) HMACs in use at the time of patent included the use of hash functions such as SHA-1 and SHA-2. Cryptographic systems of MAC and HMACs were in use in at least the early 1990s, many years prior to the priority date of the '512 patent. Regardless of what algorithm is used or what its function it performs all uses of cryptographic algorithms (e.g., encryption, decryption, hashing, HMACing) are cryptographic operations.

D. CRYPTOGRAPHY IN WEB TRANSACTIONS

45. Authentication and use of encryption techniques for web transactions

⁵ Note that in many cases, K_e and K_d are the same. Such algorithms are called symmetric key algorithms in this case. Prior to 2000s, an alternative to symmetric-key cryptography was developed that use a public-key cryptography, which uses a pair of keys: public (K^+) and private (K^-). In public-key cryptography, data encoded with either key, can be decoded using the other key.

was widely used prior to the priority date.

46. HTTP cookies were developed at least in the 1990s. An HTTP cookie is a small block of data created by a web server while a user is browsing a website. The HTTP protocol allows for the transfer of a cookie from the server to the client device. Web browsers store cookies on the client device (*e.g.*, the user's device), and the HTTP protocol allows for a Web server to request that the client send a cookie previously created by the server. When a user sends login information to a site (say, by providing a username and credential such as a password), the server can respond by setting a cookie on the client. The cookie encodes the fact that the user has logged in. Subsequently, the server can request the client's cookie(s), and when it receives the cookie, it can check whether its encoding indicates that the user has logged in. Thus, the website can continue operations without repeatedly requiring the user to provide username and password.

47. Authentication may also be performed using public-key encryption. In the 1990s, Netscape Communications developed Secure Socket Layer (SSL). SSL, which evolved into transport layer security (TLS), is the basis for secure data transfer on the Web. By 2011, much of the electronic commerce conducted was done using the SSL/TLS and public-key cryptography.

E. TRUSTED PLATFORM MODULES

48. In the early 2000s with expansion of the computing industry, there

was a need to allow personal devices to be more resistant to unauthorized access to its data.

49. By 2006, new computing systems with Trusted Platform Module (TPM) chips were being sold.⁶ A Trusted Platform Module (TPM) is a specialized microprocessor or hardware coprocessor that is designed to provide security-related functions through restricted interfaces. The typical role of a TPM in a computing system is to securely store and generate cryptographic keys, as well as to perform security related operations such as key and credential management, authentication, facilitating securing storage, encryption, decryption, and digital signatures. Here, the key is that the TPM serves to provide a secure, isolated portion of the computing environment (e.g., PC and operating system) that protects data and security-sensitive software

50. The TPM works in concert with less secure elements of the computing system (such as a general-purpose CPU or exposed interfaces or resources) and with applications running on such elements to perform security related operations for overall system. Most of the security of the TPM is derived from the isolation of security-related data and computations from the general-purpose CPU and from other less secure elements of the host machine. Because the security-related data

⁶ APPL-1016; <https://www.iso.org/standard/50970.html>

and computations are not directly exposed to a (potentially compromised) operating system and/or applications that instead interact (via the TPM) with secure storage maintained and secure operations performed by the TPM itself, the system is able to security notwithstanding the risk of programs running on less secure elements of the computing system.

51. TPM provides physical and application programming interfaces (APIs) that restrict access to security-related operations such as key and credential management, secure storage, encryption, decryption, and digital signatures. More specifically, TPM APIs are called by privileged programs, the operating system, firmware to access the security sensitive data and functions implemented within the TPM processor.⁷ The APIs allow the caller to create code that implements critical security functions for the systems and applications, *e.g.*, access securely stored cryptographic keys, encrypt sensitive data (called sealing), or obtain cryptographically strong random data. Because of the sensitivity of the functions and data in the TPM, access to these APIs is tightly controlled by the operating system (typically through permissions). By the priority date of the '512 patent, TPMs were widely used in the computer industry. By 2008, there were over 100

⁷ APPL-1017, https://trustedcomputinggroup.org/wp-content/uploads/TPM-Main-Part-3-Commands_v1.2_rev116_01032011.pdf

million PCs shipped with TPMs.⁸ Such devices were widely used in commercial applications including securing storage, authenticating users and software, and supporting secure network applications.

VII. CLAIM CONSTRUCTION

52. It is my understanding that in order to properly evaluate the '512 patent, the terms of the claims must first be interpreted. It is my understanding that for the purposes of this *inter partes* review, the claims are to be construed under the so-called *Phillips* standard, under which claim terms are given their ordinary and customary meaning as would have been understood by one of ordinary skill in the art in light of the specification and prosecution history, unless the inventor has set forth a special meaning for a term. For the purposes of this declaration, it is my opinion that the claim terms should be given their plain and ordinary meaning, other than the following:

A. “*restricted interface*”

53. A “*restricted interface*” is an interface that limits accessibility. In the '512 patent specification, disclosure of such an interface is sparse. For an example device that “includes a large and insecure storage 302 attached to a fast processor 304, and a smaller but secure storage 306 attached to a dedicated

⁸⁸ APPL-1024, <https://trustedcomputinggroup.org/wp-content/uploads/TPM-Applications-Whitepaper.pdf>

processor 308 and a sensor 310 (*e.g.*, a camera or a fingerprint reader),” the specification refers to access limitations in connection with an application programming interface (API). APPL-1001, 4:6-8 (“users cannot access the secure storage area, and the fast processor can only communicate with the dedicated processor/sensor via a restricted API”). Thus, reading the ’512 patent, a POSITA would understand its *restricted interface* to at least include an API that limits accessibility to a secure storage, a fingerprint sensor, or a dedicated processor.

54. I have been told that the Patent Trial and Appeal Board (“Board”) has addressed this term, *restricted interface*, in other proceedings. In particular, I have been informed in a third-party IPR challenging US 10,824,696 (“the ’696 patent”), in which the Board found “[t]he specification of the ’696 describes only an ‘API’ (Application Programming Interface) as an example of a restricted interface between two processors.” IPR2022-00244, (Institution) Paper 17, 31 (PTAB 2002). I understand in IPR2022-00244, the Board referenced (in its description of the ’696 patent) the very same description noted above (from the ’512 patent). See IPR2022-00244, Paper 17, 8. In IPR2022-00244, the Board was ultimately persuaded a trusted platform module (TPM) architecture sufficiently disclosed the restricted interface recited in the ’696 patent. IPR2022-00244 Paper 17, 40 (relying on a TPM API stack depicted in Figure 7 of U.S. Pub. App. 2008/0059804 naming Shah). I agree that certainly interface(s) from the TPM architecture

sufficiently disclose a *restricted interface*.

55. I understand that in parallel district court proceedings, Patent Owner proposed (and the court issued a preliminary construction) the term *restricted interface* is to be understood in accord with plain and ordinary meaning. APPL-1026, 1. I demonstrate in the analysis that follows that the prior art renders obvious a *restricted interface* including an API that limits accessibility to secure storage, a fingerprint sensor, or a dedicated processor, an interface that limits accessibility, and a restricted interface under its plain and ordinary meaning.

56. Further, a POSITA would have understood a plain and ordinary meaning of the term *restricted interface* is an interface with limited accessibility. One example of such an interface that limits accessibility as is an API as the API defines the rules and protocols for communication between two applications (via the interface) and thus provides a restricted interface. TPM is a widely recognized security architecture that provides a *restricted interface(s)* as the architecture employs limited-accessibility, physical and software interfaces (APIs) to provide a restricted interface to security-related operations such as key and credential management, secure storage, encryption, decryption, and digital signatures.

VIII. CLAIMS 1-6, 10-16 AND 20-21 OF THE '512 PATENT ARE UNPATENTABLE

57. I have been asked to provide my opinion as to whether Claims 1-6, 10-16 and 20-21 (“Challenged Claims”) of the '512 patent would have been

obvious in view of the prior art. The discussion below provides a detailed analysis of how the prior art references identified below teach the limitations of the Challenged Claims of the '512 patent.

58. As part of my analysis, I have considered the scope and content of the prior art and any differences between the alleged invention and the prior art. Below, I describe in detail the scope and content of the prior art, as well as any differences between the alleged invention and the prior art, on an element-by-element basis for each Challenged Claims of the '512 patent.

59. As described in detail below, the alleged invention of the Challenged Claims would have been obvious in view of the teachings of the identified prior art references as well as the knowledge of a POSITA. I apply the prior art as follows:

Ground	Claims	Basis
#1	1-6, 10-16 and 20-21	§103(a) over U.S. Patent No. 8,799,666 ("Kesanupalli") and U.S. Patent No. 8,099,789 ("Challener") in further view of U.S. Patent Publication No. 2004/0117636 ("Cheng")

60. I understand from Counsel that the '512 patent claims priority to Provisional Application No. 61/587,387 filed on Jan. 17, 2012 and Provisional Application No. 61/569,112 filed on Dec. 9, 2011. As stated above, I have used the earliest possible priority date – Dec. 9, 2011 – for my analysis of this declaration. I understand from Counsel that each of Kesanupalli, Cheng and

Challenger qualify as prior art with respect to this earliest possible priority date.

A. Ground 1: Claims 1-6, 10-16 and 20-21 are obvious under 35 U.S.C. § 103 over Kesanupalli, Challenger and Cheng.

1. Summary of Kesanupalli

61. U.S. Patent No. 8,799,666 to Kesanupalli *et al.* (“Kesanupalli”) issued from an application filed on March 24, 2010 and is entitled “Secure User Authentication Using Biometric Information.” (Kesanupalli) APPL-1006, (22), (54).

62. Kesanupalli is directed to a secure authentication system that uses biometric information to “provide[] a more secure authentication of the user than commonly used password-based methods and systems.” (Kesanupalli) APPL-1006, 2:59-62. For example, based on a match between a fingerprint template stored for a particular user and fingerprint information read from a finger presented at a fingerprint reader, Kesanupalli’s methods retrieve credentials associated with the user and communicate those credentials to an external system or process. (Kesanupalli) APPL-1006, FIGS. 8-9, 3:14-4:11. Authentication, Credentials, and Cryptography are discussed above at Section VI.A-D.

63. During an enrollment process with a resource such as a website, “a user swipes [or places] their finger [on] a fingerprint sensor several times to create [the] fingerprint template. The fingerprint template allows the user’s fingerprint to

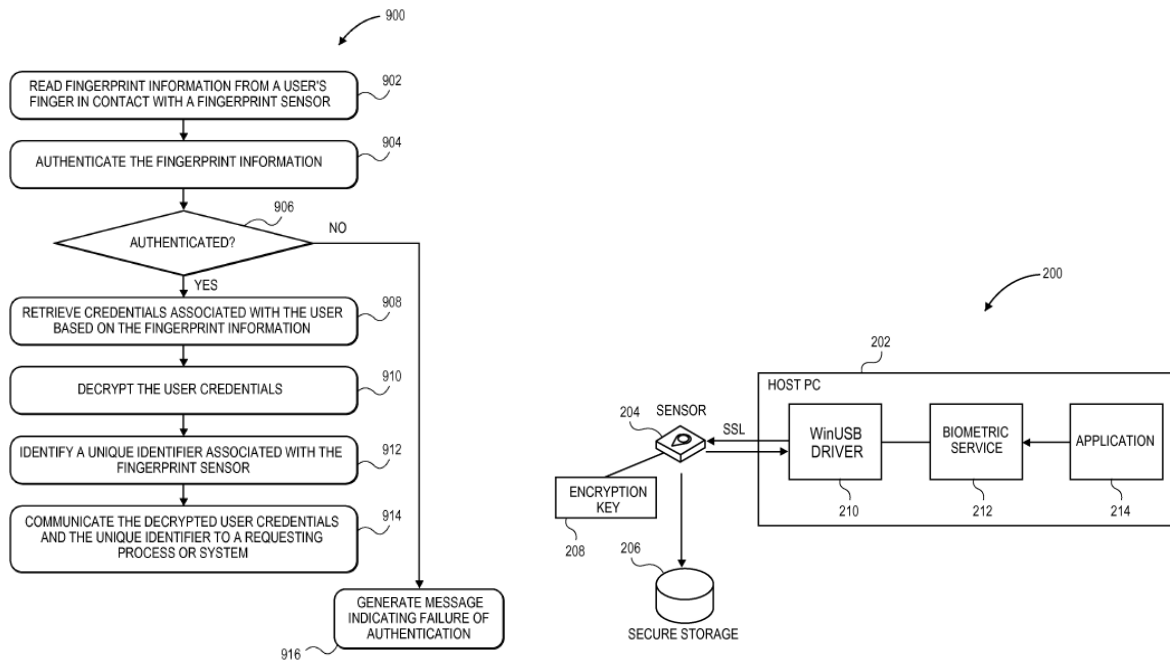
be distinguished from fingerprints associated with other users.” (Kesanupalli) APPL-1006, 3:33-41. The user (or system) also “provides user credentials, such as a password, cryptographic key ... or the like,” and the systems and methods described in Kesanupalli “bind the user’s fingerprint template with the user credentials.” (Kesanupalli) APPL-1006, 3:15-18, 3:41-46. To further protect user credentials, “[t]he fingerprint template and user credentials are then stored in a secure storage of the user’s device.” (Kesanupalli) APPL-1006, 3:46-47, *see also*, (Kesanupalli) APPL-1006, 8:29-53.

64. During a subsequent user identification or verification for the website, a user presents her biometric information (*e.g.*, places a finger on a fingerprint sensor) and if the presented fingerprint “matches a fingerprint template, the user’s credentials are released to the user and/or [the] service or process requesting the user verification.” (Kesanupalli) APPL-1006, 3:52-61. “[T]he user credentials released after finding a matching fingerprint template may include an OTP (One Time Password) token, RSA signature and the like.” (Kesanupalli) APPL-1006, 3:65-4:1, 3:41-44, FIG. 12.

65. FIG. 9 of Kesanupalli (shown below) illustrates these steps of authenticating a user in the context of an embodiment in which user credentials are cryptographically secured in storage. FIG. 2 (also shown below) illustrates an example system 200 in which user enrollment and authentication processes are

performed using a biometric sensor 204, secure storage 206 and a Host PC 202.

(Kesanupalli) APPL-1006, 6:26-49, 9:10-24.



(Kesanupalli) APPL-1006, FIGS. 9 and 2

66. In an authentication process, the biometric sensor 204 receives a biometric input from a user and releases the user's credentials from secure storage 206 upon a successful match of the input with the user's stored biometric template.

(Kesanupalli) APPL-1006, 6:31-45. The released credentials are then communicated to the external resource by an application 214 of the Host PC 202.

(Kesanupalli) APPL-1006, FIG. 2, 5:9-58, 6:11-31.

2. Summary of Challenger

67. Challenger is entitled "Apparatus and Method for Enabling Applications on a Security Processor." (Challenger) APPL-1007, (54). Challenger

“relates to securely protecting computer data, and more particularly to security processors provided in computer systems to implement security features.”

(Challener) APPL-1007, 1:7-9, 3:24-26. Challener illustrates a system 10⁹ (*e.g.*, computer or electronic device) with a system architecture suitable for securing data and functions of system 10 from unauthorized access and manipulation.

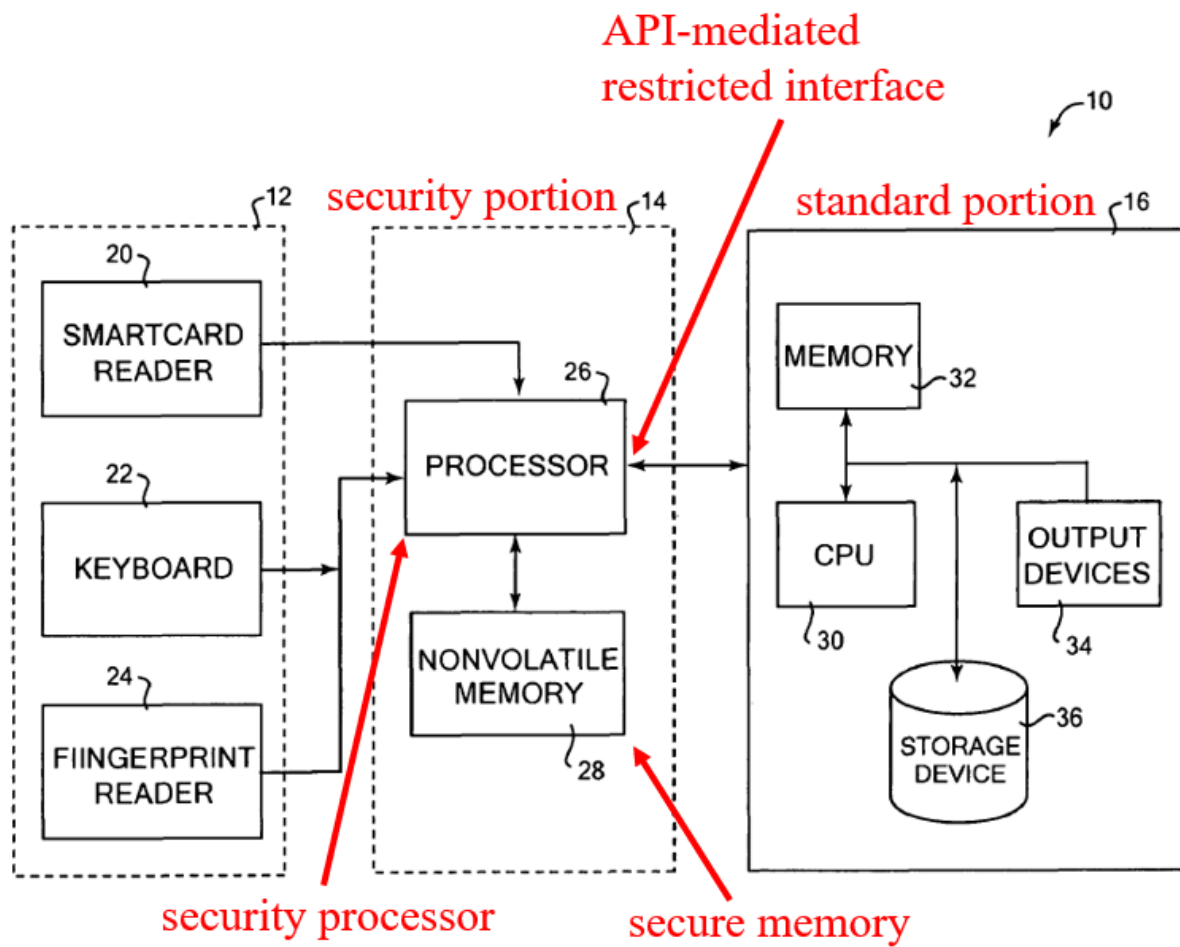
Challener) APPL-1007, 3:48-56. In particular, Challener describes system implementations in accord with a well-known Trusted Platform Module (TPM) specification promulgated by a standards organization, the Trusted Computing Group (TCG). *See* APPL-1007, 3:54-56. I discuss the TPM specification above at Section VI.E. As explained in Challener, in accord with the TPM specification, a standard chipset including a programmable microcontroller can be provided and used to store and secure information of the system that is desired to be protected.

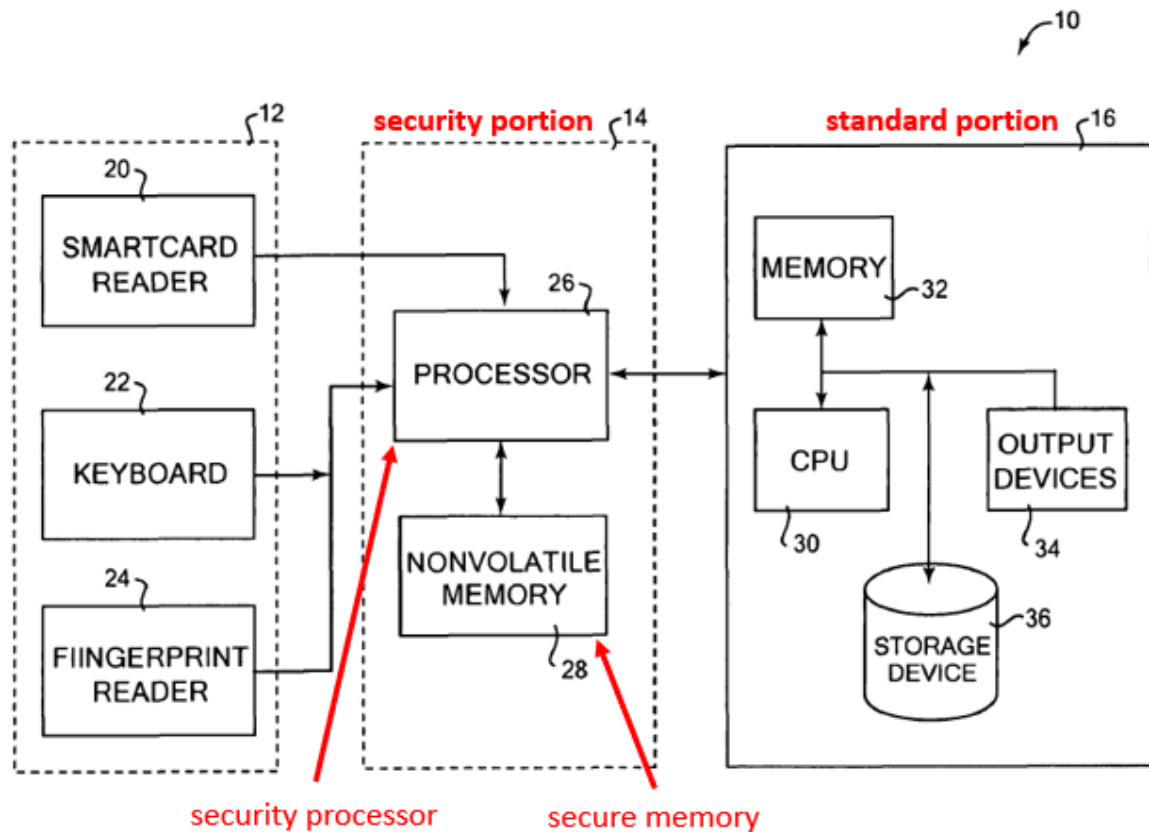
68. TPMs provide a platform for effective trusted computing features. (Challener) APPL-1007, 1:18-31. The security architecture (*i.e.*, the processor functions and the application programming interface (API)) that runs on the

⁹ “System 10 is a computer system having any of a variety of forms. For example, the computer system can be a mainframe computer, desktop computer, workstation, portable computer, or electronic device.” (Challener) APPL-1007, 3:37-51.

microcontroller) can be referred to as a TPM. A TPM can offer a variety of functions, including a random number generator, facilities for secure generation of cryptographic keys, and abilities to limit the use of keys, *e.g.*, to signing and verification, and/or encryption and decryption. (Challener) APPL-1007, 1:18-31. And as recognized by Challener, the TPM provides API-mediated interfaces. (Challener) APPL-1007, 1:25-27, 4:14-20. The TPM specification defines these API mediated interfaces as part of the TPM. One example of such the API-mediated interface in a TPM architecture is “The SEAL operation allows software to explicitly state the future “trusted” configuration that the platform must be in for the secret to be revealed. [...] If the UNSEAL operation succeeds, proof of the platform configuration that was in effect when the SEAL operation was performed is returned to the caller, as well as the secret data.”¹⁰ In other words, a component can ONLY access the TPM if it is in a “trusted” (i.e., restricted) configuration.

¹⁰ APPL-1017, https://trustedcomputinggroup.org/wp-content/uploads/TPM-Main-Part-3-Commands_v1.2_rev116_01032011.pdf





(Challener) APPL-1007, FIG. 1, annotated

69. Challener provides an implementation framework for secure systems, which may be TPM-based, and which includes an input portion 12, a security portion 14, and a standard portion 16. (Challener) APPL-1007, 3:46-57. Security portion 14 is used to secure data and provide applications for access and use only by authorized users through a dedicated security processor 26 and a memory 28. (Challener) APPL-1007, 4:9-65. The nonvolatile memory 28 is connected to the processor 26 and “is a secure memory which cannot be easily tampered with to read its data contents” and can be on the same chip or package as the memory 28. *Id.* The standard portion 16 is connected to the security processor 26 (vias its API-

mediated interfaces) and includes a microprocessor 30 that controls general operation of the system including input/output functions. (Challener) APPL-1007, 5:15-25.

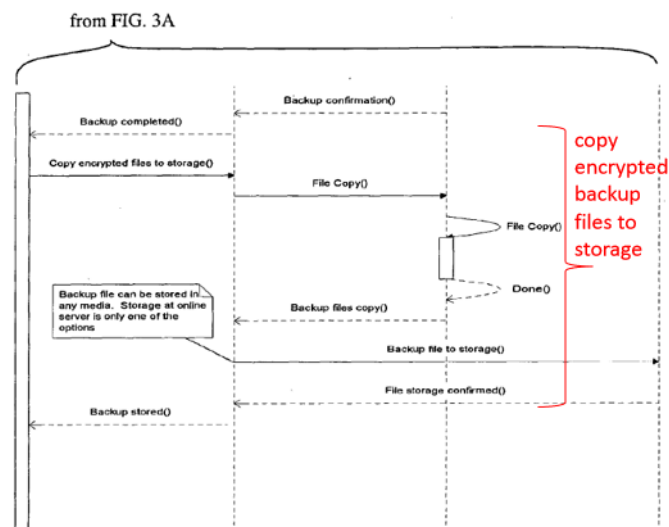
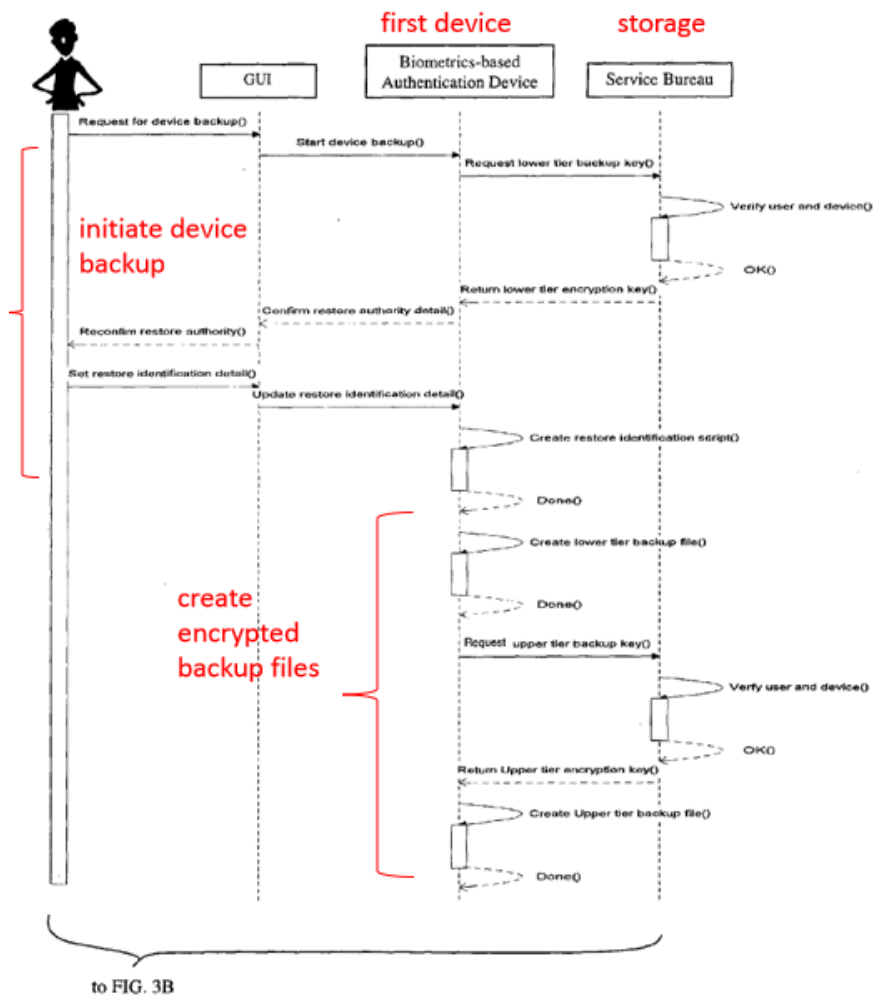
3. Summary of Cheng

70. Cheng is entitled “System, Method and Apparatus for Secure Two-Tier Backup and Retrieval of Authentication Information” and provides ways to “securely backup and restore” a user’s authentication-based device without compromising the security thereof. (Cheng) APPL-1008, (54), Abstract.

71. Cheng recognizes “[w]ith the rapid growth of computers, electronics, communications, networks, and the Internet, access control in general and network security in particular have become increasingly important for obvious reasons.” (Cheng) APPL-1008, [0003]. “Data, property interests, personal identity as well as personal safety could be at risk if security is breached” and to satisfy security needs, authentication systems, methods, and devices are needed to “certify the validity of an identity, account, object, and so on.” (Cheng) APPL-1008, [0003]. According to Cheng, “[b]iometrics-based authentication” is commonly implemented, which leads to portable devices that include “user’s authentication information, electronic identity and any data associated therewith.” (Cheng) APPL-1008, [0004], [0006].

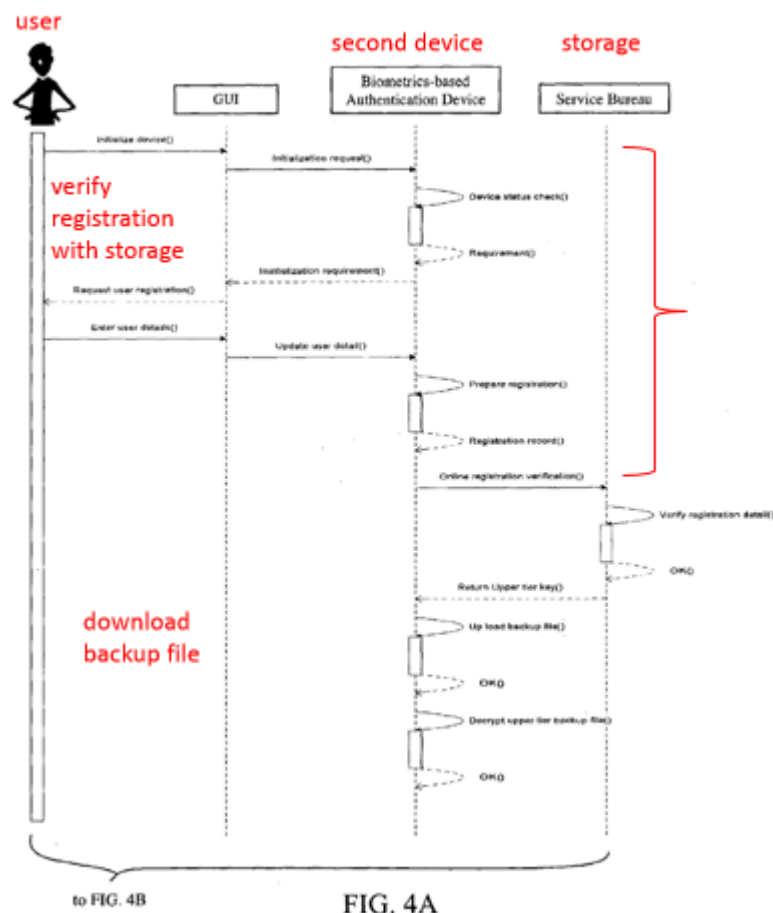
72. Cheng performs a secure backup of this data to avoid costs of creating

the data on new device from scratch should a first device containing the data be lost or stolen. *See* (Cheng) APPL-1008, Abstract, [0006]. An exemplary backup process is illustrated in Cheng's FIGS. 3A-3B where a backup of data of a first user's device is performed through secure communication between a user service bureau (storage) and the device. (Cheng) APPL-1008, [0029]-[0030]. Cheng's backup process includes a two-tier encryption of the backed-up data. (Cheng) APPL-1008, [0007]-[0008], [0031]-[0039].



(Cheng) APPL-1008, FIG. 3A – 3B, annotated

73. Cheng describes a use case of a user's first device, having been backed-up to a user service (a storage service), becoming unavailable. A secure connection is provided between the user service and a second device to retrieve the backed-up data downloaded to the second device. See (Cheng) APPL-1008, [0044]–[0050], FIGS. 4A-4C.

**(Cheng) APPL-1008, FIG. 4A, annotated**

4. Summary of the Kesanupalli and Challenger Combination

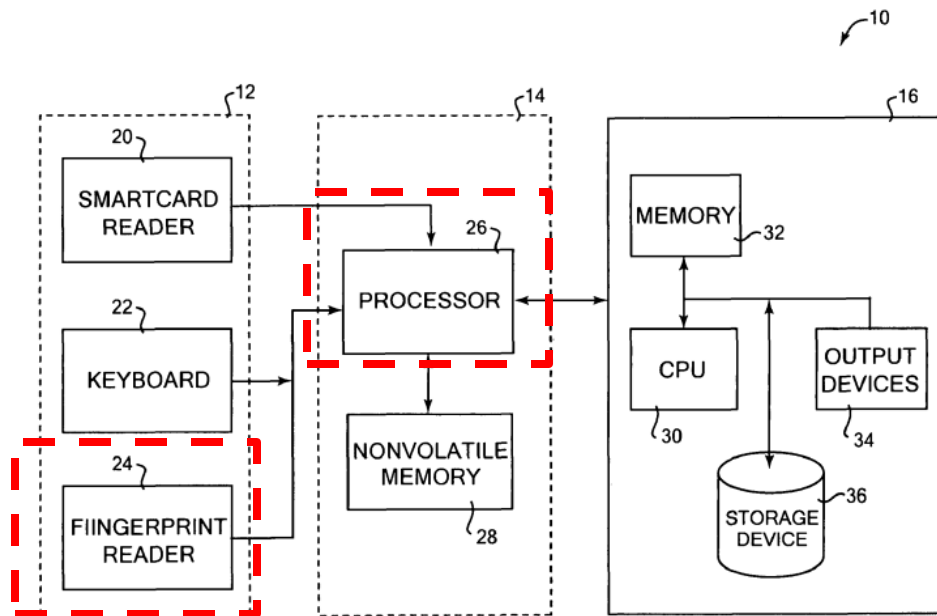
74. As demonstrated in my discussion of Kesanupalli above, Kesanupalli discloses the authentication translation techniques promoted by the '512 patent, including obtaining biometric input such as a fingerprint from a user, comparing this input to a stored biometric template to authenticate the user, and if those elements are matching, (and thus the user's identify verified) transmitting appropriate credentials of the user to the associated service on the user's behalf.

75. Kesanupalli also describes and illustrates example systems (including that of FIG. 2, which I discuss at length here) on which these techniques may be performed. For example, sensor 204 is capable of receiving and authenticating biometric information and communicating with its secure storage 206, which stores the user credentials and biometric templates, and on successful authentication, releases the user credential. (Kesanupalli) APPL-1006, 6:26-49. A computing host (PC) 202 receives information from the sensor 204 and communicates with external resources. (Kesanupalli) APPL-1006, 6:26-31, 5:59-6:25 (describing similar components relative to FIG. 1).

76. Kesanupalli explains that these components can be manufactured together in a single device, or separately. (Kesanupalli) APPL-1006, 2:48-50. A POSITA would have understood that Kesanupalli's techniques are achieved through processor(s) executing the described functions including those of sensor

204 (*e.g.*, matching, decrypting, etc.) and Host PC 202 (*e.g.*, executing applications 110, 112) as this is a basic function of a processor and implementation of an electronic device. *See* (Kesanupalli) APPL-1006, 6:28-49, 5:62-64 (describing similar components relative to FIG. 1). Kesanupalli suggests that specialized or general processors may perform its tasks. (Kesanupalli) APPL-1006, 6:11-17, 6:42-45, 9:58-10:12.

77. A well-known platform for implementing secure systems and for securely handling sensitive information is a Trusted Platform Module (TPM) security architecture as I have discussed above in Section VI.E. TPM implementations were widely implemented based on specifications from Trusted Computing Group (TCG) and would have been well understood by a POSITA before the earliest priority date of the '512 patent. Challenger, discussed above, is one architecture that uses a TPM implementation and is generally illustrative of a TPM architecture adapted to a biometric security application generally. More specifically, Challenger provides a specialized processor - security processor - that performs security functions, receives input from an authentication device and communicates with a main processor, which performs general tasks including connecting to external resources. The security processor is illustrated in FIG. 1 as processor 26:

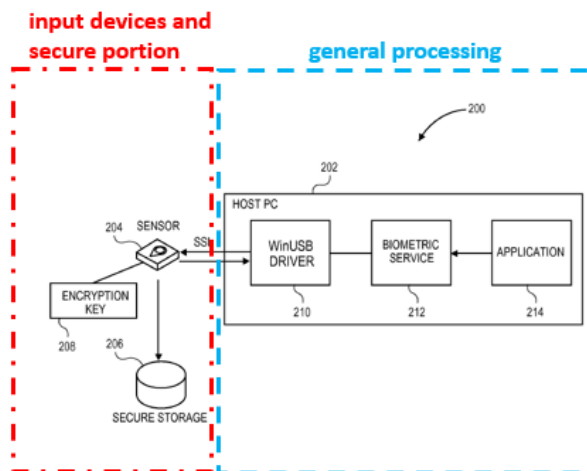


(Challener) APPL-1007, FIG. 1, annotated

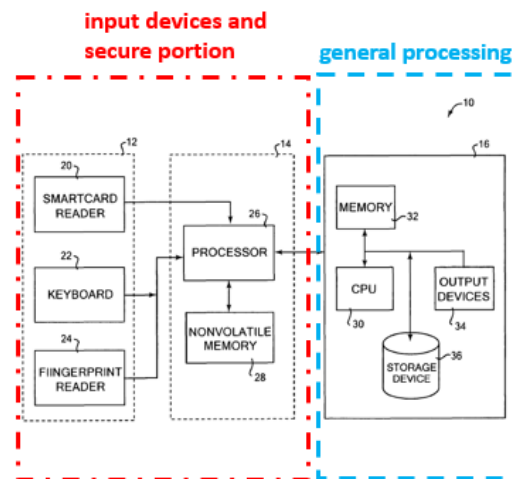
“Input portion 12 can include a variety of different input devices which allow a user to input data to the system 10 and authenticate the user’s identity to the system...For example, [] fingerprint reader 24 can be included in the system 10.” (Challener) APPL-1007, 3:58-62. *See also*, (Challener) APPL-1007, 4:24-65 on secure portion including using keys for “signing and verification”), 5:15-40 discussing the standard portion.

78. Challener’s TPM security architecture would have been an attractive platform on which to implement a biometric security system such as that described in Kesanupalli, particularly for a POSITA who would value a proven, widely adopted security architecture and who would expect that the marketplace would

favor a standards-based implementation (at least for the compatibility and reliability of the platform). As I discuss above in Section VI.E, TPM was a well-regarded and widely adopted architecture to achieve these goals. Additionally, Challenger's description of TPM architecture would have been attractive because of the correspondence of general and secure processor portions, secure storage, and a fingerprint reader described in Challenger's TPM based implementation with functional blocks described in Kesanupalli.



Kesanupalli, FIG. 2, Annotated



Challenger, FIG. 1, Annotated

79. Referring to annotated figures above, Kesanupalli and Challenger each involve a **first portion**—annotated blue—that executes applications, performs general computing tasks, and communicates with external resources and a **secure second portion**—annotated red—that executes functions for accepting biometric input, performing authentication, and securely storing the associated data. A

POSITA would have been motivated to implement Kesanupalli's functionality on Challenger's TPM architecture using a specialized security processor (*e.g.*, Challenger's security processor 26 including its APIs) to isolate the secure functions of Kesanupalli's sensor 204 and secure storage 206, while hosting on a general processor (Challenger's CPU or microprocessor 30) the applications and other functions of Kesanupalli's Host PC 202. The separation of the secure portion and the general processing portion is the separation of security-related data and computations from the general-purpose CPU and from other less secure elements of the host machine. Challenger's TPM-based decomposition into secure and general processor portions would have provided a well-known and standardized security architecture and a natural fit for performing the respective functions described by Kesanupalli.

5. Reasons to Combine Kesanupalli and Challenger

80. Continuing the discussion above, a POSITA would have been motivated to apply Challenger's teachings of a TPM-type security architecture to Kesanupalli's techniques to provide a security processor (a second processor as discussed below) in a secure portion communicating with a general processor (a first processor as discussed below) in a standard portion with API-mediated interfaces of the second processor in between. The API-mediated interfaces as those provided by implementation of the TPM-based architecture and provided

with the security processor. The combination of Kesanupalli's features and Challenger's TPM-based architecture would have produced the obvious, beneficial, and predictable results of a standardized and widely-deployed security architecture (TPM) for implementation of Kesanupalli's techniques as discussed below.

81. First, Challenger, Kesanupalli and the '512 patent are analogous art and are all in the same field of endeavor—namely, information security, and more particularly, security arrangements for protecting computers, components thereof, programs or data against unauthorized activity. *See, e.g.,* (Kesanupalli) APPL-1006, 2:30-38 (“[S]ystems and methods described herein relate to biometric authentication of users. ... biometric information provides an additional level of security when used in systems and procedures related to authentication of a user.”), 2:48-50 (“client device, such as a computer, cellular phone, and so forth”); (Challener) APPL-1007, 1:7-9 (“The present invention relates to securely protecting computer data, and more particularly to security processors provided in computer systems to implement security features.”), 3:52-56 (“Trusted Platform Module (TPM) ... for securing data and functions of the system [] from unauthorized access and manipulation.”); ('512 patent) APPL-1001, 1:35-44 (“Providing credentials to a service, whether via a mobile or other device, is often a tedious experience for a user. ... users will often engage in practices ... which render their credentials less secure against attacks. Accordingly, improvements in

authentication techniques would be desirable.”), FIG. 1 (showing client devices of notebook computer 102, smartphone 104, tablet 106).

82. I also recognize on the front pages of Challener, Kesanupalli and the '512 patent, the international classification (IPC), U.S. classifications (CPC) and/or the field of search classification consulted by the Office during examination for each patent include the same subclass, namely CPC/IPC G06F 21 which I am informed is directed to security arrangements for protecting computers, components thereof, programs or data against unauthorized activity. *See* ('512 patent) APPL-1001, (51), (52); (Kesanupalli) APPL-1006, (51) (58); (Challener) APPL-1007, (51).¹¹

83. Second, a POSITA would have been motivated to apply Challener's teachings of a standards-based TPM architecture as an implementation framework for Kesanupalli's biometric security techniques given the support a TPM implementation provides for many of the underlying cryptographic security services (encryption/decryption) and secure storage facilities provided. APPL-

¹¹ *see also* www.uspto.gov/web/patents/classification/cpc/html/cpc-

[G06F.html#G06F](http://www.uspto.gov/web/patents/classification/cpc/html/cpc-G06F.html#G06F) (subclass definitions);

www.uspto.gov/web/patents/classification/cpc/html/cpc-G06F.html#G06F

(CPC to IPC concordance).

1007, 1:18-31 (summarizing typical TPM functions and APIs). Challenger's description of its TPM-based security architecture includes *both* secure storage (nonvolatile memory 28) for secure data, *e.g.*, "secure hash values and ... keys," *and* fingerprint reader 24 behind a security processor 26 that "implements the particular security features and functions (e.g., API functions, algorithms)" in a system that includes secure and non-secure portions. (Challener) APPL-1007, 4:57-5:9, 3:67-4:23. These are precisely the security architecture features POSITA having Kesanupalli before them would be motivated to implement in a standards-based security architecture that would provide compatibility and robustness, and therefore engender trust amongst users and customers.¹² This is at least because a POSITA would recognize that authentication performed by Kesanupalli requires secure architecture to protect the sensitive, personal data.

84. Finally, a POSITA would have had reasonable expectations of success in implementing Kesanupalli's biometric authentication techniques in Challenger's TPM-based architecture techniques, as TPM techniques were well-known, standard and widely-deployed in the industry. *See* APPL-1007, 1:18-31, *see* Section VI.E, VIII.A.4. In contrast, Kesanupalli's architecture, which while it shares features with a TPM-based system is, as Challenger states, one of many

¹² *See, e.g.*, APPL-1016, <https://www.iso.org/standard/50970.html>

different techniques used to secure information and does not enjoy the confidence of the end-to-end systems implemented under the known TPM specification.

Further, Kesanupalli itself suggests the use of specialized processors.

(Kesanupalli) APPL-1006, 9:64-10:1. TPM is just such a specialized processor.

6. Reasons to Combine Kesanupalli, Challenger and Cheng

85. In addition to implementing Kesanupalli with Challenger's TPM architecture, as discussed above, a POSITA would have had been motivated to apply Cheng's teachings of backing up data of a device, such as the data of Kesanupalli's device including biometric information and credentials for user authentication.

86. Kesanupalli in view of Challenger has stored biometric information and user's credentials, which are a user's personal data stored in a user's first device. When combined with Cheng's backup teachings, a POSITA would recognize producing the obvious, beneficial, and predictable results of reducing the need to recreate the user's personal data when using a new device (*e.g.*, a second device replacing a lost or stolen first device). In other words, it was a known desire of a user to not require re-creation of the user's data onto a second device or replacement of a first device. This has been standard desire since electronic data on electronic devices was introduced.

87. Furthering the discussion above with respect to Kesanupalli,

Challener and the '512 patent's shared field of endeavor, Cheng is also in same field of endeavor such as security arrangements and procedures for protecting data in computer systems. *See, e.g.*, Section VIII.A.3, (Cheng) APPL-1008, [0003], [0004] ("computers, electronics, communications, networks, and the Internet, access control in general and network security in particular have come increasingly important...[d]ata, property interests, personal identity as well as personal safety could be at risk if security is breached. To satisfy different security needs, various authentication systems, methods, and devices exist today and new ones are continually being developed."). While Cheng lacks an explicit a search classification, a POSITA would recognize that it also is suitable for the same consideration and falls under the same field of endeavor under security arrangements for protecting computers, components thereof, programs or data against unauthorized activity. (Cheng) APPL-1008, [0007] ("Enabling technologies include biometrics, authentication, cryptography, and encryption/decryption.")

88. Second, a POSITA would have been motivated to employ Cheng's backup technique in connection with Kesanupalli's secure user authentication methods as implemented by Challener's architecture because Kesanupalli's stored biometric information and credentials are personal data. Biometric templates and user's credentials for external services such as websites are not only associated

with the user, but unique to them, and their personal data. The stored personal data (biometric information and credentials) in Kesanupalli as implemented on Challenger's architecture would need to be created and registered from scratch should the device be lost or stolen. (Cheng) APPL-1008, [0006]. As the devices of Kesanupalli/Challenger are personal electronic devices they are at risk of loss, theft, destruction, or damage. Thus, a desire to provide a copy of the stored data in outside of the user's device was well-recognized. Further, such a problem of inconvenience would have also been reasonably pertinent to the inventor of the challenged patent. See ('512 patent) APPL-1001, 8:18-21 (noting "[s]ince user data can be frequently backed up to the cloud storage, and recovered from this using the new device registration process, inconvenience to the user will be minimized" [when wiping a device]"). Cheng would have been reasonably pertinent to a problem faced by the inventor of the '512 patent and a POSITA in general as there is a known risks of theft and loss of devices and recovery of data. Therefore, a POSITA have been motivated to address this risk by combining Cheng with Kesanupalli/Challenger. Further there was a desire in the market of electronic devices to be capable of being backed-up long before the '512 patent. The desire to back up data included personal data, as well as secure data. And Cheng specifically teaches backing up personal data related to biometric information and secret keys. (Cheng) APPL-1008, [0007] – [0009]. This is data

just like the data stored in Kesanupalli's device. (Kesanupalli) APPL-1006, 3:32-51. Cheng provides a benefit of the desired backup of personal data to provide access in other user devices in a "reliable and secure way[]." (Cheng) APPL-1008, [0006].

89. Cheng also recognizes that it was a known practice in the industry to implement procedures that, on the loss or damage of a user's device, to implement a wiping procedure. Cheng recognizes this same problem:

In the event of loss, damage, or destruction of the device, the user's authentication information, electronic identity and any data associated therewith would be lost forever. Indeed, to prevent or at least to minimize the possibility of compromising the secrecy of the electronic identity and the authentication information, when a portable authentication device is reported loss or stolen, the general practice is to first deactivate or erase completely from the authentication system or secure network the electronic identity and authentication information associated with the lost/stolen authentication device and then create and register new ones from scratch.

(Cheng) APPL-1008, [0006]. These wiping procedures, such as discussed in U.S. Patent No. 9,652,629 to Brown et al. (APPL-1009) illustrate the need to provide backup systems to reduce the need to recreate the data. And Cheng continues to introduce a solution of backup of the data to a remote entity, from which the data

can be sent to a second device in the event of a wiping of a user's first device.

(Cheng) APPL-1008, [0007].

90. Continuing this discussion, to address costs of this risk of loss of device and/or data, Cheng teaches backing up of the data of the electronic device including biometric information and secret keys. (Cheng) APPL-1008, [0007] – [0009]. A POSITA would benefit from providing Cheng's teachings to the system/method/ computer program product of Kesanupalli in order to avoid the "cost and inconvenience" of recreating the data wiped from an electronic device when the data is desired to be provided on a new or replacement device. See (Cheng) APPL-1008, [0006] - [0007]. The user would continue to desire access to the websites with which she had enrolled, and thus, desire to have continued access to the credential (and biometric template) in order to obtain that access. Simply put, backing up the data avoids recreating the enrollment of FIG. 7 of Kesanupalli. Moreover, a POSITA would have recognized a desire in the market of electronic devices for secure data to be capable of being backed-up long before the '512 patent.¹³ As stated above, from the first introduction of electronic data itself, a POSITA recognized a user's desire to backup that data to avoid its loss. A

¹³ See, e.g., APPL-1020,

<https://www.popularmechanics.com/technology/gadgets/how-to/a6819/how-to-restore-your-data-after-a-hard-drive-crash/>

POSITA would appreciate a desire of the user to backup secure data such as personal data, which can be particular targets for wiping techniques due to their sensitive, personal nature.

91. Finally, there would have been a reasonable expectations of success in adding backup feature such as taught by Cheng in a biometric authentication system such as taught by Kesanupalli in view of Challener are all are similarly directed to handling of personal information including biometric authentication information in a secure manner. Adding a backup feature would merely have been applying a known technique to a known device (method, or product) that ready for improvement to yield predictable results. Indeed, the backup feature is implemented by routine software executed on generic processing components, which would have been a predictable application relative to Kesanupalli and Challener's systems and devices.

7. Claim 1

[1.0] A system, comprising:¹⁴

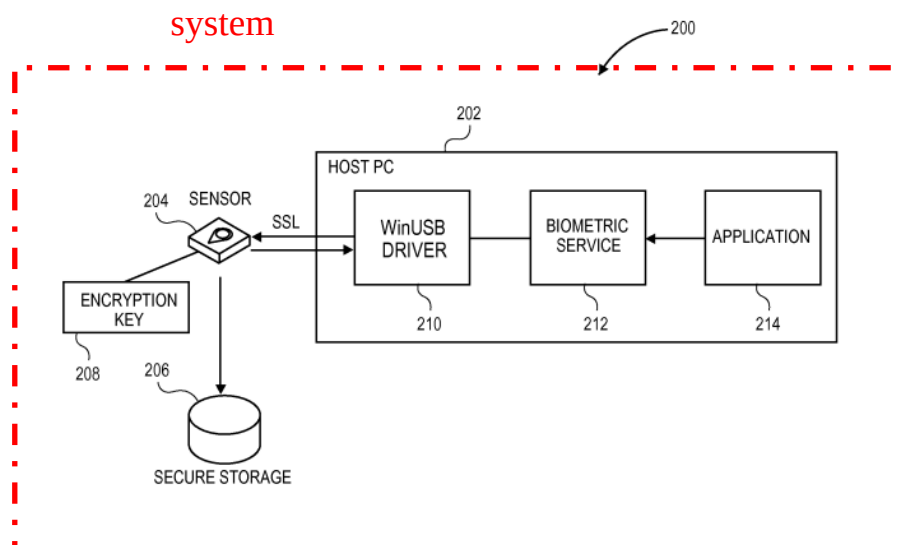
92. To the extent the preamble is limiting, Kesanupalli (in combination with Challener) renders obvious *a system* of claim 1. I have discussed this system

¹⁴ Again, all emphasis in quoted material has been added. Claim terms are generally *italicized*.

above in Sections VIII.A.4 - VIII.A.5 as well as in the elements that follow.

Kesanupalli is discussed in Section VIII.A.1 and Challenger is Section VIII.A.2 and applicable technical background in Section VI.

93. Kesanupalli describes in its FIG. 2 a system 200 “capable of performing biometric user enrollment and authentication.” (Kesanupalli) APPL-1006, 2:1-2. *See also*, (Kesanupalli) APPL-1006, Abstract, FIGS. 11-12.



(Kesanupalli) APPL-1006, FIG. 2, annotated

94. Kesanupalli’s system of FIG. 2 shares many similar features to that of FIGS. 1 and 6 for example and the discussion of each is generally applicable to the other systems with the specific features of each system noted by Kesanupalli. The system of FIG. 1 of Kesanupalli introduces many of the elements that are provided in the system of FIG. 2. (Kesanupalli) APPL-1006, 6:26-31. In Kesanupalli’s

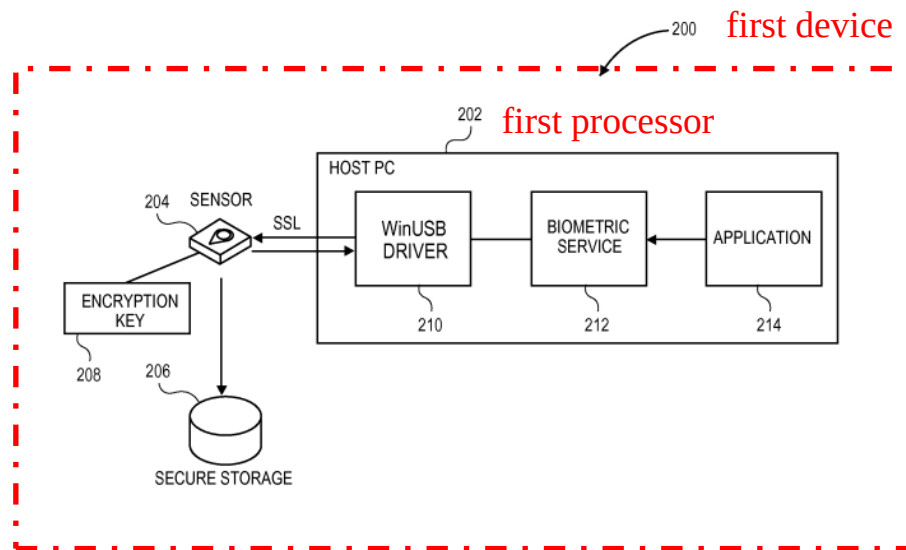
system 200, the sensor 204, including its secure storage 206, provides authentication translation and communicates with Host PC 202, which provides communication with external web servers including the communication of the user credential to a website provided by the webserver. See (Kesanupalli) APPL-1006, Abstract, 2:54-3:3, 4:3-11, 4:36-50, 6:11-49, FIG. 12. Challenger describes a known architecture that would have been applied to Kesanupalli including as I discussed above. Section VIII.A.4, VIII.A.5. See also, (Challenger) APPL-1007, FIG. 1, 3:45-4:65.

95. Thus, Kesanupalli and Challenger (in combination) render obvious *a system*, as claimed as detailed below.

[1.1A] a first processor of a first device, wherein the first processor is configured to:

96. Kesanupalli and Challenger together teach this element.

97. As to limitation [1.1A], Kesanupalli and Challenger disclose *a first processor of a first device*. Kesanupalli describes *a first device* exemplified in a first user device instantiation of FIG. 2 with the implementation of Challenger discussed above.



APPL-1006, FIG. 2, annotated

98. Biometric sensor 204, its secure storage 206, and Host PC 202 (including web browser application executing thereon) constitute *a device* that provides authentication translation and ultimately communicates user credentials to webservers. APPL-1006, Abstract, 2:54-3:3, 4:3-11, 4:36-50, 6:11-49.

99. FIG. 2 constitutes a *device*. This term is not further defined in claim 1, and the '512 patent uses this term generically ('512 patent) APPL-1001, 2:18-20. To the extent that Patent Owner imparts some more integrally formed meaning for the term, Kesanupalli also discloses integrally formed realizations of *a first device*. (Kesanupalli) APPL-1006 including at 2:48-50 ("In particular embodiments, the biometric sensor is physically attached (or manufactured into) a client device such as a computer, cellular phone, and so forth. which also includes the sensor 204") and 3:46-51 ("In one embodiment the secure storage device is contained within the

fingerprint sensor hardware. In other embodiments, the secure storage device is contained in a device that utilizes the fingerprint sensor.”)

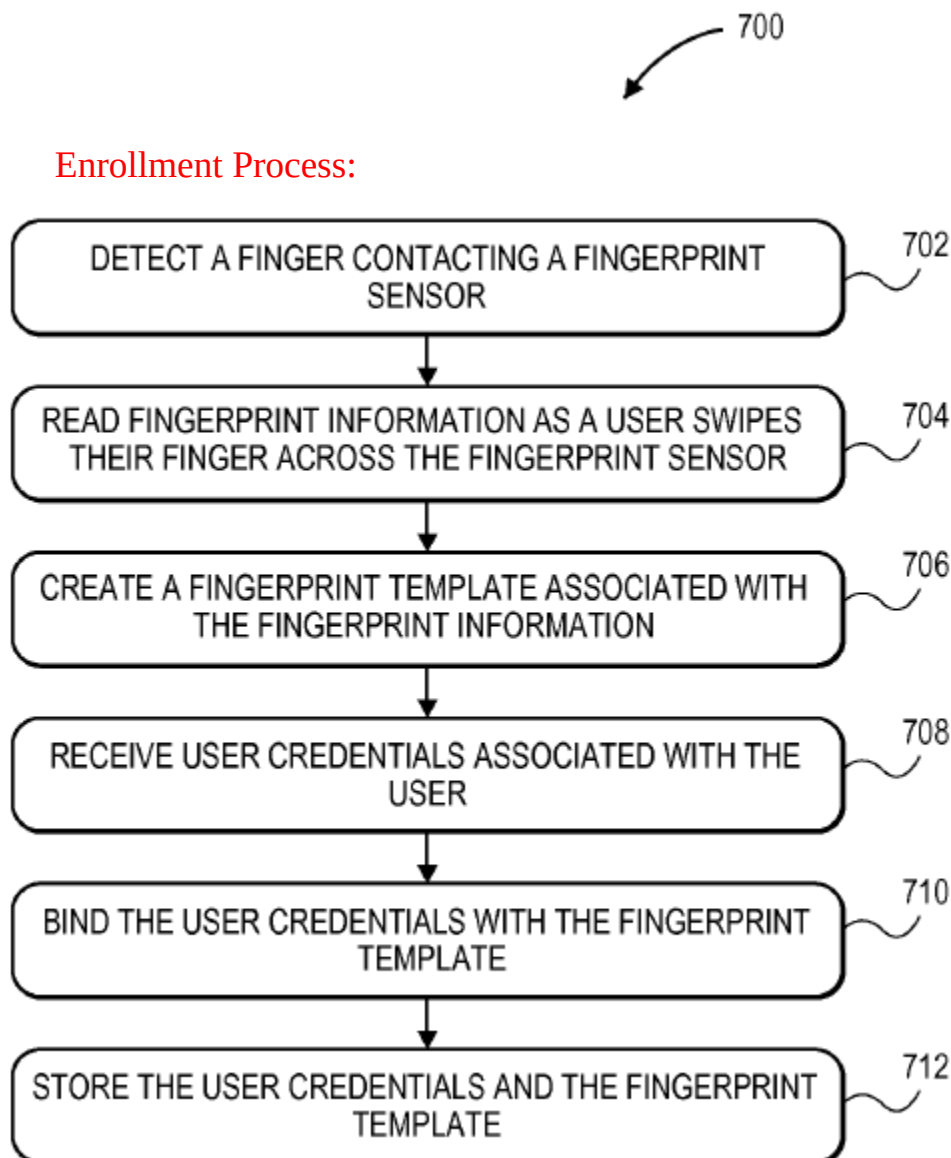
100. As discussed above, one would have had reason and found it obvious to implement Kesanupalli’s system 200 and the techniques performed by the system using a TPM-based architecture of Challenger. In doing so, the functions of the HOST PC 202, including communication with an external resource as discussed below, are implemented as *first processor*, which is a general processor shown in Challenger’s architecture as general processor 30. This processor is *configured* to perform the functions of Kesanupalli’s PC 202, which the same as the claim limitations with respect to the first processor as discussed below. (Kesanupalli) APPL-1006, 2:54-3:3, 6:11-25, 9:58-10:12 (discussing PC 202 functions executing on a processor); (Challener) APPL-1007, FIG. 1, 5:15-25 (discussing similar functions of general microprocessor 30)

101. Thus, Kesanupalli and Challenger (in combination) render obvious this portion of the element.

[1.1B] based at least in part on a request associated with a user to access an external resource, establish a secure connection with the external resource;

102. As to limitation [1.1B], Kesanupalli discloses *based at least in part on a request associated with a user to access an external resource* (e.g., user visiting a website), the first processor performs the action of *establish[ing] a secure connection with the external resource* (website).

103. As discussed above, Kesanupalli discloses an enrollment process, where the user visits an *external resource* (a website) to enroll with the resource and create, and store, the user's fingerprint template and user credentials associated with the website. See Section VIII.A.1, *see also*, APPL-1006, 5:33-43, and FIG. 7, which walks through the enrollment process.



(Kesanupalli) APPL-1006, FIG. 7, annotated

See also, Kesanupalli, FIG. 10, 11, 4:51-5:8, 8:29-53.

104. After enrollment, subsequent visits by the user to the external resource initiate an authentication translation procedure. In particular, Kesanupalli describes *a request* (user's request in accessing website to which she has enrolled) *to access an external resource* (the website). (Kesanupalli) APPL-1006, 4:36-5:14

...When a user accesses the Web site...[it] suggests that user enroll with their biometric sensor to provide a more secure user authentication...

After a user has enrolled with a particular Web site that supports biometric authentication, subsequent visits to the same Web site cause the Web browser plug-in to detect that the user has already enrolled with the Web site [and] the Web site prompts the user to perform user authentication (e.g., using the biometric device). ...If the fingerprint information matches a fingerprint template...the Web browser plug-in releases user secrets from the user credentials [and] [a]fter the credentials are released, they are communicated to the Web site...

See also, (Kesanupalli) APPL-1006, 2:54-3:3.

105. In particular, Kesanupalli's application such as the browser plug-in communicates with the external resource:

As used herein, a "web application", a "web-based application", and a "web-

enabled application” refers to a software application or software routine that is capable of communicating with one or more web servers or similar devices via the Internet or other data communication network. Additionally, a “plug-in” or a “browser plug-in” refers to an application or extension that provides a variety of different features and functions. Particular examples of “plug-ins” and “browser plug-ins” discussed herein provide features and functions related to user authentication while, for example, accessing web sites, making secure transactions, and the like.

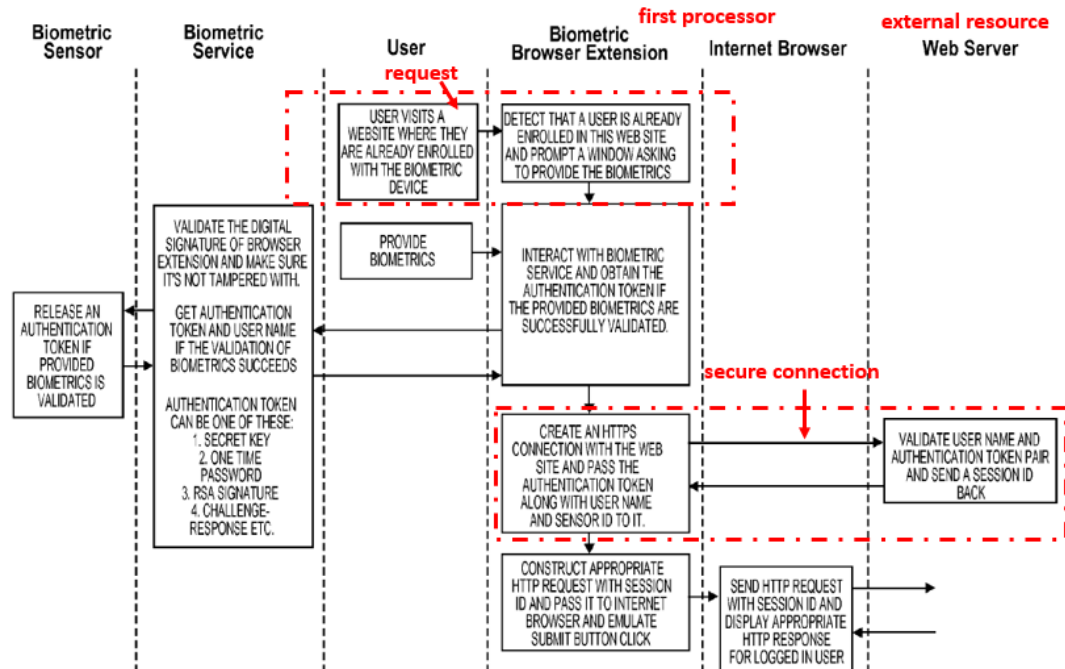
(Kesanupalli) APPL-1006, 2:54-3:3. *See also*, (Kesanupalli) APPL-1006, 7:25-32, 7:67-8:4, FIG. 6.

106. *Based at least in part on this user’s request to access the website with which she has enrolled, Kesanupalli establishes a secure connection with the external resource (website). Referring to the system of FIG. 2, this connection is provided between the application 214 of the PC 202 and the external resource, which is the webserver of the website. (Kesanupalli) APPL-1006, 2:54-3:3 (an application is “capable of communicating with one or more web servers or similar devices via the Internet or other data communication network”), 4:36-5:43, 6:11-50, 7:25-32, 7:67-8:4, 9:58 – 10:1. As discussed above, the application 214 is executing on the *first processor* of PC 202, and as such the *first processor* establishes a connection, via the Internet, with a web server of the website.*

(Kesanupalli) APPL-1006, 6:11-31, FIG. 9, block 914. Challenger confirms that a *first processor*, microprocessor 30 of a standard portion of a system, performs typical device functions including input/output operations, which would include Kesanupalli's connections to external resources. APPL-1007, 5:15-25.

107. Kesanupalli describes the connection between the PC 202 and the web server as a “secure communication link,” which is a *secure connection*. APPL-1006, 7:54-8:5, 12:62-67.

108. Kesanupalli's FIG. 12 is illustrative of communication between the processor and the webserver providing the *external resource* on the *secure connection*. FIG. 12 shows actions for authentication a user, which includes actions performed by a “user,” actions performed by the *first processor* (e.g., PC 202 providing biometric service, browser extension, browser applications) and the external resource's web server and shows communication between the PC 202 application and the web server.



(Kesanupalli) APPL-1006, FIG. 12, annotated

109. Thus, Kesanupalli and Challenger together disclose this element as *first processor* (Kesanupalli PC 202/Challenger microprocessor (CPU) 30) *based at least in part on a request associated with a user to access an external resource* (user's visit to a website with which she is/has enrolled), *establish a secure connection with the external resource* (secure communication link illustrated between PC 202 (application 214) and the webserver).

110. Further, as recognized by the background of Kesanupalli, a POSITA would have recognized websites for secure transactions, which are suggested in Kesanupalli, implement a *secure connection* such as a secure socket layer (SSL) or other security protocols in their connection with client devices. A POSITA would understand the plain and ordinary meaning secure connection is one that protects

the confidentiality and integrity of data. These guarantees are the results of the use of cryptography and network protocols, such as SSL. See Section VI.D above. See, e.g., (Kesanupalli) APPL-1006, 1:26-34, 2:54-64, 5:49-58.

111. Kesanupalli and Challener (in combination) render obvious this portion of the element.

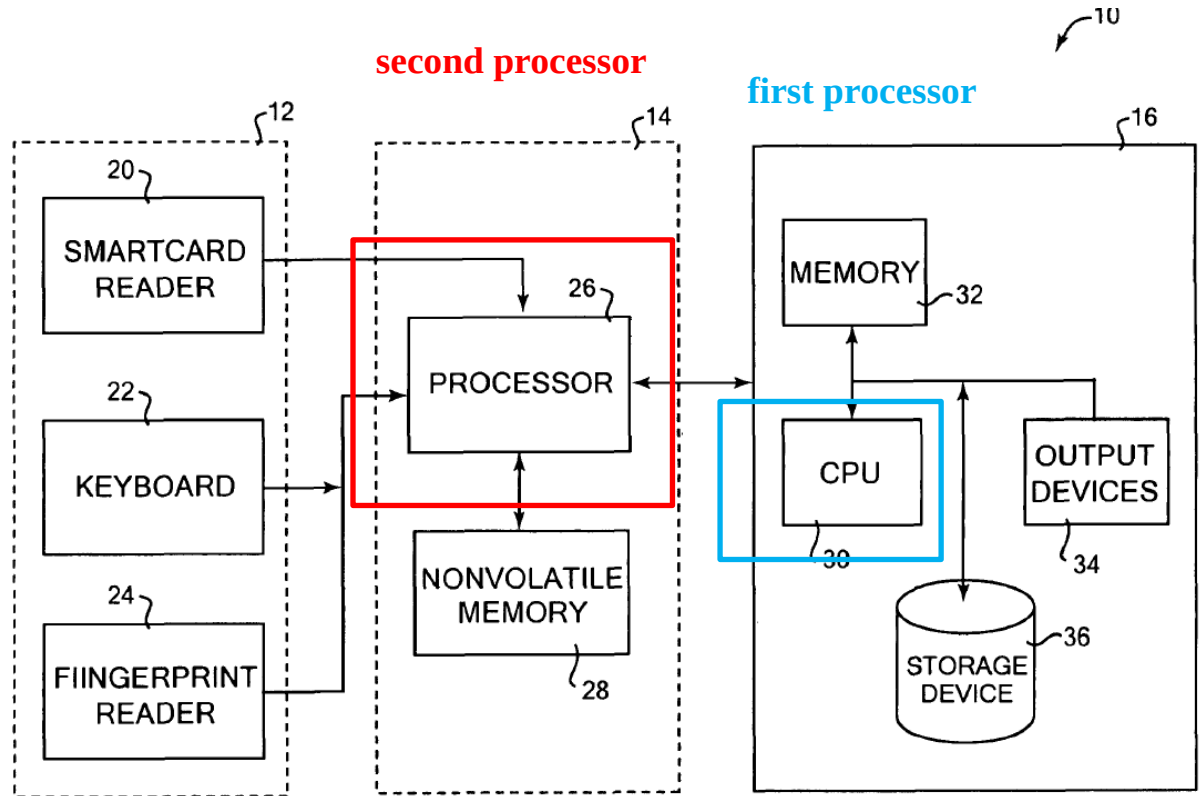
[1.1C] and [the first processor is configured to] communicate with a second processor using a restricted interface,

112. Kesanupalli and Challener together teach this element.

113. In Kesanupalli's system 200, Host PC 202 communicates with sensor 204. (Kesanupalli) APPL-1006, FIG. 2, 6:17-20. In the Kesanupalli-Challener combination introduced above (see Section VIII.A.4, VIII.A.5), when implementing the security functions of Kesanupalli's sensor 204 one would have been motivated to build the functions on Challener's TPM-based security architecture using a specialized security processor—a *second processor*—such as illustrated by Challener's security processor 26. As I discuss above the authentication portions of Kesanupalli's system, with its match-on-chip, secure storage, and sensor input map to the secure portion of the TPM-based framework of Challener. See Challener (APPL-1007), 4:9-24, FIG. 1. The *first processor* (Kesanupalli's HOST PC 202 functionality operating in standard portion 16 of Challener's TPM framework) *is configured to communicate with* this security processor, *a second processor*. (Kesanupalli) APPL-1006, 6:39-42 (“The

biometric sensor 204 creates a secure communication with Host PC 202 using SSL v3 protocol or other secure communication technique.”). *See also*, (Kesanupalli) APPL-1006, FIG. 2, 6:17-20, FIG. 12; (Challener) APPL-1007, FIG. 1, 5:15-17.

114. Challener also illustrates—and Kesanupalli’s functionality implemented using Challener’s TPM-based framework maintains—communication between a first processor (general processor 30) and second processor (security processor 26).



(Challener) APPL-1007, FIG. 1, annotated

(Challener) APPL-1007, 5:15-25 (“the standard portion 16 of system 10 is connected to the processor 26”). This communication between the first processor

and the second processor uses a *restricted interface*. First, an interface is a software and/or hardware structure that enables two components in a computer system to exchange information. Interfaces enable components to interact with each other while isolating internal algorithms and structures from external logic. When practiced in software, documented interfaces are generally referred to as APIs (application programmer interfaces). In computer systems, processors access coupled components such as storage using defined (and often standardized) interfaces. These interfaces act as protocols or “rules of the road” for data transfer between components. Such rules are necessary to ensure that different processors and storage systems can be used (or replaced) within the computer system without affecting other component logic with which they interact over the interface. Such abstraction and internal logic isolation is at the heart of modern computing systems and is a driving force behind the scalability and flexibility of their design.

115. As explained above, “*restricted interface*” is an interface that limits accessibility. Furthermore, based on the ’512 patent’s limited description, a POSITA would understand “*restricted interface*” to at least include an API that limits accessibility to a secure storage, a (fingerprint) sensor, or dedicated processor. See Section VII.A. Thus, when Kesanupalli is implemented using Challenger’s TPM framework for secure systems, communication between the security processor (26 as illustrated in Challenger) executing the functions of sensor

204 (and secure storage 206) and the standard portion of the device, the general processor (30 as illustrated in Challenger) executing the functions of PC 202 occurs through a *restricted interface*.

116. TPM-based implementations, including Challenger’s TPM-based implementation, are widely recognized examples of a security architecture that provides a *restricted interface* in the form of an access limiting API to restrict access to the TPM (secure processor, here *second processor*) by programs running on a general processor (here *first processor*) in a standard (non-secure) portion of a system. In particular, the access is limited to protect the security-related operations and data, such as key and credential management, secure storage, encryption, decryption, and digital signatures, which are provided or maintained in a secure portion of the system. This is the purpose of the TPM-based system as discussed above at Sections VI.E, VIII.A.2, VIII.A.4, VIII.A.5. *See also*, (Challenger) APPL-1007, 4:3-15 discussing controlling “access.”

117. Challenger discloses an interface (an API) that used in the communication of the *first processor* and the *second processor* in its TPM-based architecture. This is because the *second processor*, the secure processor (26) of Challenger, implements a security architecture that includes API functions used in its communication with other components – including the general processor 30, the *first processor*. APPL-1007, 4:9-24 (“The processor 26 typically implements a

‘security architecture’ which, as used herein, refers to the particular security features and functions (e.g. API functions, algorithms) implemented for the system 10 as determined by a security architecture application loaded on the processor 26.”), FIG. 1. Simply put, an API is software that defines how two components, here the software executing on processor 26 and the software executing on general processor 30 communicate with one another. And a TPM-based API, such as provided with Challenger’s processor 26, provides restrictions by the way of a set of rules or restrictions that are defined by the TPM standard for access to the security portion of the system.

118. Thus, implementing Kesanupalli using the architecture of Challenger provides communication between a *first processor* - general processor (PC 202) of a general portion of the TPM-based system - and a *second processor* – a security processor (executing sensor 204) of a security portion of the TPM-based system, where this communication is via a *restricted interface* (e.g., via an API of the TPM security architecture of the security processor executing the processor functions of sensor 204). This is merely a predictable use of a known element (the security processor performing as a TPM including its APIs) according to its established function in the TPM architecture to execute secure operations apart from the general portion of the system.

119. There is an API-type interface between processors in Kesanupalli

implemented on Challenger's architecture limits access to security functions and data on the secure processor—in short, this API provides, using TPM standardized techniques, a *restricted interface*. Such restrictions limit access and thereby protect the functions and data in the security portion of a TPM-based security architecture (e.g., sensor 204 and the secure storage 206) from being misused by an untrusted or malicious program. For example, programs may be loaded onto the Host PC 202, which are restricted from access to the secure portion of the system. The API interface provided by the TPM is a *restricted interface*.

120. Thus, Kesanupalli and Challenger (in combination) render obvious limitation [1.1C], “[*the first processor is configured to*] *communicate with a second processor using a restricted interface,*” and limitation [1.1] in its entirety.

[1.2] wherein the second processor is configured to:

receive a biometric input from a sensor, wherein the biometric input corresponds to at least one of a fingerprint, a feature usable for facial recognition, a voiceprint, a feature usable for a retina scan, or a typing feature; and

121. Kesanupalli and Challenger together teach this element.

122. As described above with reference to limitations [1.1A], [1.1C] and Section VIII.A.4, VIII.A.5, a POSITA would implement Kesanupalli using the TPM-based architecture and provide a security processor – *a second processor on the device* - executing the functions of Kesanupalli's sensor 204, which include receiving biometric input and performing authentication techniques as discussed

below. This is functionality taught by Kesanupalli as being performed by the sensor 204 and thus, would have been obvious to execute using the secure processor of Challenger.

123. First, Kesanupalli describes a *biometric input from a sensor*. In one example, Kesanupalli describes the sensor 204 receives a biometric input of a “fingerprint” from a “fingerprint sensor.”

FIG. 9 is a flow diagram depicting another embodiment of a procedure 900 for authenticating a user of a biometric authentication system. Initially, procedure 900 reads fingerprint information from a user's finger in contact with a fingerprint sensor (902). The procedure then authenticates the fingerprint information (block 904). If the fingerprint information is not authenticated, a message is generated indicating an authentication failure (block 906). If the fingerprint information is authenticated, the procedure retrieves credentials associated with the user based on the fingerprint information (block 908). The procedure then decrypts the user credentials (block 910) and identifies a unique identifier associated with the fingerprint sensor (block 912). The decrypted credentials and the unique identifier are communicated to a requesting process or system (block 914).

(Kesanupalli) APPL-1006, 9:10-24, FIG. 9. *See also*, Kesanupalli FIG. 8, 8:54-9:3 (“The authentication procedure reads fingerprint information from a user's finger in

contact with a fingerprint sensor (block 802).”), 6:31-36, (“In the example of FIG. 2, **the biometric sensor** decrypts the user credentials only after a successful biometric reading, such as a fingerprint swipe or fingerprint scan (using a placement style fingerprint sensor). For example, in a successful fingerprint swipe, the swiped fingerprint information matches a fingerprint template associated with the fingerprint sensor.”) Kesanupalli not only describes *biometric input corresponds to a fingerprint* but suggests other biometric information including those further recited in the recited group. APPL-1006, 6:31-36 including (“‘Biometrics’ and ‘biometric information’ refers to measurable biological characteristics of a user, such as a **fingerprint, facial characteristics, eye characteristics**, voice characteristics (also referred to as a ‘voiceprint’) and the like.”)

124. Thus, Kesanupalli and Challener (in combination) render obvious this element.

[1.3] [wherein the second processor is configured to:] access a record stored in a secure storage, wherein the record is associated at least with the external resource;

125. Kesanupalli and Challener together teach this element.

126. As described above with reference to limitations [1.1A], [1.1C] and Section VIII.A.4, VIII.A.5, a POSITA would understand Kesanupalli implemented using the TPM-based architecture of Challener has a security processor – *a second*

processor on the device - executing the functions of Kesanupalli's sensor 204, which include the limitation [1.3]. This is functionality taught by Kesanupalli as being performed by the sensor 204 and thus, would have been obvious to execute using the secure processor of Challenger.

127. With limitation [1.3], Kesanupalli describes a user's enrollment process in an external resource such as a website where the user's credential and biometric template are stored in a secure storage. (Kesanupalli) APPL-1006, 3:32-51:

Initially, a user enrolls with the biometric user authentication system by binding their user credentials with the user's biometric template (a 'fingerprint template' in specific implementations). ... During an example enrollment process that uses a fingerprint sensor as the biometric sensor, a user swipes their finger across the fingerprint sensor several times to create a fingerprint template.... After creating a fingerprint template, the user (or a web server or other system) provides user credentials, such as a password, cryptographic key, random seed, and the like. The systems and procedures described herein bind the user's fingerprint template with the user credentials. The fingerprint template and user credentials are then stored in a secure storage device.

See also, (Kesanupalli) APPL-1006, FIG. 2, FIG. 7, 8:29-53, 4:51-63, 3:65-4:1,

5:35-43, FIG. 12, 8:6-22. Kesanupalli in FIG. 7 walks through this enrollment process. (Kesanupalli) APPL-1006, 8:29-53 (including block 710 where “the procedure binds the user credentials with the fingerprint template[], then stores the user credentials and the fingerprint template [] in a secure storage.”) With respect to FIG. 2, Kesanupalli states “User credentials and other information may be **stored** within biometric sensor 204, in a secure storage 206, or any other storage mechanism.” (Kesanupalli) APPL-1006, 6:45-47.

128. The user’s credential bound with a biometric template is a *record*, which is stored on the first device in a secure storage as shown above. (Kesanupalli) APPL-1006, 4:36-5:43, 6:42-49 (“...User credentials and other information may be stored within biometric sensor 204, in a secure storage 206, or any other storage mechanism...”).

129. It is well known (and indeed taught in first year computer science courses), that computer systems and programs organize related data into records. Considered as the basis for nearly all software design, a record provides a basic data structure that contains different data types or fields grouped together under a single structure, *e.g.*, a block of memory, a compound data type in a program, part of a file in storage, or a row of a table in a database. This grouping allows for the representation of in a structured and organized manner and simplifies the program code and prevents bugs. A POSITA would understand Kesanupalli’s biometric

template and user credential to be associated data that is stored and referenced together (in the secure storage), and thus would be represented as a record. A “record” can be defined as “a data structure that is a collection of fields (or elements).” APPL-1019 (Computer Dictionary). The template and user credential provide a collection of elements together.

130. The record is *associated at least with the external resource* as the credential and/or the template are associated with the website.

In particular implementations, the user performs the enrollment process for each Web site the user accesses that supports biometric authentication. Additionally, different user credentials are associated with each Web site with which the user enrolls. Thus, if the user enrolls with five different Web sites that support biometric authentication, the biometric sensor in the user's computing device stores five separate sets of user credentials, each of which is associated with one of the five different Web sites. Additionally, if different users access the same Web site, separate user credentials and separate biometric templates are maintained for each user.

(Kesanupalli) APPL-1006, 5:36-43. *See also*, (Kesanupalli) APPL-1006, 8:47-53 (“In a specific embodiment, the procedure also binds a particular web site (e.g., a web site requesting biometric enrollment and/or biometric authentication of a user)

with the fingerprint template. Thus, a particular user may perform the biometric enrollment procedure for each web site that the user is to provide future biometric authorization or biometric authentication.”)

131. Kesanupalli’s secure storage 206 provides the recited *secure storage*. See APPL-1006, FIG. 2, 6:45-49. A *secure storage* is one that is protected from unauthorized access. By describing its storage as “secure storage” and through the illustration of FIG. 2, a POSITA would understand Kesanupalli to be providing a secure storage area that is protected from unauthorized access. As illustrated by Kesanupalli’s FIG. 2, access to the secure storage 206 is controlled by the sensor 204 and the storage is not directly accessible by others (*e.g.*, application 214 executing on the Host PC 202); for example, the user credential in the secure storage 206 cannot be accessed (and released) except by the sensor 204 (*e.g.*, when verifying a match of biometric input). Challenger confirms that the storage in a secure portion of a TPM-based architecture is accessible by only its secure processor and thus is a *secure storage*. (Challener) APPL-1007, FIG. 1, nonvolatile memory 28, 4:57-65 (“A nonvolatile memory 28 is included in security portion 14 and is connected to the processor 26. For example, memory 28 can be Flash memory, Electrically Erasable Programmable Read Only Memory (EEPROM), or other type of memory. In the described embodiment, memory 28 is a secure memory which cannot be easily tampered with to read its data contents. In

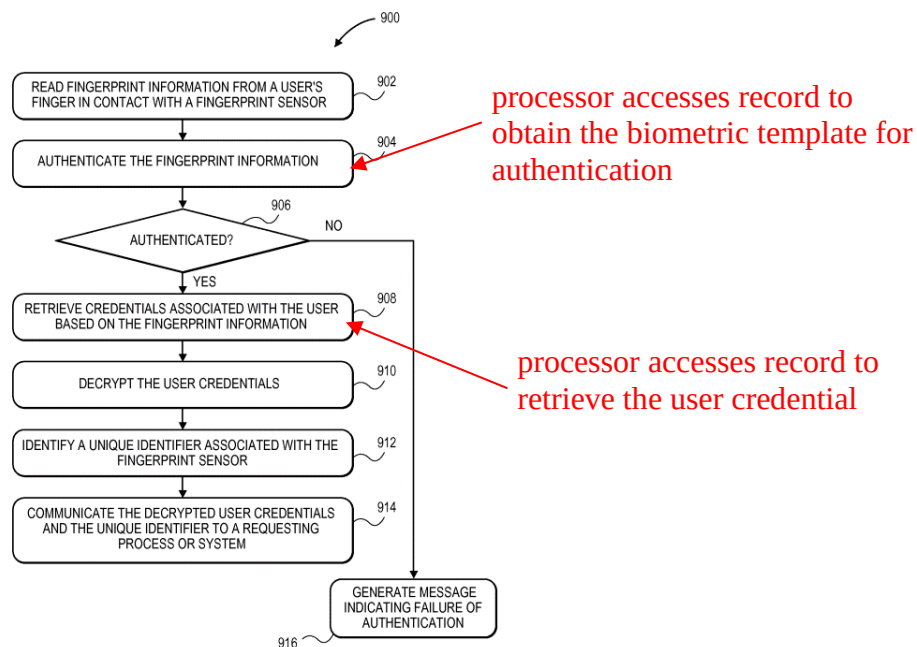
some embodiments, the nonvolatile memory 28 can be included on the same integrated circuit chip, or package, of the processor 26.”)

132. Kesanupalli’s sensor 204 *access[es] a record* (e.g., user credential, template) *stored in a secure storage* – Kesanupalli’s “secure storage 206”. This is because the sensor 204 has a match-on-chip functionality that accesses the template to perform the matching and again accesses the record to retrieve the user credential when a match is determined to release the credential. (Kesanupalli) APPL-1006, 6:31-49. *See also*, Kesanupalli 3:52-4:12, 5:17-32, 8:40-53, 9:4-9, 9:17-24, FIG. 12. Thus, in the authentication translation, the second processor, performing the functions of sensor 204, accesses at least one record. In particular, the security (second) processor accesses the biometric template of the record to perform the matching and accesses the user credential of the record to release the credential to the external resource. FIG. 9 of Kesanupalli is illustrative:

FIG. 9 is a flow diagram depicting another embodiment of a procedure 900 for authenticating a user of a biometric authentication system. Initially, procedure 900 reads fingerprint information from a user's finger in contact with a fingerprint sensor (902). The procedure then authenticates the fingerprint information (block 904). If the fingerprint information is not authenticated, a message is generated indicating an authentication failure (block 906). If the fingerprint information is

authenticated, the procedure retrieves credentials associated with the user based on the fingerprint information (block 908). The procedure then decrypts the user credentials (block 910) and identifies a unique identifier associated with the fingerprint sensor (block 912). The decrypted credentials and the unique identifier are communicated to a requesting process or system (block 914).

(Kesanupalli) APPL-1006, 9:10-24.



(Kesanupalli) APPL-1006, FIG. 9, annotated

133. Referring to the device of FIG. 2 as implemented by Challener, the *second processor* on the device and in particular the processor circuitry executing the functions of sensor 204 (secure processor 26) accesses the biometric template of the record stored in the secure storage 206 in order to authenticate the user

through matching a biometric input with the biometric template of the stored record:

FIG. 2 shows another example system 200 capable of performing biometric user enrollment and authentication via a biometric sensor 204. System 200 includes a host PC 202, a WinUSB driver 210, a biometric service 212 and an application 214 similar to the components discussed above with respect to FIG. 1. In the example of FIG. 2, the biometric sensor decrypts the user credentials only after a successful biometric reading, such as a fingerprint swipe or fingerprint scan (using a placement style fingerprint sensor). For example, in a successful fingerprint swipe, the swiped fingerprint information matches a fingerprint template associated with the fingerprint sensor. In a particular embodiment, the validity enterprise sensor has a unique 256 bit encryption key 208 and a unique identifier (e.g., serial number). The biometric sensor 204 creates a secure communication with Host PC 202 using SSL v3 protocol or other secure communication technique. In a particular implementation, biometric sensor 204 includes a “match on chip” functionality that releases a user's credentials only upon a successful fingerprint swipe or other biometric reading. User

credentials and other information may be stored within biometric sensor 204, in a secure storage 206, or any other storage mechanism.

(Kesanupalli) APPL-1006, 6:31-49.

134. Thus, Kesanupalli and Challener (in combination) render obvious this element.

[1.4] [wherein the second processor is configured to:] retrieve, from the record, at least one of a password, a cookie, or a cryptographic key;

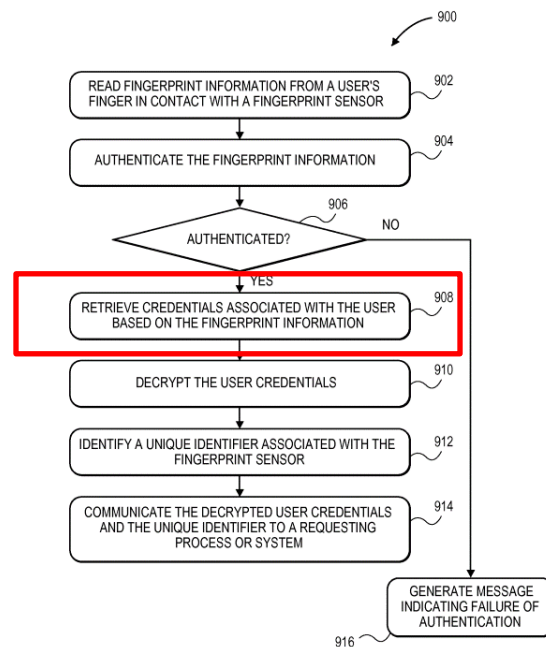
135. Kesanupalli and Challener together teach this element.

136. As described above with reference to limitations [1.1A], [1.1C] and Sections VIII.A.4, VIII.A.5, a POSITA would understand Kesanupalli implemented using the TPM-based architecture having a security processor – *a second processor on the device* - executing the functions of Kesanupalli’s sensor 204, includes executing the functions of limitation [1.4]. This at least because Kesanupalli’s sensor 204 performs the retrieving of data from its secure storage 206 as discussed below.

137. As discussed above in limitation [1.3], in an enrollment process the *record* (template and credential) is stored in the secure storage 206 of the first device. And Kesanupalli also describes one type of user credential is a *cryptographic key* and another is a *password*. (Kesanupalli) APPL-1006, 3:41-44 (“user credentials, such as a password, cryptographic key, random seed, and the like.”) See also, (Kesanupalli) APPL-1006, 3:65-4:1, 5:35-43, FIG. 12 referring to

“authentication tokens”, 8:6-22.

138. After enrollment including the creation and storage of the record, subsequent visits by the user to the external resource initiate an authentication translation procedure. APPL-1006, 5:9-14 (“After a user has enrolled with a particular Web site that supports biometric authentication, subsequent visits cause...the Web site prompts the user to perform user authentication.”) If the user is authenticated, the procedure continues to provide the *second processor*, security processor executing functions of sensor 204, *retriev[es] from the record* the user credential (*e.g.*, password or key) to release the credential to the requesting resource (website via PC 202). FIG. 9 is illustrative of the procedure for authenticating a user after she has enrolled in a website and where, if authenticated, “the procedure **retrieves credentials** associated with the user.” APPL-1006, 9:17-20.

**APPL-1006, FIG. 9, annotated**

See also (Kesanupalli) APPL-1006, 3:52-59 (“During an example user identification process (also referred to as a user verification process), a user swipes their finger across a fingerprint sensor. The process then determines whether the user's fingerprint information matches a fingerprint template associated with the fingerprint sensor. If the user's fingerprint information matches a fingerprint template, the user's credentials are released to the user and/or a service or process requesting the user verification.”), 3:62-4:1 (“[T]he user credentials released after finding a matching fingerprint template may include an OTP (One Time Password) token, RSA signature and the like”), 6:42-45 (“In a particular implementation, biometric sensor 204 includes a “match on chip” functionality that releases a user's credentials only upon a successful fingerprint swipe or other biometric reading.”),

FIG. 12 including “Release an authentication token [user credential] if provided biometrics is validated.” Simply put the retrieving of the credentials from security upon verification of the user is the step that effectuates the goal of Kesanupalli. See (Kesanupalli) APPL-1006, 1:59-62.

139. Thus, Kesanupalli and Challener (in combination) render obvious this element.

[1.5] [wherein the second processor is configured to:] perform a cryptographic operation;

140. Kesanupalli and Challener together teach this element.

141. As described above with reference to limitations [1.1A], [1.1C] and Sections VIII.A.4, VIII.A.5, a POSITA would provide Kesanupalli implemented using the TPM-based architecture having a security processor – *a second processor on the device* –executing the functions of Kesanupalli’s sensor 204, which include limitation [1.5]. A security processor executing the functions of the sensor 204 would perform the decryption performed Kesanupalli’s sensor.

142. Kesanupalli, as discussed in limitation [1.4] above, describes on a successful authentication of a user’s biometric input, retrieving the user’s credentials from the record in storage, after which it performs a cryptographic operation. FIG. 9 of Kesanupalli’s procedure continues to explain “the procedure retrieves credentials associated with the user based on the fingerprint information (block 908). The procedure then decrypts the user credentials (block 910).” APPL-

1006, 9:17-21.

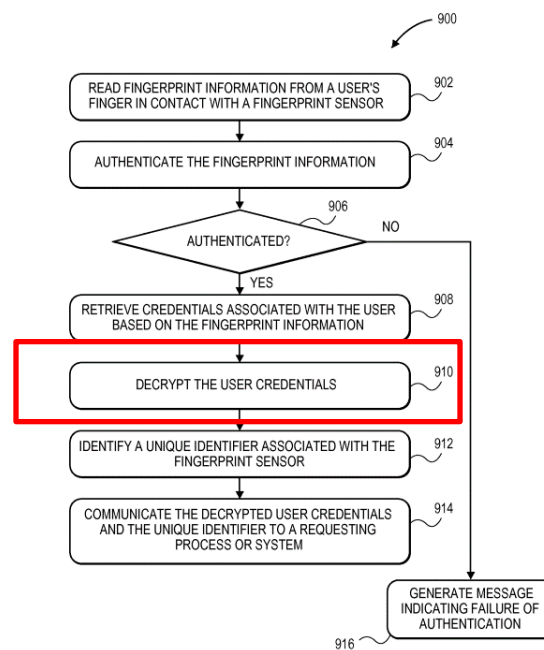
**APPL-1006, FIG. 9, annotated**

FIG. 2 shows another example system 200 capable of performing biometric user enrollment and authentication via a biometric sensor 204. System 200 includes a host PC 202, a WinUSB driver 210, a biometric service 212 and an application 214 similar to the components discussed above with respect to FIG. 1. In the example of FIG. 2, the biometric sensor decrypts the user credentials only after a successful biometric reading, such as a fingerprint swipe or fingerprint scan (using a placement style fingerprint sensor). For example, in a successful fingerprint swipe, the swiped fingerprint information matches a fingerprint template associated with the fingerprint sensor. In a

particular embodiment, the validity enterprise sensor has a unique 256 bit encryption key 208 and a unique identifier (e.g., serial number). The biometric sensor 204 creates a secure communication with Host PC 202 using SSL v3 protocol or other secure communication technique. In a particular implementation, biometric sensor 204 includes a “match on chip’ functionality that releases a user’s credentials only upon a successful fingerprint swipe or other biometric reading. User credentials and other information may be stored within biometric sensor 204, in a secure storage 206, or any other storage mechanism.

(Kesanupalli) APPL-1006, 6:31-49. A decryption is a cryptographic operation.

See also Section VI.C above.

143. As discussed, a POSITA would have implemented the sensor 204 functions using a *second processor* according to the TPM-based architecture of Challenger. Challenger also confirms a security processor *is configured to* perform cryptographic operations. (Challener) APPL-1007, 4:24-43 (“For example, processor 26 can implement a TPM security architecture, which is a well-known standard for securing computer data. ... The typical security functions of a TPM include random number generation, secure generation of cryptographic keys, secure storage, abilities to limit the use of keys (such as for signing and verification), and/or encryption and decryption.”)

144. Thus, Kesanupalli and Challener (in combination) render obvious this element.

[1.6] [wherein the second processor is configured to:]

[1.6A] in response to determining that there is a match between the biometric input and a stored biometric template accessed by the second processor,

[1.6B] facilitate a login of the user to the external resource at least in part by transmitting, via the established connection, output based at least in part on the at least one of the password, the cryptographic key, or the cookie retrieved from the record, and wherein the user is logged in to the external resource based at least in part on the output; and

145. Kesanupalli and Challener together teach this limitation [1.6], including limitations [1.6A] and [1.6B].

146. As described above with reference to limitations [1.1A], [1.1C] and Sections VIII.A.4, VIII.A.5, a POSITA would implement Kesanupalli using the TPM-based architecture of Challener providing a security processor – *a second processor on the device* - executing the functions of Kesanupalli's sensor 204, which include the limitation [1.6]. That is the sensor 204 determining that there is a match between the biometric input and a stored biometric template and facilitates a login.

147. Generally, determining a match between a biometric input and a biometric template is a general technique of authentication implemented well before the '512 patent. See Section VI.A. As to limitation [1.6A] in particular,

Kesanupalli's authentication procedure includes determining that there is *a match between the biometric input (e.g., fingerprint information) and a stored biometric template (e.g., fingerprint template)*. (Kesanupalli) APPL-1006, 9:14-20:

FIG. 9 is a flow diagram depicting another embodiment of a procedure 900 for authenticating a user of a biometric authentication system. Initially, procedure 900 reads fingerprint information from a user's finger in contact with a fingerprint sensor (902). The procedure then authenticates the fingerprint information (block 904). If the fingerprint information is not authenticated, a message is generated indicating an authentication failure (block 906). If the fingerprint information is authenticated, the procedure retrieves credentials associated with the user based on the fingerprint information (block 908). The procedure then decrypts the user credentials (block 910) and identifies a unique identifier associated with the fingerprint sensor (block 912). The decrypted credentials and the unique identifier are communicated to a requesting process or system (block 914).

(Kesanupalli) APPL-1006, 6:31-49 ("For example, in a successful fingerprint swipe, the swiped fingerprint information matches a fingerprint template associated with the fingerprint sensor. ... In a particular implementation, biometric sensor 204 includes a 'match on chip' functionality that releases a user's

credentials only upon a successful fingerprint swipe or other biometric reading. User credentials and other information may be stored within biometric sensor 204, in a secure storage 206, or any other storage mechanism.”), 3:52-56 (“During an example user identification process (also referred to as a user verification process), a user swipes their finger across a fingerprint sensor. The process then determines whether the user's fingerprint information matches a fingerprint template associated with the fingerprint sensor.” See also, (Kesanupalli) APPL-1006, 5:17-32, 8:54-9:25, FIG. 9 including “authenticated?”, FIG. 12 including “biometrics is validated”. Validating or authenticating means matching the template and the biometric input. This is standard authentication practice as discussed in Section VI.A.

148. *In response* to determining this match, Kesanupalli transmits the credentials to the requesting external resource. (Kesanupalli) APPL-1006, 9:10-24 (“If the fingerprint information is authenticated, the procedure retrieves credentials associated with the user based on the fingerprint information (block 908). ... The decrypted credentials ... are communicated to a requesting process or system (block 914).”) See also, (Kesanupalli) APPL-1006, FIG. 12, 9:4-9. In the device of FIG. 2, in response to the “on-chip matching” performed by the sensor 204, the user credential is retrieved from the secure storage 206 of the sensor 204 and provided to application 214 of PC 202 for communication to the external resource

(e.g., website). (Kesanupalli) APPL-1006, 6:11-49.

149. In particular, as to limitation [1.6B], in Kesanupalli's procedure, the security processor performing the functions of the sensor 204 (i.e., the *second processor*) *facilitate[s] a login of the user to the external resource*. The *second processor* facilitates the login as, in response to verifying the match as in limitation [1.6A], the sensor 204 retrieves the credentials from the secure storage (206) (see limitation [1.4]) and transmits the credentials (*e.g.*, password, cryptographic key) to the external resource, via the HOST PC 202. First, the '512 patent specification does not define or even use the term *facilitates*. A POSITA would understand *facilitates*, in the context of the claim and in its plain and ordinary meaning, to mean contributes or assists with the login by performing at least the actions explicitly recited in the claim. In this case, the sensor 204 in Kesanupalli enables the later accessing and presentation of the external resource credential, and therein facilitates. With respect to implementation in Challenger's TPM, this communication network does not change. The credentials would be received by the secure processor and transmitted to the general processor portion for communication to the external resource.

150. Kesanupalli describes this *transmitting output* (the user credential) to the external resource (website). (Kesanupalli) APPL-1006, 9:10-24 ("If the fingerprint information is authenticated, the procedure **retrieves credentials**

associated with the user based on the fingerprint information (block 908). ... The decrypted credentials ... are communicated to a requesting process or system (block 914).” See also, APPL-1006, 9:4-9. In particular, in the device of FIG. 2, the sensor 204 retrieves the user credential (*e.g.*, cryptographic key) the secure storage 206 and transmits, via the HOST PC 202, credentials to the website. In particular, the credentials are transmitted to application 214 of PC 202 for communication to the external resource. APPL-1006, 6:11-49. As discussed in limitation [1.1B], a connection is established between the device (in particular the first processor of PC 202) and the external resource. The sensor 204 sends decrypted credentials to the website by sending credentials to application 212 of the PC 202 (*first processor*), which, through application 214, communicates the decrypted credential to the website via this *established connection*. See (Kesanupalli) APPL-1006, 4:1-5, 4:51-63, 5:59-6:25 discussing application 112, which is substantially the same as application 214 of FIG. 2.

151. Both Kesanupalli and the ’512 patent describe their communication of credentials to website in terms of SSL (now TLS) or HTTPS protocols.

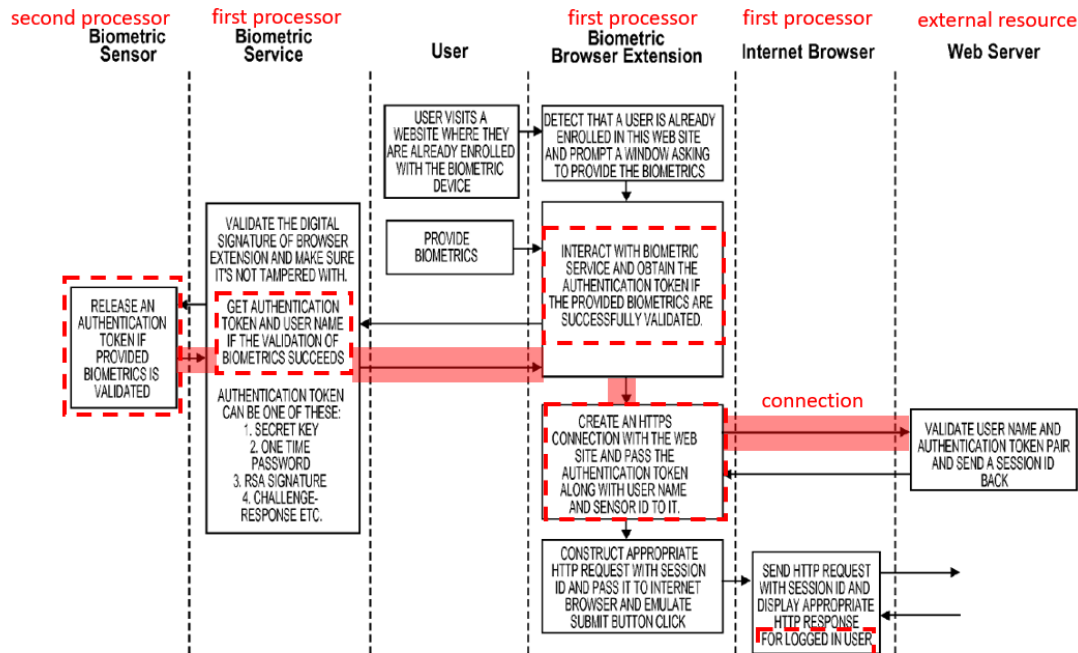
- (’512 patent) APPL-1001, 5:7-15 (“If a matching entry is found, the associated...username[] and site credential are extracted from the vault....[n]ext, [1] a secure connection is established between Alice’s computer and the service provider, and [2] Alice is authenticated....

[M]utual SSL can be used.”),

- (Kesanupalli) APPL-1006, FIG. 12 (“...get authentication token and user name if the validation of biometrics succeeds[, then]...[1] create an HTTPS connection with the website and [2] pass the authentication token along with the user name...to it.” HTTPS, also known as HTTP over SSL or HTTP over TLS, is described in RFC2818. (RFC2818, HTTP Over TLS) APPL-1023, 2 (summarizing TLS connection initiation and HTTP requests over the initiated TLS connection).

Thus, Kesanupalli and the '512 patent describe communication of a decrypted credential to the website via an *established connection* in like manners.

152. FIG. 12 annotated below illustrates transmitting user credentials to an external resource. FIG. 12 of Kesanupalli is illustrative of actions for authentication of a user, which includes actions performed by a “user,” actions performed by the components of the user’s system including by the circuitry of the sensor (*second processor*), and by applications on the Host PC (e.g., biometric service, browser extension, browser applications executed by *first processor*), and by an external resource’s web server. (Kesanupalli) APPL-1006 2:22-24, 9:52-57. FIG. 12 is annotated below illustrating the transmitting of output (user credentials) by the second processor (sensor) to the external resource



APPL-1006, FIG. 12, annotated

153. Kesanupalli's transmitting the decrypted user credential is transmitting *output based at least in part on the at least one the password or the cryptographic key* as it is output consisting of the (decrypted) user credential (cryptographic key) itself.

154. The output of the user credential to the external resource *facilitates a login of the user to the external resource* because based at least in part on the output of the user credential, the user is *logged in* to the external resource. POSITA would understand when Kesanupalli communicates the credentials "to the Web site to complete the user authentication process," the user login has occurred. (Kesanupalli) APPL-1006, 5:22-24, FIG. 12 ("logged in user"). This is the purpose of Kesanupalli's authentication procedures. See, e.g., (Kesanupalli)

APPL-1006, 1:59-61.

155. Summarizing, FIG. 9 of Kesanupalli is illustrative
FIG. 9 is a flow diagram depicting another embodiment of a procedure 900 for authenticating a user of a biometric authentication system. Initially, procedure 900 reads fingerprint information from a user's finger in contact with a fingerprint sensor (902). The procedure then authenticates the fingerprint information (block 904). If the fingerprint information is not authenticated, a message is generated indicating an authentication failure (block 906). If the fingerprint information is authenticated, the procedure retrieves credentials associated with the user based on the fingerprint information (block 908). The procedure then decrypts the user credentials (block 910) and identifies a unique identifier associated with the fingerprint sensor (block 912). The decrypted credentials and the unique identifier are communicated to a requesting process or system (block 914).

Kesanupalli, 9:10-24.

156. Thus, Kesanupalli and Challener (in combination) render obvious this limitation, including each portion as delineated above.

[1.7] [wherein the second processor is configured to:]

[1.7A] perform a secure backup of the record to a storage service,

[1.7B] wherein a second device associated with the user is registered with the storage service, and

[1.7C] wherein the record is downloaded from the storage service by the second device; and

157. Kesanupalli, Challenger and Cheng together teach this limitation [1.7], including [1.7A], [1.7B] and [1.7C]. Cheng is discussed above in Section VIII.A.3.

158. As I discuss below, Cheng describes techniques to perform a backup technique that provides data of a first device to a storage service such that it is accessible for download by a second device. For the reasons discussed above, it would be obvious to implement Cheng's technique in Kesanupalli-Challener device. In particular, it would have been obvious for the *second processor* (the sensor 204 functionality implemented on Challenger's secure processor 26 in its TPM-based system) to be configured to implement Cheng's technique and thus, for the second processor to be configured to *perform a secure backup*.

159. First, a POSITA would be motivated to configure the second processor with instructions to perform the backup process of Cheng because the secure backup is directed to the kind of data (user authentication information, electronic identity, etc.) that underlie security functions of the device and which a POSITA would want to isolate from insecure portions or processes in a Kesanupalli-Challener type device with backup. In addition, the relevant records

for backup are disposed in the secure storage accessible only by the second processor. Thus, using the second (secure) processor to perform Cheng's backup would ensure that the information accessed in the course of the backup procedure (e.g., biometric data and user credentials) is isolated from insecure portions or processes of the device while the resulting backup is reliably available in the presence of failures, theft of devices, etc. For example, Cheng states,

In the event of loss, damage, or destruction of the device, the user's authentication information, electronic identity and any data associated there with would be lost forever. Indeed, to prevent or at least to minimize the possibility of compromising the secrecy of the electronic identity and the authentication information, when a portable authentication device is reported lost or stolen, the general practice is to first deactivate or erase completely from the authentication system or secure network the electronic identity and authentication information associated with the lost/stolen authentication device and then create and register new ones from scratch.

(Cheng), APPL-1008, [0006].

160. Indeed, backing up signing data and keys of a TPM (such as the records in secure storage) was well known and widely deployed before the priority date—indeed an import management function of TPM-based systems was commonly used for backup and recovery of TPM data. For example, at least by

2006, it was common to securely backup critical TPM data by encrypting the TPM data for external storage and later decryption during a recovery procedure.

(Infineon TPM Key Backup and Recovery for Trusted Platforms), APPL-1018, p. 3 (describing secure backup protocol elements created by TCG standardization group and implemented by Infineon Professional Package for its TPM implementation). As explained in the Infineon Whitepaper, *TPM Key Backup and Recovery For Trusted Platforms*, to overcome the potential loss of keys and certificates in connection with system damage or loss,

...an Emergency Recovery mechanism is available together with the Infineon TPM professional SW package. This backup and migration capability is one of the cornerstones to allow the user a comfortable use of the applications without the danger of total data loss due to system errors and missing backup support.

...

All these functions are implemented within the Infineon Professional Package which is available as an OEM product and is part of a PC package outfitted with an Infineon TPM. This OEM package will be delivered from nearly every PC manufacturers, which integrates and sells trusted Computing (PC)-Platforms based on an Infineon TPM chip.

APPL-1018, (p 4, emphasis added)

161. A POSITA would understand that implementing the backup function of Cheng on the second processor (secure processor 26 of Challenger's TPM that also executes secure functions of sensor 204) would protect the process of securely backing up biometric data and user credentials from a myriad of potential attacks including the compromise of standard (insecure) portions of the operating system, main CPU, data in storage, etc. *See* (Challener) APPL-1007, 1:18-42. By executing the secure backup using the second processor, the backup procedure would be made substantially more secure, resistant, and in many cases, immune to the vast majority of attacks on computer systems.

162. Providing the second processor (secure processor) configured to perform the secure backup would be straightforward for a POSITA to implement. To the extent that Cheng's techniques are directed to a general-purpose processor, a POSITA would readily implement the techniques in the secure processor by porting the code to the native TPM environment and have a reasonable expectation of success. The process of porting code from OS-native to TPMs was well understood and widely practiced by the priority date. Moreover, the APIs and code structures for such a port were well documented at the time.¹⁵

¹⁵ APPL-1017, https://trustedcomputinggroup.org/wp-content/uploads/TPM-Main-Part-3-Commands_v1.2_rev116_01032011.pdf

163. A POSITA would also understand that it is desirable to update backup data as security relevant operations occur, e.g., as biometric data or credential changes or is added or deleted. Failure to update on these events would lead to data loss in the case of recovery, *e.g.*, if not updated on a timely basis, the changes occurring after the last backup would be lost and not be recoverable. Hence, a POSITA would understand to the advantage of incrementally updating a backup in correspondence with biometric data or credential changes. Further, a POSITA would have reason for such updates to be driven by the logic of the secure processor/TPM because it has the information to know when security-relevant updates occur. Further, the second processor includes circuitry with access to the secure data, and with the functionality to perform the techniques of Cheng such as encryption. Accordingly, a POSITA would have had reason to configure the second processor to be the creator of encrypted backup records and initiator of their transfer to backup service.

164. The second processor, the security processor performing the functions of sensor 204 as discussed above, limits access to the secure storage and the data stored therein (e.g., the record) for itself. Thus, a POSITA would have understood the sensor 204 would implement the retrieval of the secure data from the secure storage 206 for communication to a backup service.

165. First, as to limitation [1.7A], Cheng discloses a “biometrics-based

authentication device,” and a *secure backup* of data of this first device by creating “backup file(s)” of the encrypted data and providing these backup file(s) to a user service bureau, a *storage service*. (Cheng) APPL-1008, [0002] (“present invention relates generally to portable authentication devices. More particularly, it relates to a new and useful system, method, and apparatus for generating secure back up of authentication information of a user and for restoring the authentication information back onto a portable authentication device.”), [0023]-[0028] (“a method for creating a secure backup of a portable biometrics-based authentication device includes the following steps: (a) obtaining a lower tier encryption key from a user service bureau; (b) enciphering lower tier authentication data using the lower tier encryption key, thereby creating an encrypted lower tier backup file; (c) obtaining an upper tier encryption key from the user service bureau; (d) enciphering upper tier authentication data using the upper tier encryption key, thereby creating an encrypted upper tier backup file; and (e) storing the encrypted lower tier backup file and the encrypted upper tier backup file on a storage means.”), [0029] (“The storage means could be, for instance, an online proprietary or Internet-based storage service, a remote server...[t]he user service bureau could be proprietary or Internet-based and could also provide the storage service.”)

166. Cheng describes exemplary types of data for backing up process, which includes user authentication information, such as “private keys, certificates,

and other data held within the device” and “user’s biometrics information;” this is the same type of data included in Kesanupalli’s *record* (user credential and biometric template) as discussed above. (Cheng) APPL-1008, [0030]. *See also*, (Cheng) APPL-1008 [0007], [0008], Figs. 3A, 3B. Thus, a predictable implementation of Cheng’s backup to a storage service is to securely backup Kesanupalli’s *record* for the reasons discussed in the motivation to combine above.

167. Thus, Cheng’s backup is a *secure backup* as it creates a backup of the record (*e.g.*, backup file(s)) that is protected from unauthorized access (*e.g.*, by encryption) and transmits it to a storage service (*e.g.*, user service bureau). Cheng describes that its methods and systems “securely backup and restore a user’s authentication information.” (Cheng) APPL-1008, [0007]. *See also*, FIGS. 3A, 3B which illustrate a two-tier encryption process.

168. Cheng describes creating “backup file(s)” of data of a “biometrics-based authentication device,” which are provided from the device to a user service bureau that provides a *storage service* of the file(s). *See* (Cheng) APPL-1008, Abstract, [0002], [0007]-[0008], [0023] – [0030], FIGS. 3A and 3B. Cheng teaches examples of the data that is backed-up including authentication information such as a key and biometric data, which is a *record* just like the user credential of Kesanupalli discussed above in [1.6]. *See* (Cheng) APPL-1008, [0007], [0008], [0030]. Cheng at its FIG. 1 illustrates a structure including data

that is to be backed-up. (Cheng) APPL-1008, [0015] – [0017]. This is *record* comprising biometric data and user credential (e.g., cryptographic key), because the data is organized into collections of data arranged by type and relation, e.g., is a record.

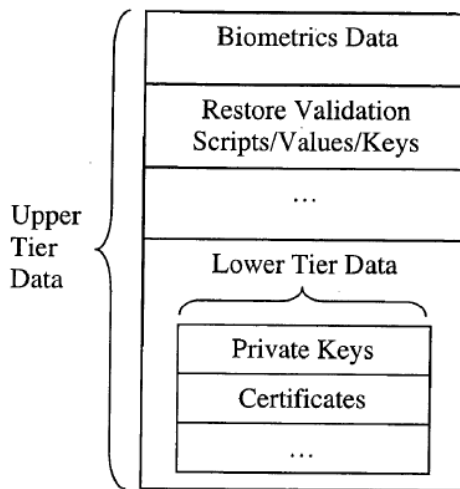


FIG. 1

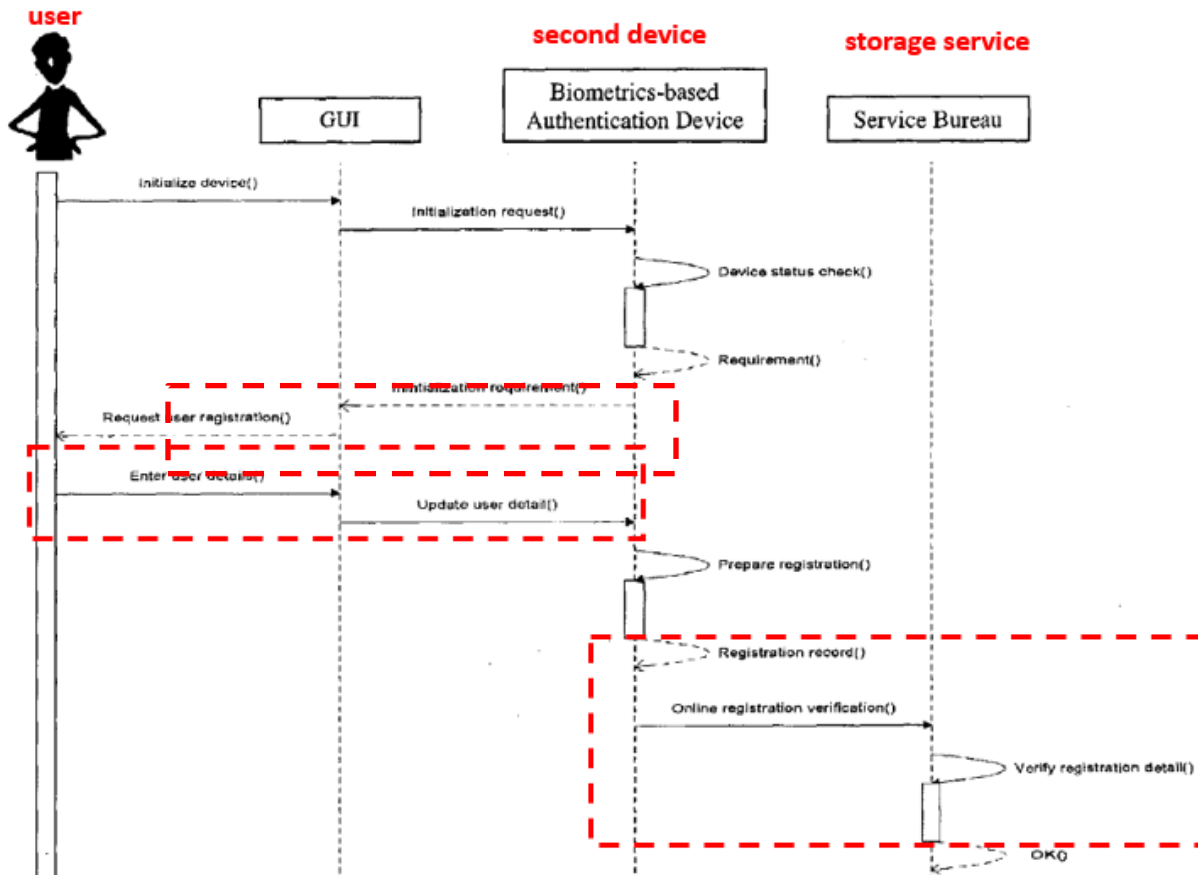
(Cheng) APPL-1008, FIG. 1

169. As discussed above, Cheng provides a *secure backup* as the record being backed-up is encrypted and an encrypted copy is transmitted by the device over a secure connection to the storage service. (Cheng) APPL-1008, FIGS. 3A-3B, [0023] – [0028]. And not only is the record itself *secure* due to the encryption, the transmission of the encrypted record to storage service is through “secure communication between the user service bureau [storage service] and the portable biometrics-based authentication device [first device] can be established.” (Cheng)

APPL-1008, [0029]. *See also*, (Cheng) APPL-1008, claim 5, Title, Abstract, [0002], [0023].

170. As to limitation [1.7B], Cheng describes embodiments where a restoration of the multiple-encrypted backup data onto a new device—a *second device*—is performed. To begin this restoration, Cheng’s backup procedure includes *register[ing] a second device associated with the user* with Cheng’s user service bureau, *the storage service*. (Cheng) APPL-1008, [0045] – [0047] (“1. A user’s biometrics-based authentication device becomes unavailable due to loss, damage, destruction, theft, etc. 2. The user obtains **a new** biometrics-based authentication **device**. ... 3. The user **enrolls** the new biometrics-based authentication device with an enrollment service/user service bureau.”); [0031]-[0039] including “The method of restoring authentication information of a user includes the following steps: (a) verifying registration information of the user with a user service bureau; (b) downloading an upper tier encryption key from the user service bureau...(d) restoring onto the portable biometric-based authentication device the upper tier authentication data from the decrypted upper tier backup file which includes the user’s backup biometrics data and any validation scripts, keys, and/or values; ...(h) restoring onto the portable biometrics-based authentication device the lower tier authentication data from the decrypted lower tier backup file.”. FIG. 4A, part of an “exemplary restore process” ([0013]) is illustrative of

the second device registering (“user details”), which are provided to the service bureau, the *storage service*, in a “registration verification.”



Cheng, portion of FIG. 4A, annotated

171. Thus, Cheng discloses a second device such as a user’s new device *is registered* with the storage service.

172. As to limitation [1.7C], Cheng describes data is *downloaded* from the service bureau, *storage service*, by the *second device*. (Cheng) APPL-1008, [0048] (“[t]he new biometrics-based authentication device establishes a secure connection with a user service bureau, begins the restore process and downloads backup data from storage.”) APPL-1008, [0048]. See also, (Cheng) FIG. 4A, 4B, [0039],

[0009] (“To restore the multiple-encrypted backup data onto a new [] device, ... upload the multiple-encrypted backup data onto the device.” *See also*, (Cheng) APPL-1008, [0009], [0031] – [0038], [0046] – [0049], [0050]. It is noted that the storage location is the service bureau and “up load backup file()” to the second device from the storage location means the file is *downloaded to the second device*.

173. In Kesanupalli as implemented on the TPM-based architecture of Challenger provides for the secure processor retrieves information from the secure storage to communicate with a general processor. To the extent that the Patent Owner argues that the second processor must solely be the performer of the backup procedure, Cheng states an onboard microprocessor of the biometric-based authentication device performs the backup. (Cheng) APPL-1008, [0022]. This suggests to a POSITA that the processor performing the authentication method – i.e., the second processor – performs the backup. (Cheng) APPL-1008. A POSITA would also find using the *second processor* to perform the backup both predictable use of the processor and such an instruction would be executed by the processor with a reasonable expectation of success due to the processor’s capabilities as discussed above in [1.3]. For example, the *second processor* has access to the records in secured storage and capability to perform the encryption taught by Cheng. (Challener) APPL-1007, FIG. 1, 4:24-36; (Kesanupalli) APPL-1006, 6:42-45. Further the TPM-based architecture provided by Challenger

provides a standardized interface (API) to implement secure (cryptographically supported) storage of data, e.g., sealed encryption.

174. As discussed above, a POSITA would recognize that applying Cheng's teachings to Kesanupalli's record would be a predictable benefit to Kesanupalli/Challener system. In particular, backing up data including at least one record to a second device allows for the user to save resources (*e.g.*, time) in recreating the data in the record (*e.g.*, biometric template and/or user credential).

175. Thus, Kesanupalli, Challener and Cheng (in combination) render obvious this element, including each portion as delineated above.

[1.8] a memory coupled to the first processor and configured to provide the first processor with instructions.

176. Kesanupalli and Challener together teach this element.

177. Kesanupalli's system includes a main memory, a *memory*. (Kesanupalli) APPL-1006, 10:58-11:47 ("Main **memory** is also usually included for storing and retrieving larger amounts of information such as data and software applications configured to perform functions according to the invention when executed by the central processing unit.") It is clear that this main memory of Kesanupalli's system is coupled to the *processor* and configured to provide the processor with *instructions* such as those that execute the functions of Host PC 202 and the sensor 204. (Kesanupalli) APPL-1006, 5:59-66, claim 26. This is the standard function of a memory and processor. *See also*, ('512 patent) APPL-1001, 2:21-23.

178. In particular, the memory is coupled to the *first processor* as the main memory provides instructions for general processing functions. Challenger confirms a memory 36, is coupled to a general processor 30, *a first processor*. APPL-1007, 5:26-40. APPL-1007, 7:67 – 8:7 (“Program instructions implementing all or part of the present invention can be stored on and be accessible from a computer readable medium...examples of which include memory....”).

179. Thus, Kesanupalli and Challenger (in combination) render obvious this element.

180. Therefore, claim 1 is obvious over Kesanupalli, Challenger, and Cheng.

8. Claim 2

[2.0] The system recited in claim 1, wherein the storage service comprises a cloud storage service

181. Kesanupalli, Challenger and Cheng disclose the elements of claim 1, and Cheng teaches the elements of claim 2.

182. Cheng describes its backup process comprise a storage service implemented as an “internet-based storage service.” (Cheng) APPL-1008, [0029]. An internet-based storage service is *a cloud storage service*. Cloud services are those are provided over large scale networks (such as the Internet) and accessed remotely by client hosts. Storage of data is one popular cloud service which was widely deployed and used by 2011. *See, e.g.*, APPL-1021 U.S. Patent Appl. No. 2007/0100913 showing geographically dispersed services for internet-based

storage service.

183. Thus, claim 2 is obvious over Kesanupalli, Challener and Cheng.

9. Claim 3

[3.0] The system recited in claim 1 wherein the second device is registered with the storage service at least in part by:

connecting to the storage service; and

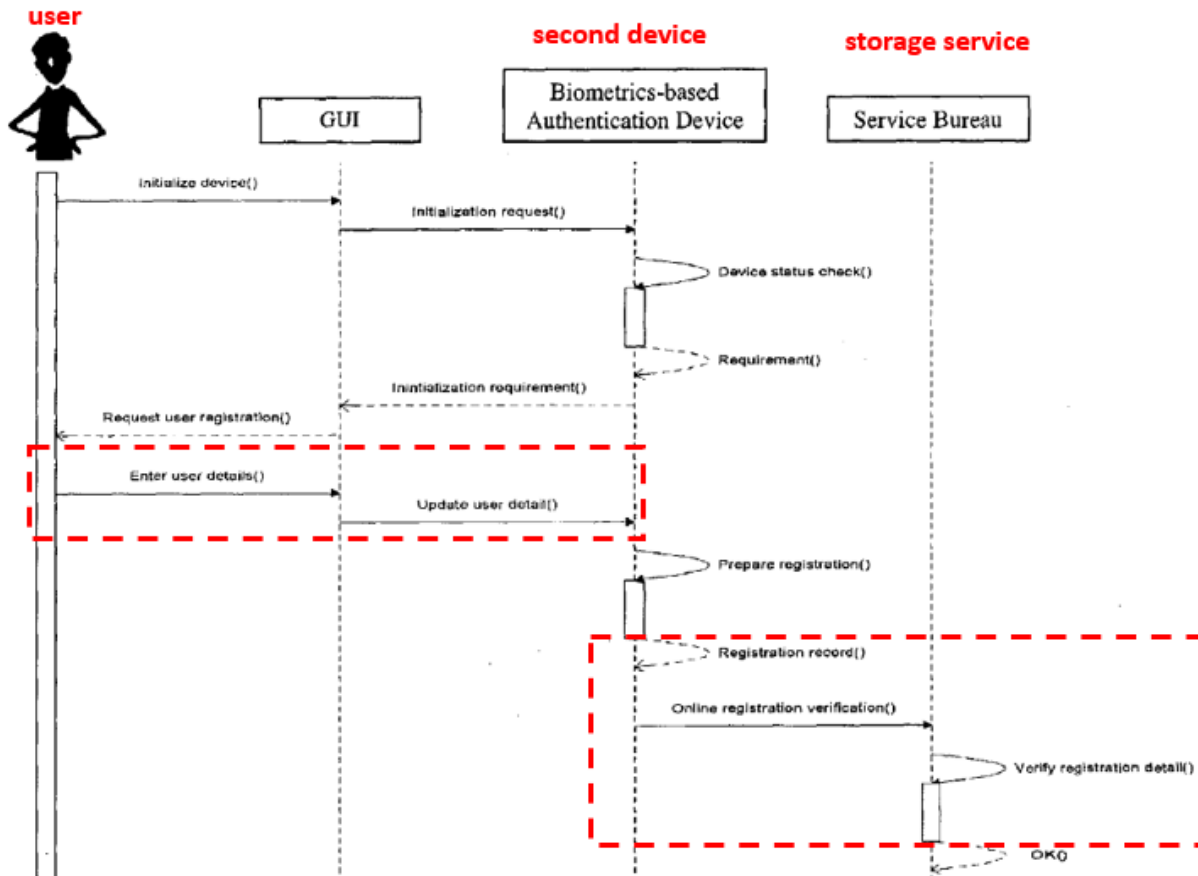
providing at least one of a user identifier or a credential to the storage service, wherein the user identifier comprises at least one of a username or an account number.

184. Kesanupalli, Challener and Cheng together teach the elements of claim 1, and Cheng further discloses the limitations of claim 3.

185. As discussed in limitation [1.6], Cheng describes the second device *connecting to the storage service*, which it refers to as the user service bureau. (Cheng) APPL-1008, [0048] (“The new biometrics-based authentication device establishes a secure connection with a user service bureau, begins the restore process and downloads backup data from storage”), FIG. 4A.

186. Cheng also discloses *providing at least one of a user identifier or a credential to the storage service*. FIG. 4A of Cheng shows the second device receiving “user detail(s),” from which the device forms a registration record that it sends - through “Online registration verification” - to the service bureau, the *storage service* (See limitation [1.6]). The service bureau then verifies the registration details. (Cheng) APPL-1008, [0032] (“verifying registration

information of the user with a user service bureau” to begin its process of restoring backed-up data by enrolling the second device. *See also*, (Cheng) APPL-1008, [0047].



Cheng, portion of FIG. 4A, annotated

187. One would understand that “user details” that provide registration information are a *user identifier*. To the extent that Cheng does not explicitly detail what identification features may be included in “user details,” it would have been conventional and obvious to include a *username* and/or *account number* as the registration information of Cheng. A user identifier of a *username* or an account number are typical identifiers used to enroll or register in any system. See,

e.g., (Kesanupalli) APPL-1006, 1:27-34. This is simply a standard piece of data for a service bureau, which is used by a plurality of users, to identify the relevant data for subsequent transfer to the requesting device.

188. Therefore, claim 3 is obvious in light of Kesanupalli, Challener and Cheng.

10. Claim 4

[4.0] The system recited in claim 1, wherein the record downloaded by the second device is encrypted.

189. Kesanupalli, Challener and Cheng teach the elements of claim 1, and Kesanupalli and Cheng further teach the limitations of claim 3.

190. As discussed above in limitation [1.7], Cheng discloses a secure backup including a two-tier encryption of the record that is backed-up. Cheng continues to describe restoring, which is downloading, the record on the second device where the *encrypted* record is downloaded by the second device:

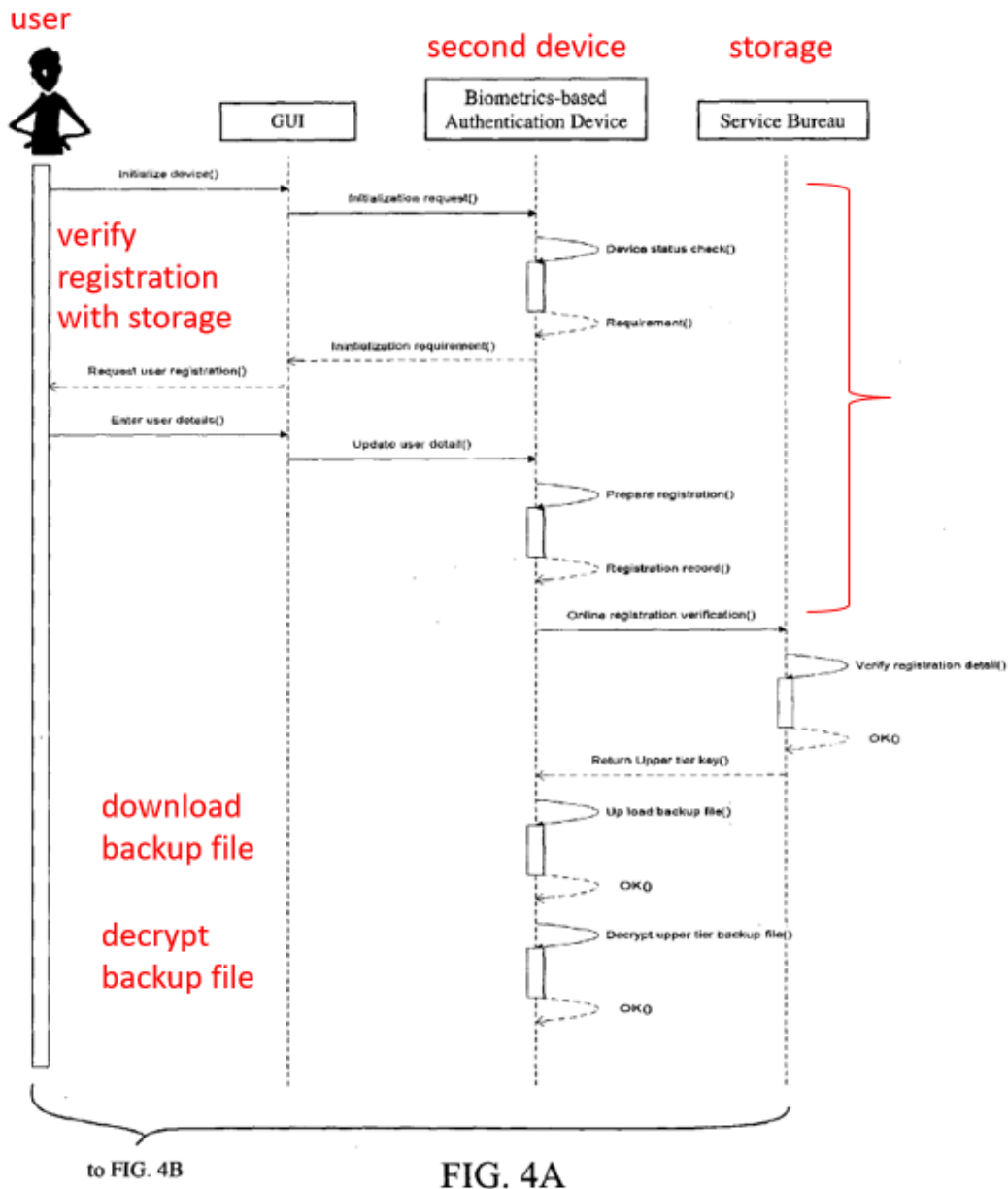
To restore the multiple-encrypted backup data onto a new biometrics-based authentication device, the user first needs to enroll the relevant biometrics in the new device and upload¹⁶ the multiple-encrypted backup data onto the device, then contact the corresponding user service

¹⁶ Uploading the data onto the second device means the data is downloaded from the storage to the second device.

bureau to obtain an access clearance to the encrypted lower and upper tier encryption keys. The access clearance enables the device to establish a secure connection with the user service bureau service. Upon establishing the secure connection, the restore process begins automatically. ...

(Cheng) APPL-1008, [0009]. *See also*, (Cheng) APPL-1008, [0044] – [0050].

Cheng at FIG. 4A – 4C illustrates “an exemplary restore process.” Cheng, [0013].



(Cheng) APPL-1008, FIG. 4A, annotated

191. Therefore, claim 4 is obvious in light of Kesanupalli, Challener and Cheng.

11. Claim 5

[5.0] The system recited in claim 4, wherein the encrypted record is stored to an insecure storage associated with the second device.

192. Kesanupalli, Challener and Cheng in combination teach the elements of claims 1 and 4, and Cheng in view of Kesanupalli and Challener also teach this element.

193. The '512 patent does not define “insecure storage” but states “[u]sers (and applications) can read from and write to the insecure storage area.” ('512 patent) APPL-1001, 4:4-5. Continuing from claim [4.0], as Cheng teaches downloading a record encrypted under Cheng’s two-tier encryption structure, a POSITA would have stored the received encrypted record in an *insecure storage* of the second device.

194. As discussed above with reference to claim 1, the system of Kesanupalli implemented on the architecture of Challener, includes a general portion (PC 202) separate from a security portion. Challener illustrates a general processing portion includes a storage suitable for storage of downloaded, encrypted data. (Challener) APPL-1007, 5:26-36, FIG. 1, storage device 36 (Challener) 5:26-32 (“A storage device 36, such as a hard disk drive, is also typically included to store data and applications to be used by the system 10. In the present invention, applications such as security applications to be loaded into processor 26, as well as other applications used by the processor 26 and microprocessor 30, can be stored securely on storage device 36 using encryption and other security methods.”) POSITA understand that storage device 36 is insecure storage at least because it is

accessible by multiple components. (Challener) APPL-1007, FIG. 1, 4:57-65, 5:26-40.

195. Additionally, to the extent that the '512 patent labels storage as “secure” or “insecure” this is merely a choice of two types of memory with predictable tradeoffs. As discussed above, as the file is encrypted and thus itself secure, and as such, suitable for *insecure storage*. This is merely a predictable use of prior art elements according to their established functions.

196. Therefore, claim 5 is obvious over Kesanupalli, Challener and Cheng.

12. Claim 6

[6.0] The system recited in claim 4, wherein the second device is configured to decrypt the downloaded record and store the decrypted record to a secure storage associated with the second device.

197. Kesanupalli, Challener and Cheng teach the elements of claims 1 and 4, and Cheng in view of Kesanupalli and Challener further teach this element.

198. First, Cheng discloses the *second device is configured to decrypt the downloaded record*.

To restore the multiple-encrypted backup data onto a new biometrics-based authentication device, the user first needs to enroll the relevant

biometrics in the new device and upload¹⁷ the multiple-encrypted backup data onto the device, then contact the corresponding user service bureau to obtain an access clearance to the encrypted lower and upper tier encryption keys. The access clearance enables the device to establish a secure connection with the user service bureau service. Upon establishing the secure connection, the restore process begins automatically. The device first requests the upper tier data decryption key from the user service bureau server to decipher the encrypted upper tier data. The device then compares the decrypted backup biometrics data with the newly enrolled biometrics data. If they match, then the newly enrolled biometrics data are replaced with the decrypted backup biometrics data. Only then, will the system confirm the match to the user service bureau server and request the lower tier decryption key. Once the lower tier decryption key is received, the lower tier data is deciphered and stored in the device. This completes the restore process. If they do not match, the restore process is terminated. When the restore process is complete or otherwise terminated, the device automatically

¹⁷ Uploading the data onto the second device means the data is downloaded from the storage to the second device.

disconnects from the user service bureau and communicates the results to the user.

(Cheng) APPL-1008, [0009]. *See also*, (Cheng) APPL-1008, [0044] – [0050], Cheng at FIG. 4A – 4C illustrates “an exemplary restore process.” [0013].

199. After decryption of the downloaded record, one would have reason to *store the decrypted record [in] a secure storage associated with the second device* as taught in Kesanupalli and Challener. This is merely storing the data in the second device, in its secure storage, just as taught by Kesanupalli and Challener discussed above. In particular, each of Kesanupalli and Challener suggest storing personal data in a *secure storage*. (Kesanupalli) APPL-1006, 6:42-49; (Challener) APPL-1007, 4:57-5:9. Further, there is a reason to place data, decrypted with respect to the two-tiered encryption of Cheng, in a *secure storage* that is protected from unauthorized access to further prevent identity theft and/or unwanted intruders from accessing the data.

200. Therefore, claim 6 is obvious over Kesanupalli, Challener and Cheng.

13. Claim 10

[10.0] The system recited in claim 1, wherein performing the cryptographic operation comprises decrypting the record.

201. Kesanupalli, Challener and Cheng together teach the elements of claim 1, and Kesanupalli further teaches this element.

202. Kesanupalli as implemented on the architecture of Challener discloses

a *second processor* configured to *perform a cryptographic operation* as discussed above in limitation [1.5], which illustrates *cryptographic operation* is *decrypting* the record. See Section VI.C, (Kesanupalli) APPL-1006, 6:31-49; (Challener) APPL-1007, 4:24-43.

203. Thus, claim 10 is obvious in light of Kesanupalli, Challener and Cheng.

14. Claims 11 – 16 and 20

204. The limitations of independent claim 11, except limitation [11.0], the preamble, are substantively identical to limitations of claims I have already addressed above, as correlated in the table below:

Claim 11 Limitation	Corresponding Claim Limitation of Claim 1 illustrating the substantially similar limitation of Claim 11 is disclosed in Kesanupalli, Challener and Cheng
[11.1] <i>based at least in part on a request associated with a user to access an external resource, establishing, by a first processor of a first device, a secure connection with the external resource; and communicating with a second processor using a restricted interface,</i>	[1.1A], [1.1B], [1.1C]
[11.2] <i>wherein the second processor is configured to: receive a biometric input from a sensor, wherein the biometric input corresponds to at least one of a fingerprint, a feature usable for facial recognition, a</i>	[1.2]

<i>voiceprint, a feature usable for a retina scan, or a typing feature; and</i>	
<i>[11.3] access a record stored in a secure storage, wherein the record is associated at least with the external resource;</i>	[1.3]
<i>[11.4] retrieve, from the record, at least one of a password, a cookie, or a cryptographic key;</i>	[1.4]
<i>[11.5] perform a cryptographic operation;</i>	[1.5]
<i>[11.6] in response to determining that there is a match between the biometric input and a stored biometric template accessed by the second processor, facilitate a login of the user to the external resource at least in part by transmitting, via the established connection, output based at least in part on the at least one of the password, the cryptographic key, or the cookie retrieved from the record, and wherein the user is logged in to the external resource based at least in part on the output; and</i>	[1.6] including [1.6A], [1.6B]
<i>[11.7] perform a secure backup of the record to a storage service, wherein a second device associated with the user is registered with the storage service, and wherein the record is downloaded from the storage service by the second device</i>	[1.7] including [1.7a], [1.7B], [1.7C]

[11.0] A method, comprising:

205. To the extent this preamble to the claim is limiting, Kesanupalli and Challener render it obvious. The citations to Kesanupalli, Challener and Cheng

above apply to a *system* as in claim 1 and also to a *method*. (Kesanupalli) APPL-1006, 2:30-38 (“The systems and methods described herein relate to biometric authentication of users. ...”); (Challener) APPL-1007, Abstract (“Method and apparatus for enabling applications on security processors of computer systems. ...”); (Cheng) APPL-1008, Abstract (“it relates to a new and useful system, method and apparatus for generating secure back up of authentication information of a user and restoring the authentication information back onto a portable authenticated device.”)

206. Thus, claim 11 is obvious in light of Kesanupalli, Challener and Cheng.

207. Dependent claims 12 – 16 and 20 are substantively identical to claims already addressed above, as correlated in the table below but depend from a method claim. The combination of Kesanupalli, Challener, and Cheng renders obvious these claims for the same reasons discussed above with respect to claim 11 and the corresponding claims in of the table.

Claim	Corresponding Claim
[12]	[2]
[13]	[3]
[14]	[4]
[15]	[6]
[16]	[5]
[20]	[10]

15. Claim 21

208. The limitations of claim 21, except the preamble of limitation [21.0], are substantively identical to limitations of claim 1 already addressed above, as correlated in the table below:

Claim 21 Limitation	Corresponding Claim Limitation of claim 1 illustrating the substantially similar limitation of claim 11 is disclosed in Kesanupalli, Challener and Cheng
[21.1] <i>based at least in part on a request associated with a user to access an external resource, establishing, by a first processor of a first device, a secure connection with the external resource; and communicating with a second processor using a restricted interface,</i>	[1.1A], [1.1B], [1.1C]
[21.2] <i>wherein the second processor is configured to: receive a biometric input from a sensor, wherein the biometric input corresponds to at least one of a fingerprint, a feature usable for facial recognition, a voiceprint, a feature usable for a retina scan, or a typing feature; and</i>	[1.2]
[21.3] <i>access a record stored in a secure storage, wherein the record is associated at least with the external resource;</i>	[1.3]
[21.4] <i>retrieve, from the record, at least one of a password, a cookie, or a cryptographic key;</i>	[1.4]
[21.5] <i>perform a cryptographic operation;</i>	[1.5]
[21.6] <i>in response to determining that there is a match between the biometric input and a stored biometric template accessed by the second</i>	[1.6]

<i>processor, facilitate a login of the user to the external resource at least in part by transmitting, via the established connection, output based at least in part on the at least one of the password, the cryptographic key, or the cookie retrieved from the record, and wherein the user is logged in to the external resource based at least in part on the output; and</i>	
<i>[21.7] perform a secure backup of the record to a storage service, wherein a second device associated with the user is registered with the storage service, and wherein the record is downloaded from the storage service by the second device.</i>	[1.7]

[21.0] A computer program product embodied in a non-transitory computer readable storage medium and comprising computer instructions for:

209. To the extent the preamble is limiting, Kesanupalli and Challener render it obvious.

210. The citations to Kesanupalli, Challener and Cheng above apply to a system, and also to a *computer program product embodied in a non-transitory computer readable storage medium*. Kesanupalli discloses a *computer program product* can comprise instructions for the processes including those executed by its sensor 204 and a computing host (PC) 202 discussed above. (Kesanupalli) APPL-1006, 9:58-10:12 (“configured to perform particular tasks according to the invention, by executing machine-readable software code that defines the particular tasks embodied by the invention.”), 13:8-39. Challener confirms what was well-understood by a POSITA, that its first and second processors perform the tasks

detailed by executing instructions comprised in *computer program product embodied in a non-transitory computer readable storage medium*. (Challener)

APPL-1007, 7:64-8:7.

211. Thus, claim 21 is obvious over Kesanupalli, Challener and Cheng.

IX. CONCLUSION

This declaration and my opinions herein are made to the best of my knowledge and understanding, and based on the material available to me, at the time of signing this declaration. I declare that all statements made herein on my own knowledge are true and that all statements made on information and belief are believed to be true, and further, that these statements were made with the knowledge that willful false statements and the like so made are punishable by fine or imprisonment, or both, under Section 1001 or Title 18 of the United States Code.



Date: January 19, 2024