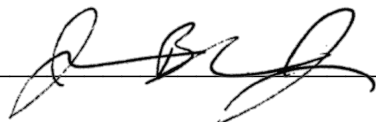


IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re Patent of: Scott MacDonald Ward
U.S. Patent No.: 10,992,480 Attorney Docket No.: 39843-0109PS1
Issue Date: April 27, 2021
Appl. Serial No.: 16/597,773
Filing Date: October 9, 2019
Title: METHOD AND SYSTEM FOR PERFORMING A
TRANSACTION AND FOR PERFORMING A VERIFICATION
OF LEGITIMATE ACCESS TO, OR USE OF DIGITAL DATA

DECLARATION OF DR. BLACK

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 of Title 18 of the United States Code.

By: 
John R. Black, Jr.

Date: Oct 29, 2021

Table of Contents

I.	QUALIFICATIONS AND BACKGROUND INFORMATION.....	4
II.	OVERVIEW OF CONCLUSIONS FORMED	8
III.	LEVEL OF ORDINARY SKILL IN THE ART	9
IV.	LEGAL STANDARDS	10
	A. Terminology.....	10
	B. Legal Standards.....	11
	1. Anticipation.....	11
	2. Obviousness	12
V.	THE '480 PATENT.....	16
	A. Overview of the '480 Patent	16
	B. Prosecution History of the '480 Patent.....	19
VI.	OVERVIEW AND COMBINATIONS OF PRIOR ART REFERENCES ..	20
	A. Ellison	20
	B. Muir.....	32
	C. Combination of Ellison and Muir	35
	D. Al-Salqan	39
	E. Combination of Ellison, Muir, and Al-Salqan.....	43
	F. Searle.....	46
	G. Combination of Ellison, Muir, and Searle	50
VII.	MANNER IN WHICH THE PRIOR ART REFERENCES RENDER THE '480 CLAIMS UNPATENTABLE	53
	A. Claim 1	53
	B. Claim 3	76
	C. Claim 6.....	78
	D. Claim 7.....	80
	E. Claim 8.....	81
	F. Claim 9.....	81
	G. Claim 10.....	82
	H. Claim 11	83
	I. Claim 12.....	84
	J. Claim 13.....	85
	K. Claim 14.....	86
	L. Claim 15.....	87
	M. Claim 16.....	89
	N. Claim 17.....	94
	O. Claim 18.....	96

P.	Claim 19.....	97
VIII.	THE '480 CLAIMS DO NOT HAVE WRITTEN SUPPORT	98
A.	The '480 Patent does not describe a private key in the manner recited in the claims	98
1.	The '480 Patent does not describe providing a private key in a memory that is a secure part of the BIOS or any other secure location in the electronic device operated by the first transaction party (independent claims 1, 16-18).....	100
2.	The '480 Patent does not describe the private key being inaccessible to the user (independent claims 1 and 16).....	102
3.	The '480 Patent does not describe the private key being encrypted when the private key is stored in the memory (claims 1, 16-18)	103
4.	The '480 Patent does not describe the private key being decrypted in a secure processing environment inaccessible to said user, to any user-operated software, and to said operating system (claims 1, 16-18)	104
B.	The '480 Patent does not describe claim 2	108
C.	The '480 Patent does not describe claims 4 and 5.....	110
D.	The '480 Patent does not describe claims 16, 18, and 19.....	111
E.	The priority applications and original claims do not cure the written description deficiencies in the '480 Patent	113
IX.	CONCLUSION.....	115

I. QUALIFICATIONS AND BACKGROUND INFORMATION

1. My education and experience are described more fully in the attached curriculum vitae (APPENDIX A). For ease of reference, I have highlighted certain information below.

2. I hold a Bachelor of Science from the California State University at Hayward (now “California State University, East Bay”) in Mathematics and Computer Science, conferred in 1988. After my B.S. degree in 1988, I worked for 6 years for Ingres Corp. as a software developer. I received a Doctor of Philosophy in Computer Science from the University of California at Davis in 2000. My Ph.D. thesis is titled “Message Authentication Codes” (MAC), which provides examples of MACs, discusses MAC-related attacks, and emphasizes the Carter-Wegman style of MACs.

3. After receiving my Ph.D., I became an Assistant Professor of Computer Science at the University of Nevada, Reno, and, in 2002, I was appointed as an Assistant Professor of Computer Science at my current institution, which is the University of Colorado at Boulder. I was awarded tenure in 2008. My area of specialty is cryptography and security, including the security of computer systems and networks, and protecting digital property from illicit copying and distribution via encryption and authentication.

4. I have taught more than 40 classes in computer science, on subjects related to cryptography and security, and have authored or coauthored more than 25 journal and conference papers. Since 2011, I have consulted on industrial security projects that were put into commercial products; for example:

- I helped design CMAC, used in EAP-PSK, as part of Enterprise WiFi security;
- I was hired by Fitbit to help design their encryption and authentication algorithm for transferring data from a device to a phone or laptop over Bluetooth;
- I designed and implemented the security architecture for the Comverge Pro1 smart thermostat; and
- I implemented the DRM facility for the Yonder music app for the Android platform.

5. In addition to these activities, I have authored chapters related to cryptography and authenticated encryption for two books and have served as a referee and on the programs committee for numerous journals and conferences for the last two decades. I also served as an Advisor to numerous PhD and Masters students pursuing research in cybersecurity related fields. In 2002, I was the recipient of the NSF Career Award and have been the recipient of over \$ 1 million in grant money from the NSF for my research in cryptography.

6. I have particular experience with the types of security algorithms utilized in the Asserted Patents and related art, such as encryption technology, digital signatures, hashing, and so forth.

7. I have been retained on behalf of Samsung Electronics Co., Ltd. and Samsung Electronics America, Inc. to offer technical opinions relating to U.S. Patent No. 10,992,480 (“**the ’480 Patent**”) and prior art references relating to its subject matter. I have reviewed the ’480 Patent, and relevant excerpts of the prosecution history of the ’480 Patent. I have also reviewed the following prior art references:

Prior Art Reference
U.S. Patent No. 7,082,615 (“Ellison” or SAMSUNG-1006)
PCT Pub. No. WO 01/93212 (“Muir” or SAMSUNG-1008)
U.S. Patent No. 6,549,626 (“Al-Salqan” or SAMSUNG-1007)
U.S. Patent No. 6,683,954 (“Searle” or SAMSUNG-1009)
EP Patent Application Pub. No. EP 1465092 (“Shen” or SAMSUNG-1010)
EP Patent Application Pub. No. EP 1240568 (“Ansell” or SAMSUNG-1011)
U.S. Patent No. 11,063,766 (“the ’766 Patent” or SAMSUNG-1012)

PCT application PCT/NL2003/000436 (SAMSUNG-1013)
PCT application PCT/NL2004/000422 (SAMSUNG-1014)
U.S. Patent No. 5,872,917 (“Hellman” or SAMSUNG-1019)
U.S. Patent No. 7,801,775 (“Roseman” or SAMSUNG-1020)
U.S. Patent No. 6,898,288 (“Chui” or SAMSUNG-1021)
Excerpts from the Dictionary of Electrical & Computer Engineering, McGraw-Hill, 2004 (SAMSUNG-1022)

8. I have also reviewed various supporting references and other documentation as further noted in my opinions below.

9. Counsel (Fish & Richardson) has informed me that I should consider these materials through the lens of one of ordinary skill in the art related to the ’480 Patent at the time of the earliest possible priority date of the ’480 Patent, and I have done so during my review of these materials. The application leading to the ’480 Patent was filed on October 9, 2019 and claims priority from an earlier application filed on June 13, 2003. I have used June 13, 2003 for the purposes of my analysis below although Counsel has informed me that there may be some issues with the priority claim.

10. I have no financial interest in the outcome of this proceeding. I am being compensated for my work as an expert on an hourly basis. My compensation is not dependent on the outcome of these proceedings or the content of my opinions.

11. In writing this declaration, I have considered the following: my own knowledge and experience, including my work experience in the fields of computer science generally and computer security specifically; my experience in teaching those subjects; and my experience in working with others involved in those fields. In addition, I have analyzed various publications and materials, in addition to other materials I cite in my declaration.

12. My opinions, as I explained below, are based on my education, experience, and expertise in the fields relating to the '480 Patent. Unless otherwise stated, my testimony below refers to the knowledge of one of ordinary skill in the art as of June 2003. Figures appearing within this document have been prepared with the assistance of Counsel and reflect my understanding of the '480 Patent and the prior art discussed below.

II. OVERVIEW OF CONCLUSIONS FORMED

13. This declaration explains the conclusions that I have formed based on my analysis. To summarize those conclusions, based upon my knowledge and experience and my review of the prior art references listed above, I believe that:

- The '480 Patent does not describe or support all the features recited in claims 1-19.
- Claims 1, 3, 6, 11, and 14-19 are obvious over Ellison in view of Muir.
- Claims 7, 9, 13 are obvious over Ellison in view of Muir and Al-Salqan.
- Claims 8, 10, 12 are obvious over Ellison in view of Muir and Searle.

14. In support of these conclusions, I provide an overview of the references in Section VI and more detailed comments regarding the obviousness of claims 1-19 (**“the Challenged Claims”**) of the '480 Patent in Section VII.

III. LEVEL OF ORDINARY SKILL IN THE ART

15. A person of ordinary skill in the art at the time of the '480 Patent (a “POSITA”) would have had a Bachelor’s degree in an academic discipline emphasizing the design of electrical, computer, software technologies, or a related field, in combination with training or at least two years of related work experience with software security technologies such as cryptography or encryption/decryption systems, or in the development of hardware and software for carrying out secure electronic transactions. Alternatively, the person could have also had a Masters or Doctor of Philosophy degree in a relevant academic discipline with less than a year of related work experience in the same discipline.

16. Based on my experiences, I have a good understanding of the capabilities of a POSITA. Indeed, I have taught, mentored, participated in organizations, and worked closely with many such persons over the course of my career. Based on my knowledge, skill, and experience, I have an understanding of the capabilities of a POSITA. For example, from my industry consulting or conference interactions, I am familiar with what a POSITA would have known and found predictable in the art. From teaching and supervising my post-graduate students, I also have an understanding of the knowledge that a person with this academic experience possesses. Furthermore, I possess those capabilities myself.

IV. LEGAL STANDARDS

A. Terminology

17. I have been informed by Counsel and understand that the best indicator of claim meaning is its usage in the context of the patent specification as understood by one of ordinary skill. I further understand that the words of the claims should be given their plain meaning unless that meaning is inconsistent with the patent specification or the patent's history of examination before the Patent Office. Counsel has also informed me, and I understand that, the words of the claims should be interpreted as they would have been interpreted by one of ordinary skill at the time of the invention was made (not today). I have been informed by

Counsel that I should use June 13, 2003 as the point in time for claim interpretation purposes.

B. Legal Standards

18. I have been informed by Counsel and understand that documents and materials that qualify as prior art can render a patent claim unpatentable as being anticipated or obvious.

19. I am informed by Counsel and understand that all prior art references are to be looked at from the viewpoint of a person of ordinary skill in the art at the time of the invention, and that this viewpoint prevents one from using his or her own insight or hindsight in deciding whether a claim is anticipated or rendered obvious.

1. Anticipation

20. I understand that patents or printed publications that qualify as prior art can be used to invalidate a patent claim as anticipated or as obvious.

21. I understand that, once the claims of a patent have been properly construed, the second step in determining anticipation of a patent claim requires a comparison of the properly construed claim language to the prior art on a limitation-by-limitation basis.

22. I understand that a prior art reference “anticipates” an asserted claim, and thus renders the claim invalid, if all limitations of the claim are disclosed in that prior art reference, either explicitly or inherently (i.e., necessarily present).

2. Obviousness

23. I understand that even if a patent is not anticipated, it is still invalid if the differences between the claimed subject matter 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 POSITA.

24. I have been informed by Counsel and understand that a claim is unpatentable for obviousness and that obviousness may be based upon a combination of prior art references. I am informed by Counsel and understand that the combination of familiar elements according to known methods is likely to be obvious when it does no more than yield predictable results. However, I am informed by Counsel and understand that a patent claim composed of several elements is not proved obvious merely by demonstrating that each of its elements was, independently, known in the prior art.

25. I am informed by Counsel and understand that when a patented invention is a combination of known elements, a court determines whether there was an apparent reason to combine the known elements in the fashion claimed by the patent at issue by considering the teachings of prior art references, the effects of demands known to people working in the field or present in the marketplace, and the background knowledge possessed by a person having ordinary skill in the art.

26. I am informed by Counsel and understand that a patent claim composed of several limitations is not proved obvious merely by demonstrating that each of its limitations was independently known in the prior art. I am informed by Counsel and understand that identifying a reason those elements would be combined can be important because inventions in many instances rely upon building blocks long since uncovered, and claimed discoveries almost of necessity will be combinations of what, in some sense, is already known. I am informed by Counsel and understand that it is improper to use hindsight in an obviousness analysis, and that a patent's claims should not be used as a "roadmap."

27. I am informed by Counsel and understand that an obviousness inquiry requires consideration of the following factors: (1) the scope and content of the prior art, (2) the differences between the prior art and the claims, (3) the level of ordinary skill in the art, and (4) any so called "secondary considerations" of non-obviousness, which include: (i) "long felt need" for the claimed invention, (ii) commercial success attributable to the claimed invention, (iii) unexpected results of the claimed invention, and (iv) "copying" of the claimed invention by others.

28. I have been informed by Counsel and understand that an obviousness evaluation can be based on a single reference or a combination of multiple prior art references. I understand that the prior art references themselves may provide a suggestion, motivation, or reason to combine, but that the nexus linking two or

more prior art references is sometimes simple common sense. I have been informed by Counsel and understand that obviousness analysis recognizes that market demand, rather than scientific literature, often drives innovation, and that a motivation to combine references may be supplied by the direction of the marketplace.

29. I have been informed by Counsel and understand that if a technique has been used to improve one device, and a person of ordinary skill at the time of invention would have recognized that it would improve similar devices in the same way, using the technique is obvious unless its actual application is beyond his or her skill.

30. I have been informed by Counsel and understand that practical and common sense considerations should guide a proper obviousness analysis, because familiar items may have obvious uses beyond their primary purposes. I have been informed by Counsel and understand that a person of ordinary skill looking to overcome a problem will often be able to fit together the teachings of multiple prior art references. I have been informed by Counsel and understand that obviousness analysis therefore takes into account the inferences and creative steps that a person of ordinary skill would have employed at the time of invention.

31. I have been informed by Counsel and understand that a proper obviousness analysis focuses on what was known or obvious to a person of ordinary skill at the

time of invention, not just the patentee. Accordingly, I understand that any need or problem known in the field of endeavor at the time of invention and addressed by the patent can provide a reason for combining the elements in the manner claimed.

32. I have been informed by Counsel and understand that a claim can be obvious in light of a single reference, without the need to combine references, if the elements of the claim that are not found explicitly or inherently in the reference can be supplied by the common sense of one of skill in the art.

33. I have been informed by Counsel and understand that there must be a relationship between any such secondary considerations and the invention, and that contemporaneous and independent invention by others is a secondary consideration supporting an obviousness determination.

34. In sum, my understanding is that prior art teachings are properly combined where one of ordinary skill having the understanding and knowledge reflected in the prior art and motivated by the general problem facing the inventor, would have been led to make the combination of elements recited in the claims. Under this analysis, the prior art references themselves, or any need or problem known in the field of endeavor at the time of the invention, can provide a reason for combining the elements of multiple prior art references in the claimed manner.

35. I have been informed by Counsel and understand that in a post grant review or an *inter partes* review (IPR), “the petitioner shall have the burden of proving a

proposition of unpatentability,” including a proposition of obviousness, “by a preponderance of the evidence.” 35 U.S.C. §§ 316(e)/326(e).

V. THE ’480 PATENT

A. Overview of the ’480 Patent

36. Digital data is regularly exchanged to perform various transactions, e.g., financial transactions, in which the need to verify the identity of each party is quite important. SAMSUNG-1001, 1:1-35. PINs and passwords known to one of the parties can be used, but sending this information over digital networks can be a security risk. SAMSUNG-1001, 1:19-35. Methods involving the use of a third party transaction server have also been developed to enhance security. However, these methods may undesirably require additional hardware such as tokens and token readers. SAMSUNG-1001, 1:36-52. The ’480 Patent proposes a solution to this problem without requiring additional hardware. SAMSUNG-1001, 1:52-56.

37. The ’480 Patent is directed to “providing authentication data and authentication software to an electronic device and preferably stored in a secure storage location or other location inaccessible to the user or the operating system of the device.” SAMSUNG-1001, Abstract. FIG. 3 (reproduced below) illustrates an embodiment to which the ’480 Patent claims are directed.

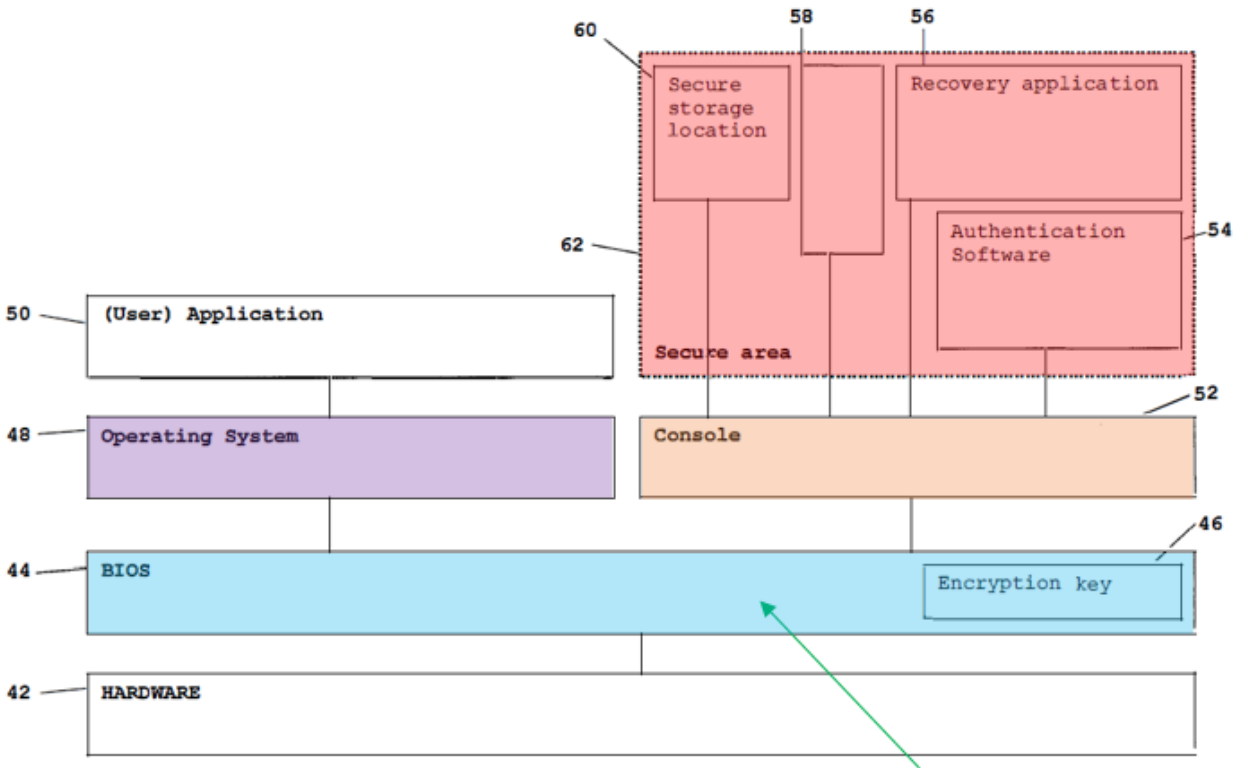


FIG. 3

BIOS 44 may block OS 48 from
accessing console 52 and secure area 62

SAMSUNG-1001, FIG. 3¹

38. In FIG. 3, “[t]he BIOS 44 controls most of the instructions and data flowing from one component to another. Data or instructions coming from one component and addressed to another need to pass through the BIOS 44. The BIOS 44 may check whether the instructions to the other component are allowed or not.”

SAMSUNG-1001, 8:22-27. “All the hardware devices are controlled by the BIOS

¹ The annotations I made to the figures in this declaration are shown in color.

44.” *Id.*, 8:15-20. “The BIOS 44 may comprise an encryption key 46.” *Id.*, 8:29-30.

39. “The operating system 48 creates a run-time environment for any user applications” and “is an interface between a user and the hardware 42.”

SAMSUNG-1001, 8:33-47. The operating system 48 may be prohibited by the BIOS 44 “to access certain parts of a hard drive or start certain applications.” *Id.*

40. Console 52 instructs “the hardware devices 42 via the BIOS 44 to start and report their status.” SAMSUNG-1001, 8:49-61. “A user may not interrupt or influence the console 52. Any instructions coming from the operating system 48 intended for the console 52 may therefore be blocked by the BIOS 44.” *Id.*

41. “In or behind the console 52, there is a secure area 62 only accessible to the BIOS.” SAMSUNG-1001, 8:62-9:6. “[T]he operating system 48 does not know that the applications and data in the storage locations [in the secure area 62] are present and maybe even running in a separate environment.” *Id.* “An authentication table may be securely stored in the secure storage location 60. In such a part of the computer, commonly seen as a part of the BIOS 44, the authentication table is unreachable for the operating system 48 and thus for a user.” *Id.*, 9:7-11. The “authentication software 54 may be installed in an application environment in the secure area 62 such that it may obtain the authentication table without passing through an unsecured part of the device.” *Id.*, 9:16-19. “By

locking the authentication table and the authentication software 54 including the digital signing algorithm in the secure area 62 they are secured.” *Id.*, 9:30-35.

B. Prosecution History of the '480 Patent

42. The '480 Patent issued from U.S. Patent Application No. 16/597,773 (“**the '773 Application**”). SAMSUNG-1001, cover. After failing to get an allowance in response to two office actions, Patent Owner amended the claims by replacing “authentication data” with “private key.” SAMSUNG-1002, 91-96. 1001. In recognition of the narrowing effect of this amendment and with an additional Examiner’s amendment to claims 27 and 28, the '773 Application was allowed. SAMSUNG-1002, 29-31. 1001. In particular, the Examiner indicated that by replacing “authentication data” with “private key” the rejections were overcome purportedly because the art cited by the Examiner did not teach “encrypting the private key for storage by another inaccessible key.” SAMSUNG-1002, 80, 111. *Remarkably, no such language is recited in the claims.* The issued claims recite “wherein the private key is encrypted when the private key is stored in said memory,” not “encrypting the private key for storage by another inaccessible key.” *See e.g.*, SAMSUNG-1001, 17:63-18:29 ('480 Patent’s claim 1). Thus, the rationale and language relied upon by the Examiner to allow the claims are untethered to the actual claims.

43. As I explain in the following parts of my declaration, the '480 Patent claims are obvious and further lack written description support in the specification of the '480 Patent.

VI. OVERVIEW AND COMBINATIONS OF PRIOR ART REFERENCES

A. Ellison²

44. Ellison discloses a computer system with “an isolated execution architecture” to prevent attacks and vulnerabilities in electronic and software systems. SAMSUNG-1006, 1:17-51, 2:40-61. An example illustration of Ellison’s “isolated execution architecture” is shown in Ellison’s FIG. 1A (reproduced below).

² General descriptions that I provided for the references and combinations discussed in Section VI of my declaration are incorporated into each subsection addressing/applying those references, as are the discussions of combinations.

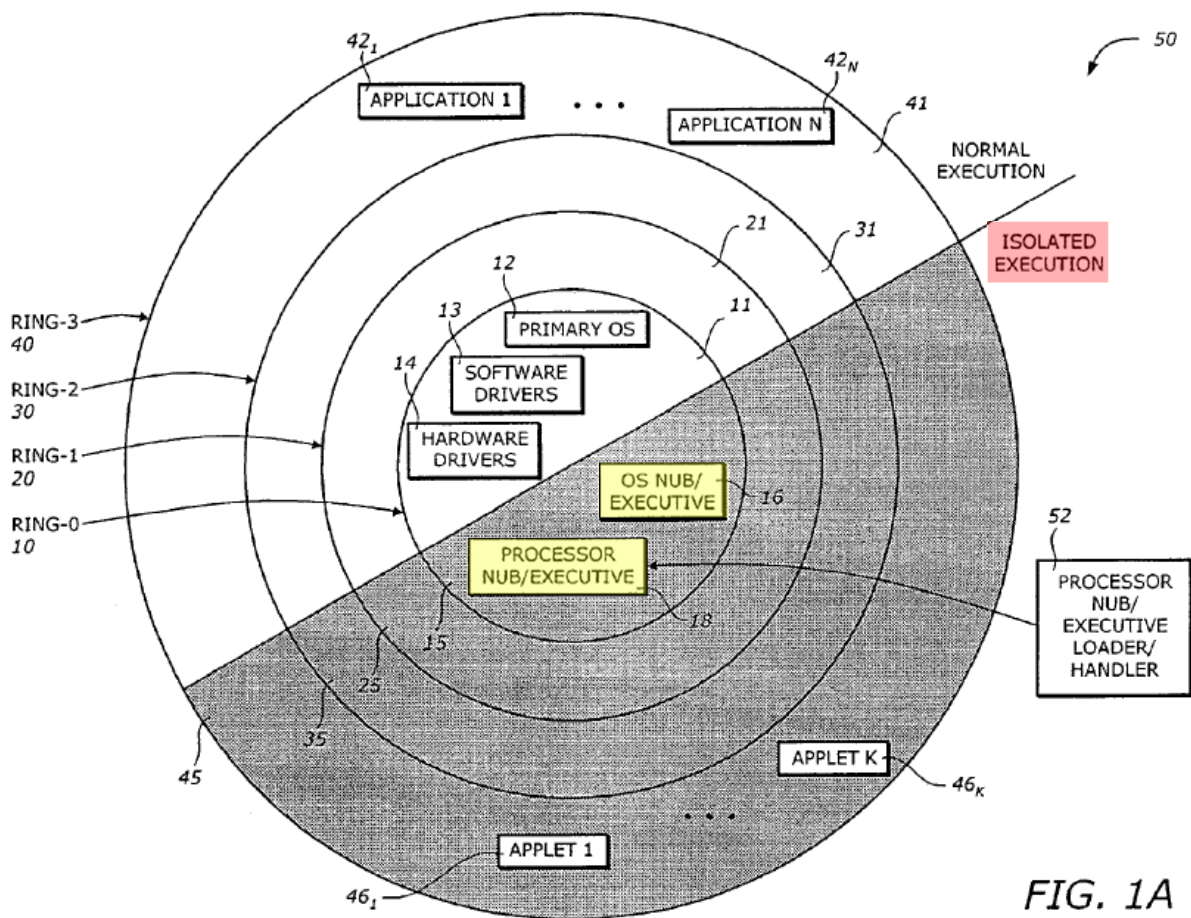


FIG. 1A

SAMSUNG-1006 (Ellison), FIG. 1A

45. The logical operating architecture 50 shown in FIG. 1A “includes ring-0 10, ring-1 20, ring-2 30, ring-3 40, and a processor nub loader 52.” SAMSUNG-1006, 2:62-3:1. The different ring levels are associated with different privilege levels. SAMSUNG-1006, 2:41-61. Ellison explains that:

A ring is a logical division of hardware and software components that are designed to perform dedicated tasks within the operating system. The division is typically based on the degree or level of privilege,

namely, the ability to make changes to the platform. For example, a ring-0 is the innermost ring, being at the highest level of the hierarchy. Ring-0 encompasses the most critical, privileged components. In addition, modules in Ring-0 can also access to lesser privileged data, but not vice versa. Ring-3 is the outermost ring, being at the lowest level of the hierarchy. Ring-3 typically encompasses users or applications level and has the least privilege. Ring-1 and ring-2 represent the intermediate rings with decreasing levels of privilege. SAMSUNG-1006, 2:41-61.

46. “The logical operating architecture 50 has two modes of operation: normal execution mode and isolated execution mode. Each ring in the logical operating architecture 50 can operate in both modes. The processor nub loader 52 operates only in the isolated execution mode.” SAMSUNG-1006, 3:4-8. “Ring-0 10 includes two portions: a normal execution Ring-0 11 and an isolated execution Ring-0 15.” SAMSUNG-1006, 3:8-10. “The isolated execution Ring-0 15 includes an operating system (OS) nub 16 and a processor nub 18.” SAMSUNG-1006, 3:15-16.

47. The operating system (OS) nub 16 and the processor nub 18 are “instances of an OS executive (OSE) and processor executive (PE), respectively. The OSE and the PE are part of executive entities that operate in a secure environment associated with the isolated area 70 and the isolated execution mode.” SAMSUNG-1006, 3:9-25.

48. Ellison's isolated execution architecture includes "an isolated region in the system memory" (*see* isolated area 70 in physical memory 60 shown in FIG. 1B (reproduced below)) and "is protected by both the processor and chipset in the computer system." SAMSUNG-1006, 3:32-36, FIGS. 1A, 1B. "The isolated area 70 is accessible only to elements of the operating system and processor operating in isolated execution mode." SAMSUNG-1006, 4:8-22. "The operating system nub 16 provides links to services in the primary OS 12 (e.g., the unprotected segments of the operating system), provides page management within the isolated area [70], and has the responsibility for loading ring-3 application modules 45, including applets 46 to 46, into protected pages allocated in the isolated area [70]." SAMSUNG-1006, 3:32-36, 4:30-37. "[T]he operating system nub 16 is also responsible for encrypting and hashing the isolated area pages before evicting the page to the ordinary memory, and for checking the page contents upon restoration of the page." SAMSUNG-1006, 3:64-4:7.

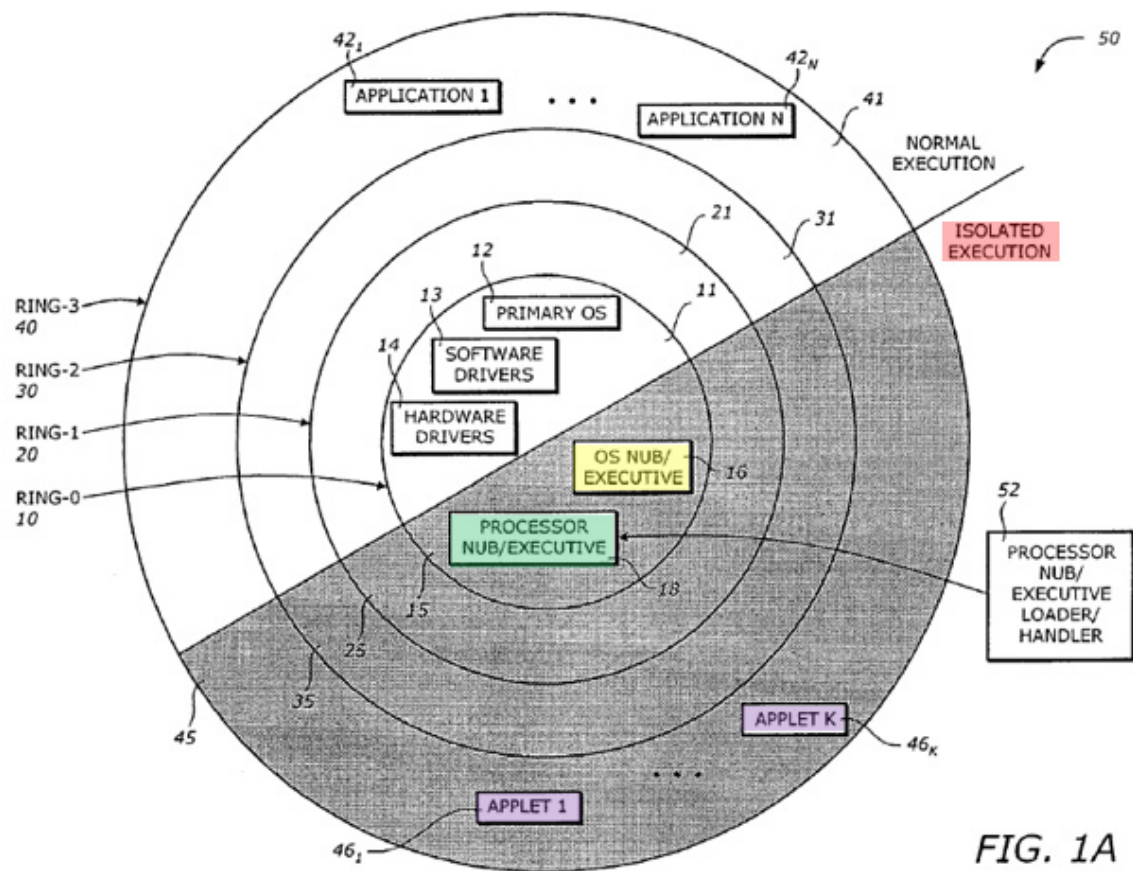


FIG. 1A

SAMSUNG-1006 (Ellison), FIG. 1A

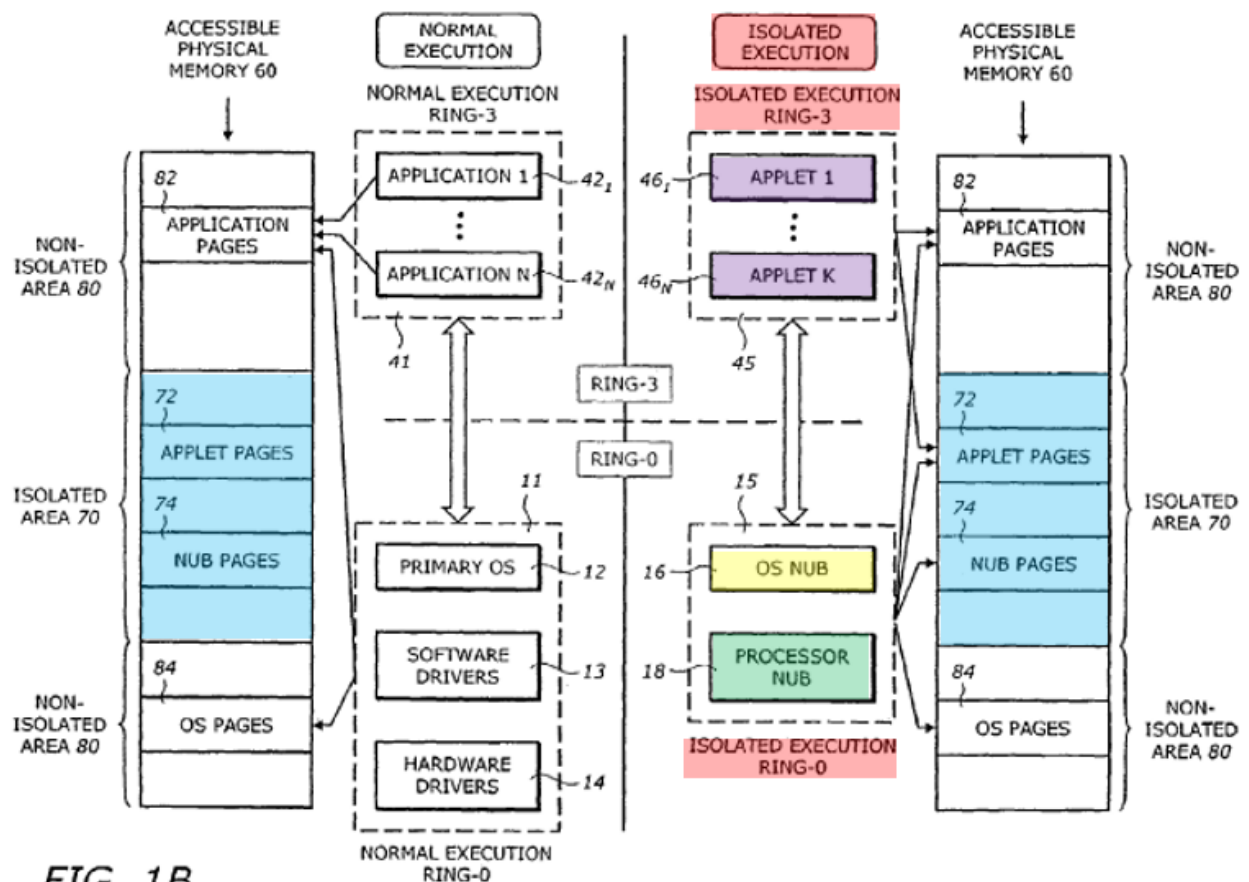


FIG. 1B

SAMSUNG-1006 (Ellison), FIG. 1B

49. “The isolated mode applets 46₁ to 46_N and their data *are tamper-resistant and monitor-resistant from all software attacks from other applets, as well as from non-isolated space applications* (e.g., 42₁ to 42_N), dynamic link libraries (DLLs), *drivers and even the primary operating system 12. Only the processor nub 18 or the operating system nub 16 can interfere with or monitor the applet’s execution.*” SAMSUNG-1006, 4:1-7.

50. A computer system 100 design that implements Ellison’s isolated execution architecture is shown in Ellison’s FIG. 1C (reproduced below). “The computer

system 100 includes a processor 110, a host bus 120, a memory controller hub (MCH) 130, a system memory 140, an input/output controller hub (ICH) 150, a non-volatile memory, or system flash, 160, a mass storage device 170,” and other components. SAMSUNG-1006, 4:38-56.

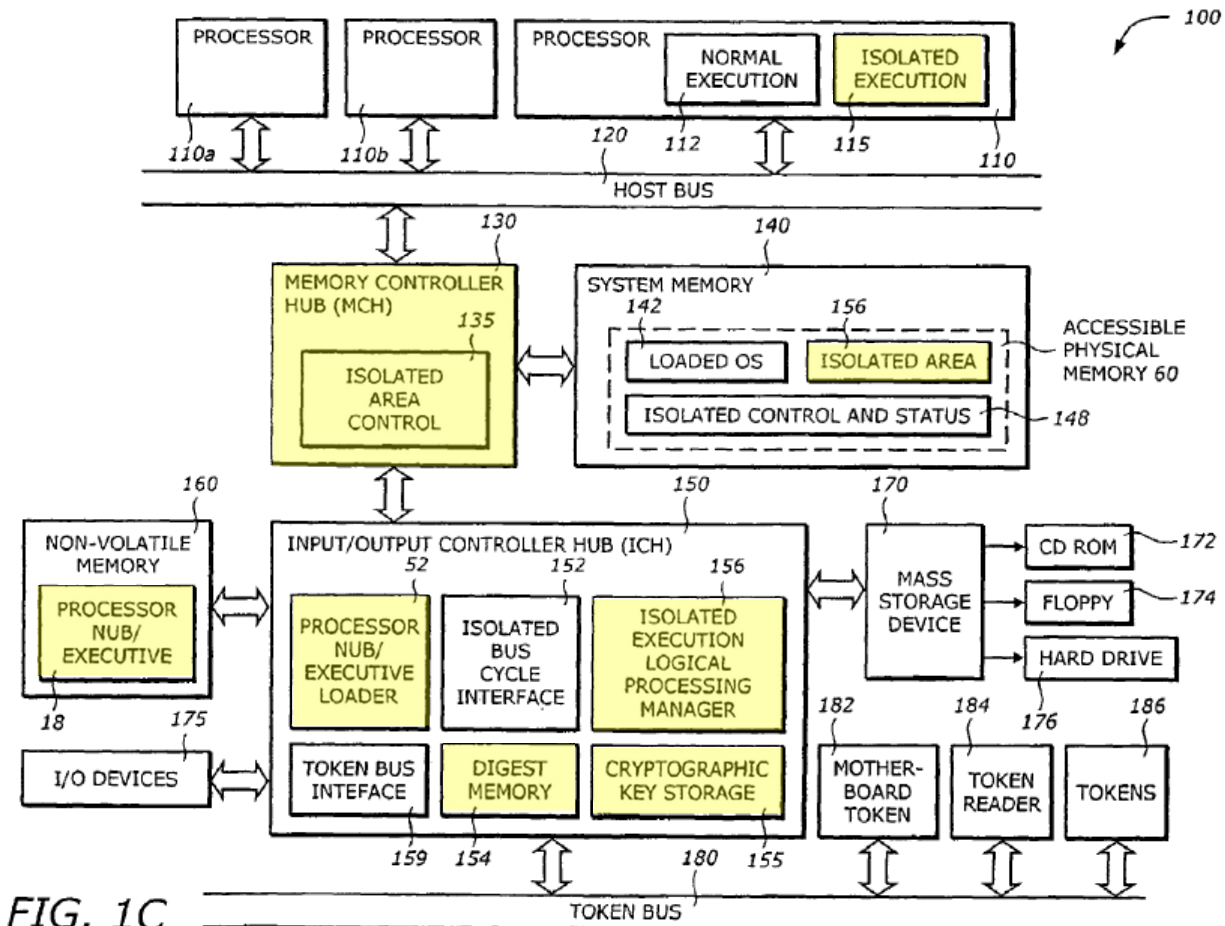


FIG. 1C

SAMSUNG-1006 (Ellison), FIG. 1C

51. “The processor 110 includes a normal execution mode 112 and an isolated execution circuit 115.” SAMSUNG-1006, 4:63-65. “The isolated execution circuit 115 provides a mechanism to allow the processor 110 to operate in an

isolated execution mode” and “provides hardware and software support for the isolated execution mode.” SAMSUNG-1006, 5:1-5. “[T]he MCH 130 has memory range registers (e.g., base and length registers) to represent the isolated area in the system memory 140. Once configured, the MCH 130 aborts any access to the isolated area that does not have the isolated access bus mode asserted.”

SAMSUNG-1006, 5:62-67. “Access to the isolated area 70 [shown in FIG. 1B] is restricted and is enforced by the processor 110 and/or the MCH 130.”

SAMSUNG-1006, 6:15-17.

52. The computer system 100 also includes a processor nub loader 52 (part of the I/C Controller Hub (ICH) 150) that “copies the processor nub 18 from the system flash memory (e.g., the processor nub code 18 in non-volatile memory 160) into the isolated area 70, verifies and logs its integrity, and manages a symmetric key used to protect the processor nub's secrets.” SAMSUNG-1006, 6:52-56. “For security purposes, the processor nub loader 52 is unchanging, tamper-resistant and non-substitutable.” SAMSUNG-1006, 6:57-61.

53. The ICH 150 “represents a known single point in the system having the isolated execution functionality” and includes a digest memory 154 and a cryptographic key storage 155. SAMSUNG-1006, 6:27-67. “The digest memory 154, typically implemented in RAM, stores the digest (e.g., hash) values of the loaded processor nub 18, the operating system nub 16, and any other critical

modules (e.g., ring-0 modules) loaded into the isolated execution space. The cryptographic key storage 155 holds a symmetric encryption/decryption key that is unique for the platform of the system 100.” SAMSUNG-1006, 6:60-67.

54. A non-volatile memory 160 is coupled to the ICH 150 and includes the processor nub 18 (*see* FIGS. 1A, 1B, 1C). SAMSUNG-1006, 7:19-22. “The processor nub 18 provides the initial set-up and low-level management of the isolated area 70 (in the system memory 140), including verification, loading, and logging of the operating system nub 16, and the management of the symmetric key used to protect the operating system nub's secrets.” SAMSUNG-1006, 7:22-27.

55. Ellison’s FIG. 2 (reproduced below) illustrates a secure platform 200 for performing isolated execution. The secure platform 200 includes “the OS nub 16, the processor nub 18, a key generator 240, a hashing function 220, and a usage protector 250, all operating within the isolated execution environment, as well as a software environment 210” that may exist inside the isolated execution environment. SAMSUNG-1006, 8:12-32. Unauthorized attempts at accessing the software environment (including the usage protector 250) are blocked and terminated. SAMSUNG-1006, 13:4-14:5, FIGS. 4-7.

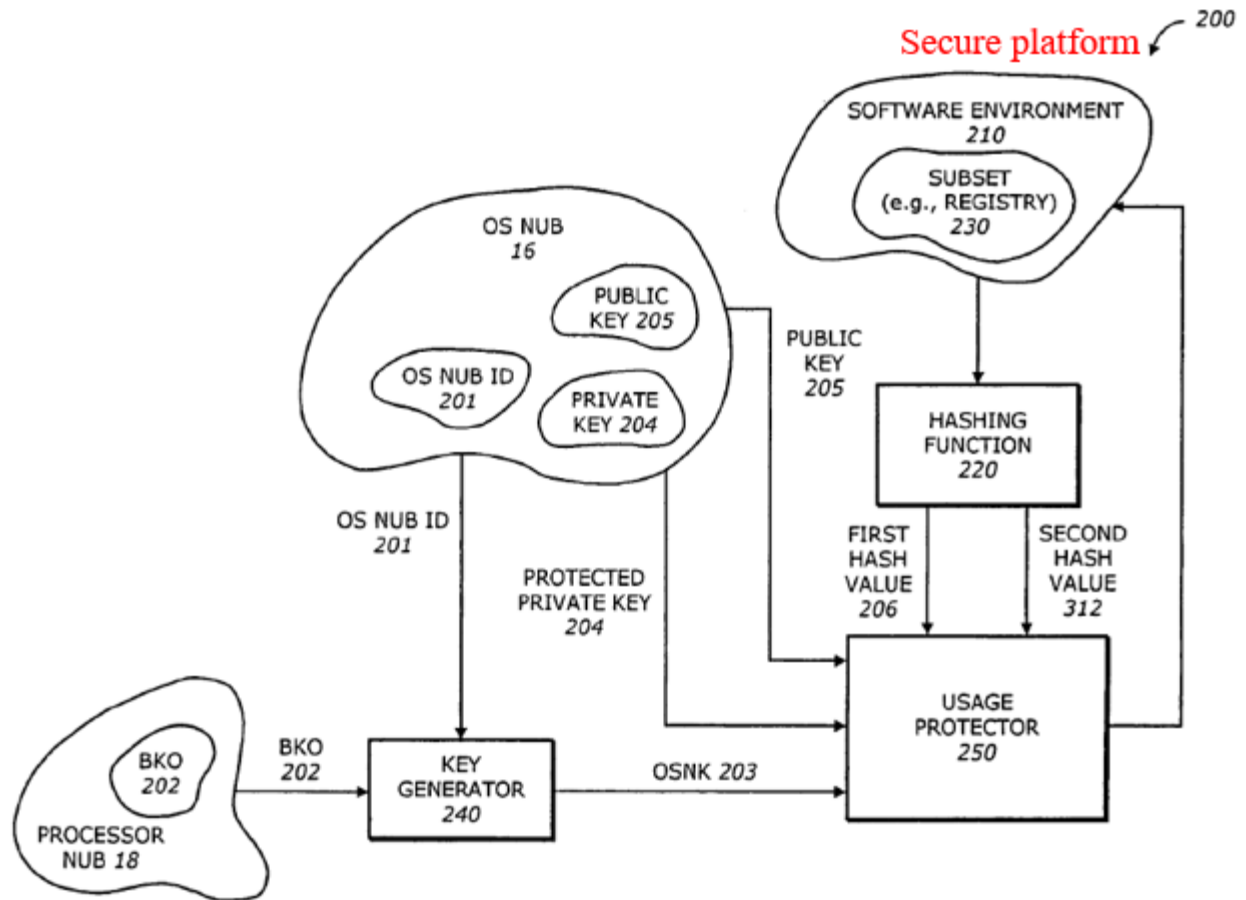


FIG. 2

SAMSUNG-1006 (Ellison), FIG. 2

56. “The OS nub or OS executive (OSE) 16 is part of the operating system running on the secure platform 200” and includes a protected private key 204 and a public key, as shown in FIG. 2. SAMSUNG-1006, 8:33-65. “The protected private key 204 may be programmed into the fuses of a cryptographic key storage 155 of the input/output control hub (ICH) 150 or elsewhere in persistent storage within the platform 200.” SAMSUNG-1006, 8:44-47. “Only the OS nub 16 can retrieve and decrypt the encrypted private key for subsequent use. In the digital

signature generation process, the protected private key 204 is used as an encryption key to encrypt a digest of a message producing a signature, and the public key 205 is used as a decryption key to decrypt the signature, revealing the digest value.”

SAMSUNG-1006, 8:57-65.

57. A “key generator 240 generates a key operating system nub key (OSNK) 203” “by combining [a master binding key] BK0 202 and [an] OS Nub ID 201 using a cryptographic hash function.” SAMSUNG-1006, 8:66-9:40. “The OS nub 16 may supply the OSNK 203 to *trusted agents, such as the usage protector 250.*” *Id.*

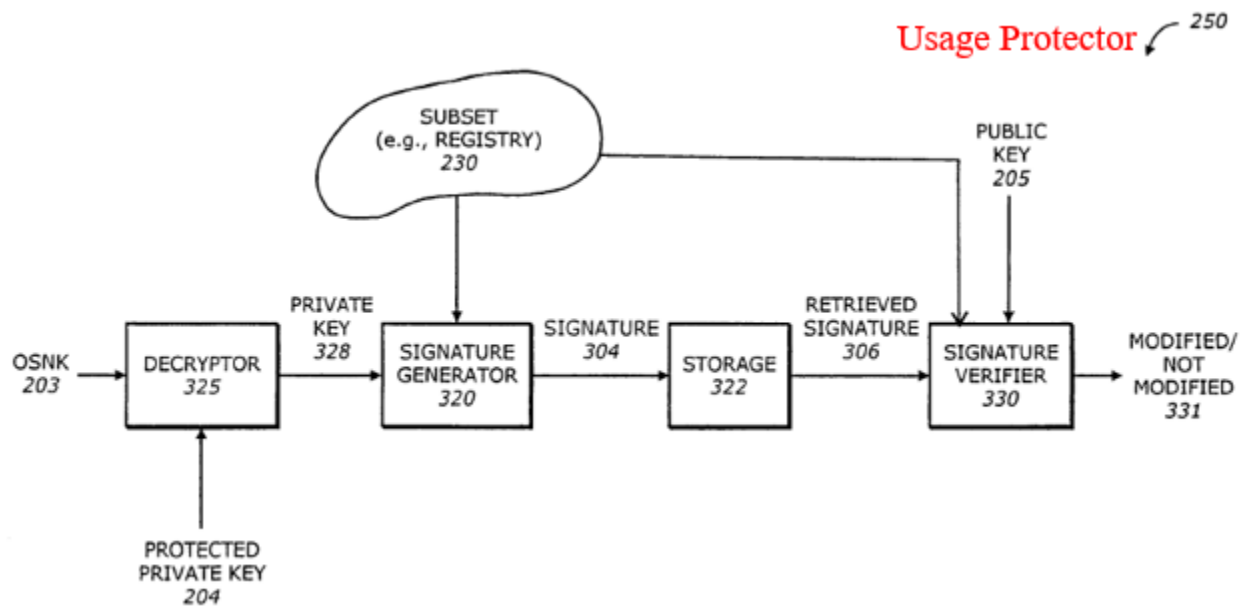
58. “The usage of the software environment 210 or the usage of [a] subset 230 [(e.g., a registry)] is protected by the usage protector 250,” which “uses the OSNK 203 to protect the usage of the subset 230.” SAMSUNG-1006, 9:28-35. “The usage protector 250 is coupled to the key generator 240 to protect usage of the software environment 210 or the subset 230, using the OSNK 203. The usage protection includes protection against unauthorized reads, and detection of intrusion, tampering or unauthorized modification.” SAMSUNG-1006, 9:48-52. “The[re] are several different embodiments of the usage protector 250. In one embodiment, the usage protector 250 decrypts the subset using the OSNK 203.” SAMSUNG-1006, 9:63-65. In “other embodiments, the usage protector uses the

OSNK 203, the protected private key 204 and the public key 205.” SAMSUNG-1006, 10:1-3.

59. FIG. 3C illustrates one embodiment of the usage protector 250.

SAMSUNG-1006, 10:63-67. “The usage protector 250 includes a decryptor 325, a signature generator 320, a storage medium 322, and a signature verifier 330.”

SAMSUNG-1006, 10:63-67.



SAMSUNG-1006 (Ellison), FIG. 3C

60. “The decryptor 325 accepts the OS nub's encrypted private key (i.e., protected) 204, and decrypts it using the OSNK 203, exposing the private key 328 for use in the isolated environment. The signature generator 320 generates a signature 304 for the subset 230 using the private key 328.” SAMSUNG-1006, 11:1-30. “The signature algorithm used by the signature generator 320 may be

public-key digital signature algorithm which makes use of a secret private key to generate the signature, and a public key to verify the signature.” SAMSUNG-1006, 11:1-30.

B. Muir

61. Muir is directed to providing data security in computer systems and electronic devices. SAMSUNG-1008, pg. 1-¶1. In particular, as show in Muir’s FIG. 5 (reproduced below), Muir discloses an example computer system and network in which a transaction, e.g., a financial transaction for e-commerce, can be performed between different electronic devices. SAMSUNG-1008, pp. 9-¶2, 11-¶3, 1-¶2.

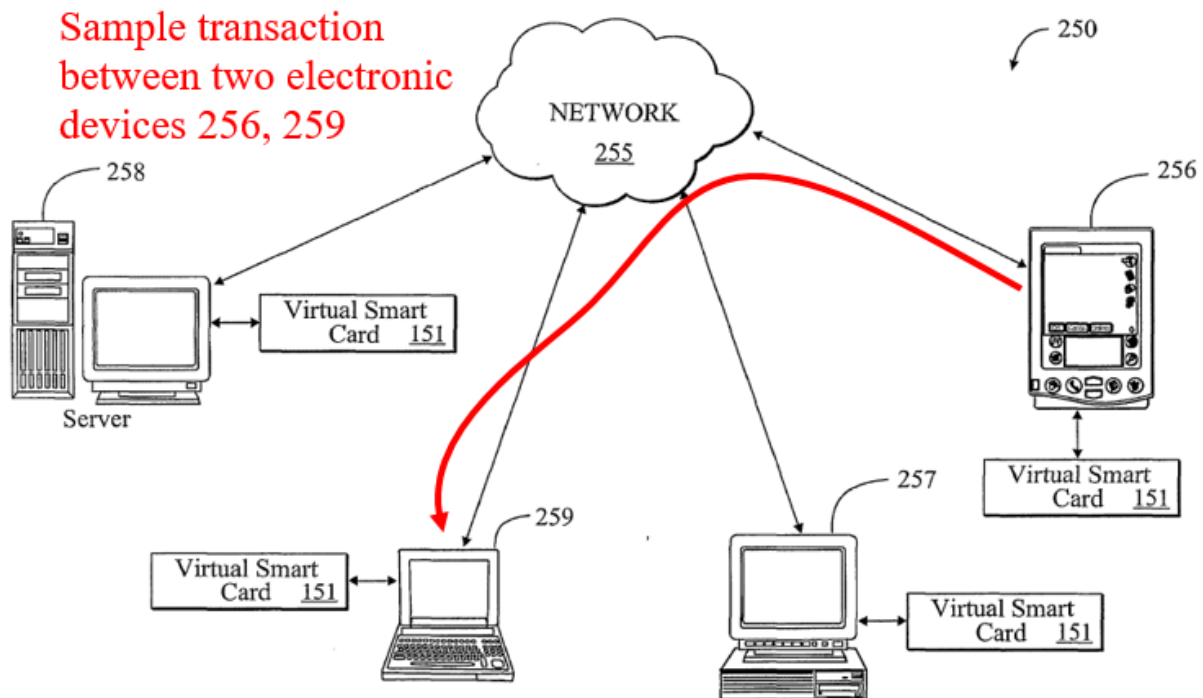


Fig. 5

SAMSUNG-1008 (Muir), FIG. 5

62. Communication medium 255 connects computing devices (e.g., server 258, laptop computer 259, personal computer 257, [] personal data assistant 256, cellular phone) with virtual smart cards 151. SAMSUNG-1008, pp. 11-¶3, 8-¶1. “[A] user of personal data assistant 256 may send a user of laptop computer 259 a digital signature using private key 167 stored in virtual smart card 151 of personal data assistant 256. Similarly, other devices may use the virtual smart card 151 for similar functions.” SAMSUNG-1008, pg. 12-¶1.

63. The *virtual* smart card 151 in the computing devices “provides software code, which upon execution, emulates a physical smart card for device operating system 150.” SAMSUNG-1008, pg. 7-¶4. The *virtual* smart card 151 includes a “restricted memory space 170 used for storing private keys 167. Restricted memory space 170 and private key 167 are normally inaccessible to operating system 125.” SAMSUNG-1008, FIG. 3, pg. 7-¶4. The private key 167 is stored in a restricted memory space 170 to avoid the private key 167 from being exposed to the operating system 125 or an unauthorized user, which was a problem in earlier systems. SAMSUNG-1008, pp. 3-¶3, 7-¶1. The private key 167 may be used to generate a digital signature and sent to another device. SAMSUNG-1008, pp. 9-¶5-10-¶2, 11-¶1.

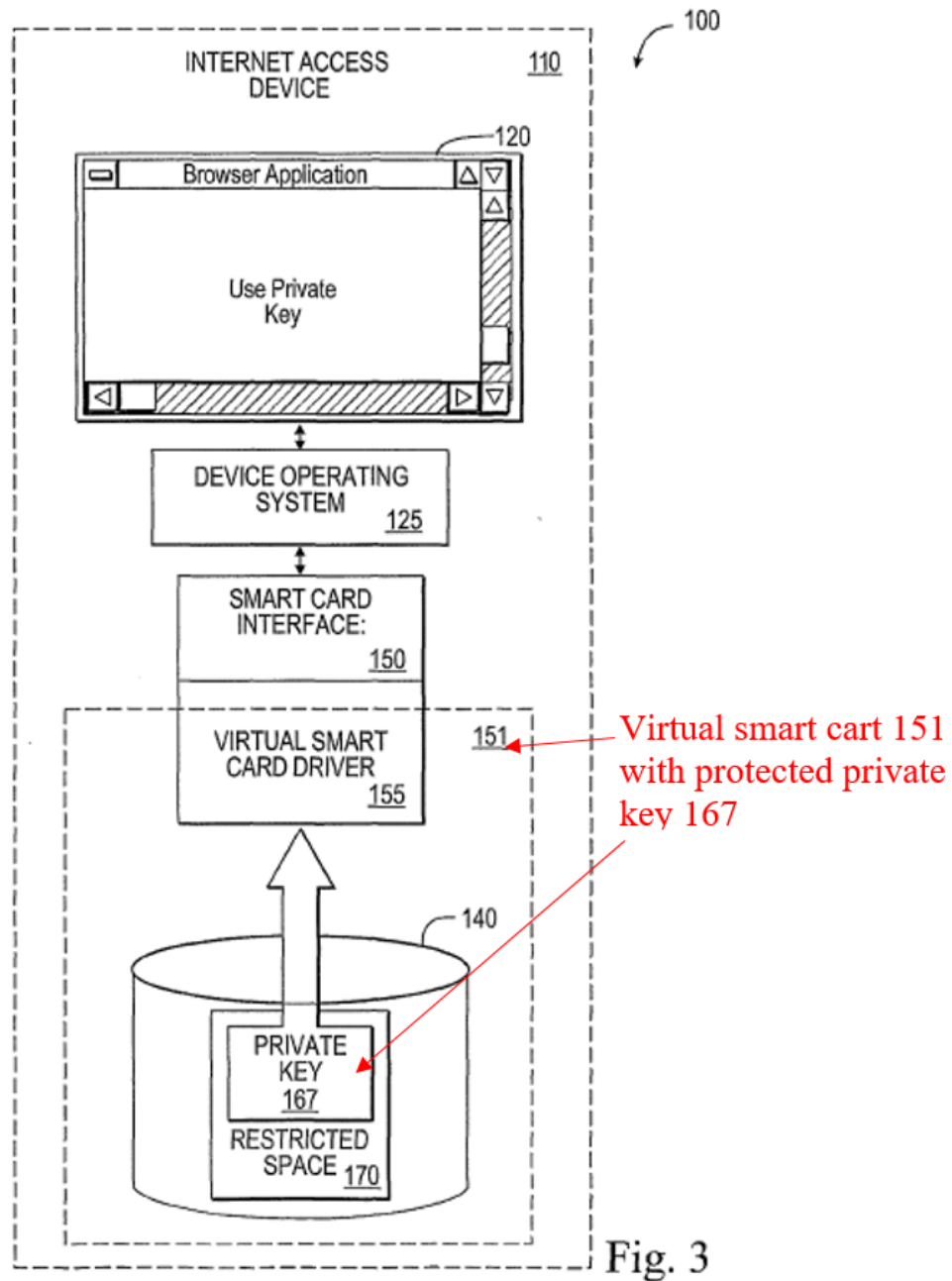


Fig. 3

SAMSUNG-1008 (Muir), FIG. 3

C. Combination of Ellison and Muir

64. Ellison discloses an isolated execution architecture that improves security in a computer system or platform by implementing several levels of hierarchy and normal and isolation execution modes for the operating system and processor.

SAMSUNG-1006, 1:12-15, 2:40-61, Abstract, FIG. 1. Ellison's isolated execution architecture can be applied to systems involving "[e]lectronic commerce (E-commerce) and business-to-business (B2B) transactions," which were previously "[v]ulnerable to unscrupulous attacks" such as "intrusion, security breach, and tampering." SAMSUNG-1006, 1:17-30. Earlier systems and work environments, such as systems involving security co-processors or smart cards using cryptographic or other security techniques, had limitations and drawbacks "in speed performance, memory capacity, and flexibility." SAMSUNG-1006, 1:38-51. Because Ellison's isolated execution architecture provides a solution to such limitations and drawbacks, a POSITA would have applied Ellison's isolated execution architecture to such systems, e.g., systems involved in e-commerce or B2B transactions or smart cards using cryptography. However, Ellison does not reveal details of such a system or transaction between multiple parties. Muir does.

65. As I noted above in Section VI.B, Muir discloses a computer network with multiple devices involved in electronic transactions, e.g., e-commerce.

SAMSUNG-1008, pp. 9-¶2, 11-¶3, 1-¶2. In Muir's computer network, devices

(e.g., server 258, laptop computer 259, personal computer 257, PDA 256, or cellular phone) are configured to perform secure transactions. SAMSUNG-1008, pp. 11-¶3, 8-¶1. Muir's virtual smart cards 151 include a restricted memory space 170 to store a private key 167 and for performing operations such as encryption of data or generation of a digital signature using the private key. SAMSUNG-1008, pp. 9-¶3, 2-¶1, 3-¶¶1-2, 6-¶3, 10-¶¶1-2, 12-¶1, 18-¶6, FIG. 3.

66. A POSITA looking to implement Ellison's isolated execution architecture in a computer system with multiple devices would have looked to Muir's teachings to implement a secure electronic transaction between two parties. The combination would have been obvious, predictable, and a POSITA would have had a reasonable expectation of success in implementing the combination because both Ellison and Muir are related to providing improved security for transactions involving keys using cryptographic techniques. SAMSUNG-1006, 1:38-51; SAMSUNG-1008, pp. 11-¶3, 8-¶1, 7-¶4.

67. In the combination of Ellison and Muir (hereinafter "**EMC**"), each device involved in a transaction would have been connected through a computer network such as Muir's network shown in FIG. 5. The individual devices would have included Ellison's isolated execution architecture and one or more parts of Muir's system (e.g., a restricted memory space for storing the private key and performing cryptographic operations such as encryption). When utilized in a transaction

involving smart cards, such a combination would also have improved speed performance, flexibility, and reduced costs by eliminating the need for additional hardware such as physical smart cards coupled to the computing devices and use Muir's *virtual* smart card instead. SAMSUNG-1006, 1:38-51; SAMSUNG-1008, pp. 4-¶¶3-4, 6-¶5. Using Muir's virtual smart card is also beneficial because a user can avoid problems in earlier computer systems that did not have a standard interface for physical smart cards and would have problems using a smart card it was not configured to interface with. SAMSUNG-1008, pg. 4-¶¶3-4. Moreover, by using virtual smart cards, a user would not have to carry hardware (e.g., a physical smart card) with the user when on the move and the user would also have benefitted from reduced costs by not having to buy a physical smart card. SAMSUNG-1008, pg. 6-¶5.

68. To complete the transaction between the two devices, a digital signature generated using a private key associated with one (transmitting) party and optionally along with encrypted data would have been sent to the other (receiving) party in a transaction to securely transmit data and allow the receiving party to decrypt the data and verify the identity of the transmitting party. SAMSUNG-1008, pp. 9-¶3, 2-¶1, 3-¶¶1-2, 6-¶3, 10-¶¶1-2, 12-¶1, 18-¶6.

69. A POSITA would have viewed extension of Ellison's system to include the multi-party transactions described by Muir to have been a natural and obvious

extension. Indeed, Muir itself describes the benefits of secure multi-party transactions (*see, e.g.*, SAMSUNG-1008, pp. 4-¶¶3-4, 6-¶5) and a POSITA would have been motivated to enable such multi-party transactions (e.g., e-commerce transactions) in Ellison's system to increase the functionality and usefulness of Ellison's system. In extending Ellison's functionality to include secure multi-party transactions, as described in Muir, a POSITA would have retained the benefits of Ellison's security functionality (e.g., the isolated execution environment and key-based encryption) because such functionality remains valuable to secure operation of Ellison's system and complements the security of the multi-party transactions described by Muir.

70. In fact, Ellison itself alludes to use of its system in e-commerce and B2B transactions. *See* SAMSUNG-1006, 1:17-30. From this description, a POSITA would have envisioned use of Ellison's system in multi-party e-commerce or business transactions, although Ellison does not explicitly describe details of multi-party transactions. Given the lack of details in Ellison, a POSITA would have been motivated to look to other references that describe e-commerce and business transactions, such as Muir, and would have found it obvious to apply the details of those transactions (e.g., Muir's multi-party transactions) to Ellison. Indeed, a POSITA would have seen addition of multi-party transactions to Ellison's system

as use of a known technique (e.g., as disclosed by Muir) to improve similar devices (the Ellison and Muir secure processing systems) in the same way.

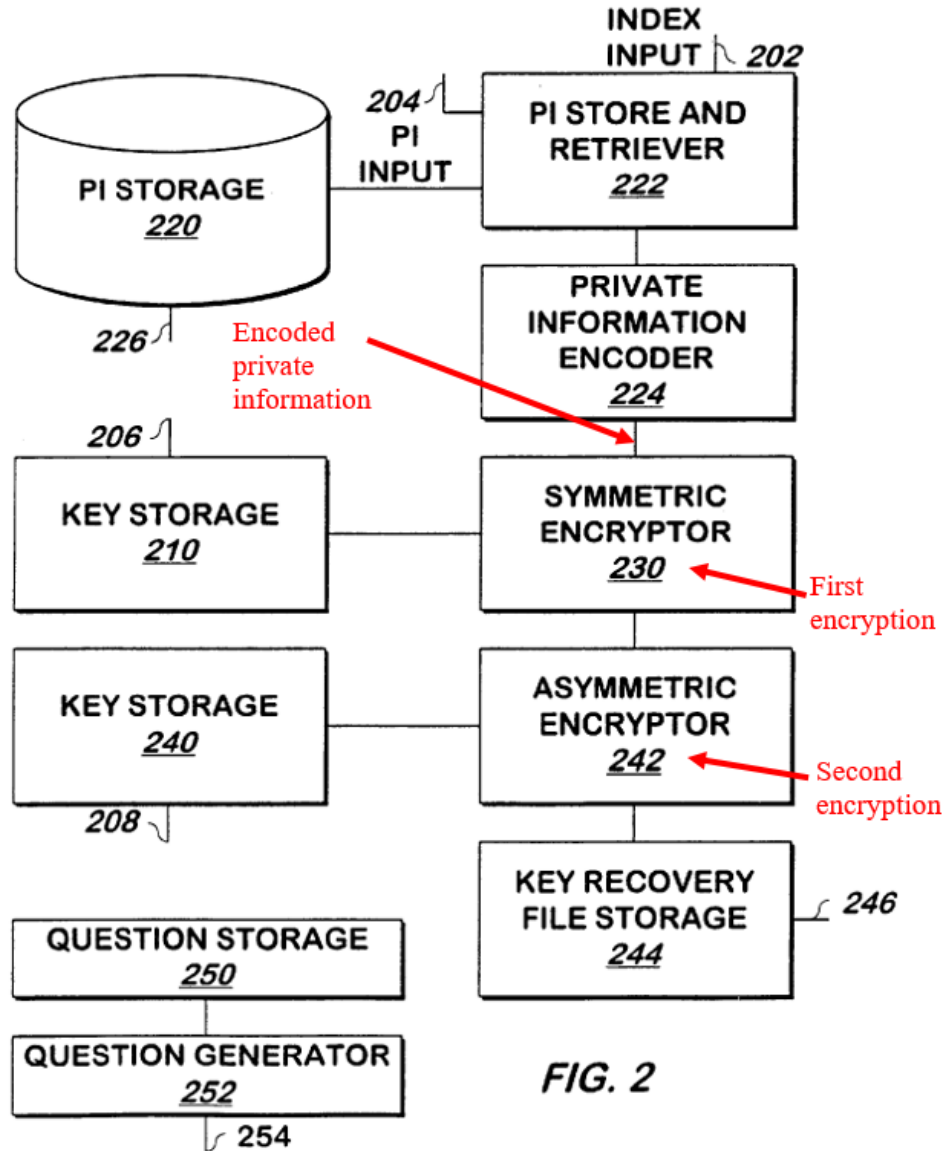
71. For at least the foregoing reasons, a POSITA would have combined the teachings of Ellison and Muir and would have had a reasonable expectation of success. Indeed, the combination would have simply involved combining prior art elements (Muir's computer system with restricted memory space and secure cryptographic operations) according to known methods (Ellison's isolated execution architecture) to yield the predictable result of performing electronic transactions securely between two parties.

D. Al-Salqan

72. Al-Salqan explains that while private keys can be used in asymmetric encryption techniques, there can be practical limitations that make it difficult to use them. SAMSUNG-1007, 1:21-2:47. For example, a principal (owner of the private key) may "lose or forget his or her private key or private key password," and a principal may be the only one in an organization to know the private key and may leave the organization without anyone having knowledge of the private key. SAMSUNG-1007, 1:55-60, 2:25-41. In addition, an intruder may breach security and steal the private key. *Id.*

73. To address these concerns, Al-Salqan discloses a method to encrypt and store a key or key password in a secure manner that would allow the principal or

members of the principal's organization to recover the key in the scenarios noted above. SAMSUNG-1007, 2:42-63. Al-Salqan's method involves encoding "[p]rivate information of the principal such as mother's maiden name and social security number ... for example by hashing. The result of this encoding is used to symmetrically encrypt the private key or key password. The encrypted private key or key password is again encrypted, for example asymmetrically using the public key of a trusted party such as a certification authority as the encryption key." SAMSUNG-1007, 2:50-60, 4:4-5:9, FIG. 2 (reproduced below).



SAMSUNG-1007 (Al-Salqan), FIG. 2

74. As shown in FIG. 2, a key based on the principal's private information is encrypted twice, first symmetrically and second asymmetrically.

75. To decrypt such keys, Al-Salqan discloses a decryption method that uses an asymmetric decryptor 314 and symmetric decryptor 330 to remove the two layers of encryption. In particular, referring to Al-Salqan's FIG. 3 (reproduced below),

“a key that will decrypt the encryption performed by the asymmetric encryptor 242 of FIG. 2 is supplied at input 304 and stored in key storage 312.” SAMSUNG-1007, 5:15-25. A “key recovery file is received at input 306 and stored in key recovery file storage 310.” *Id.* “Asymmetric decryptor 314 receives the key recovery file from key recovery file storage 310 and receives the certificate authority's private key from key storage 312. Asymmetric decryptor 314 decrypts the key recovery file using the ... key stored in key storage 312 as the key.” SAMSUNG-1007, 5:25-35. “Symmetric decryptor 330 decrypts the key recovery file decrypted by [a]symmetric decryptor 314 using the encoded private information received from private information encoder 324 as the decryption key.” SAMSUNG-1007, 5:25-35, 4:15-5:24.

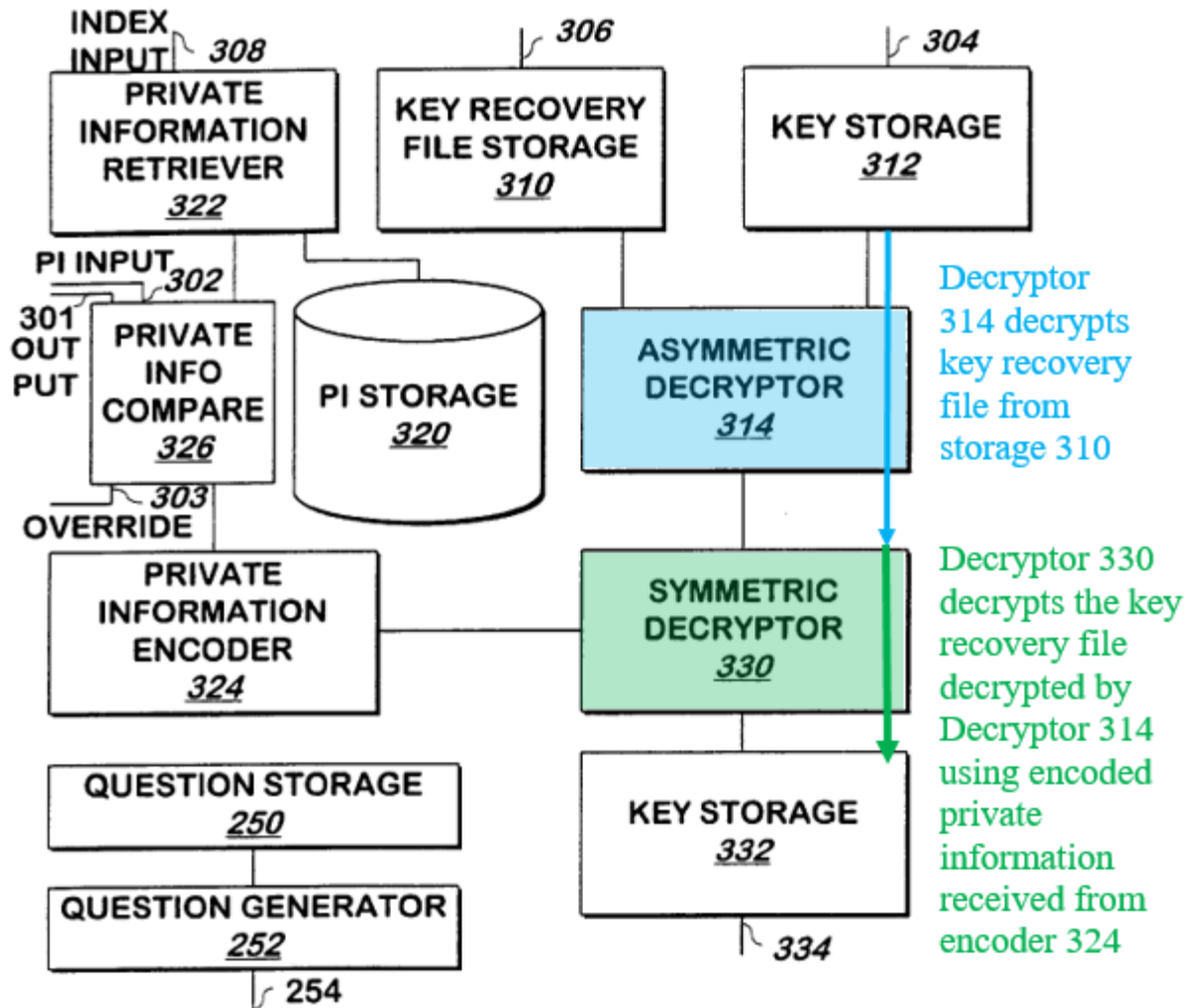


FIG. 3

SAMSUNG-1007 (Al-Salqan), FIG. 3

E. Combination of Ellison, Muir, and Al-Salqan

76. It would have been obvious for a POSITA to combine the teachings of EMC with the teachings of Al-Salqan because doing so would merely involve combining prior art elements according to known methods to yield predictable results, as I explain in more detail below.

77. Ellison teaches using an encrypted private key 204 but does not provide details of how the private key 204 is encrypted. SAMSUNG-1006, 8:33-65, 11:1-12:26, FIGS. 3C, 3D, 6:60-67, *see supra* [1b]. It would have been obvious to a POSITA that encrypting a private key 204 involves a key to encrypt the private key 204. SAMSUNG-1021, 1:13-2:7 (describing the well-known use of a key for encryption in different types of cryptography and the strength of encryption being dependent, in part, on the length of the key). In this regard, and given the lack of details in Ellison, a POSITA would have been motivated to look to other references that describe key encryption, such as Al-Salqan, and would have found it obvious to apply the details of that key encryption to implement the key encryption in Ellison. Indeed, a POSITA would have seen Al-Salqan's key encryption as use of a known technique to improve similar devices (the EMC and Al-Salqan processing systems) in the same way.

78. A POSITA interested in using an encrypted private key, such as Ellison's private key 204, also would have turned to Al-Salqan's method for generating the encrypted private key to avoid the practical limitations of using private keys such as a user/principal forgetting the private key or private key password or leaving a place of employment, as explained above. SAMSUNG-1007, 1:55-60, 2:25-41. A POSITA would have incorporated Al-Salqan's method of generating an encrypted private key using private information "such as [the principal's] social security

number, mother's maiden name, and other similar information.” SAMSUNG-1007, 4:4-7, Abstract, 2:49-63. Furthermore, by using two encryption layers—a first symmetrical encryption layer and a second asymmetric encryption layer—the private key generated based on Ellison and Al-Salqan’s teachings would have additional layers of protection.

79. To combine the teachings of Muir, Ellison, and Al-Salqan, EMC’s system would have been modified to be able to obtain principal input, as described by Al-Salqan (e.g., by prompting the principal with questions requesting the principal’s private information) and to include the symmetric encryption 230 and asymmetric encryptor 242 for adding two encryption layers to protect the private key. In order to decrypt the private key, EMC’s decryptor 325 would have been modified to include Al-Salqan’s symmetric decryptor 330 and asymmetric decryptor 314 so that it can decrypt the two layers of encryption (as disclosed by Al-Salqan’s FIG. 3) and yield the private key 328 for further use by the usage protector 250.

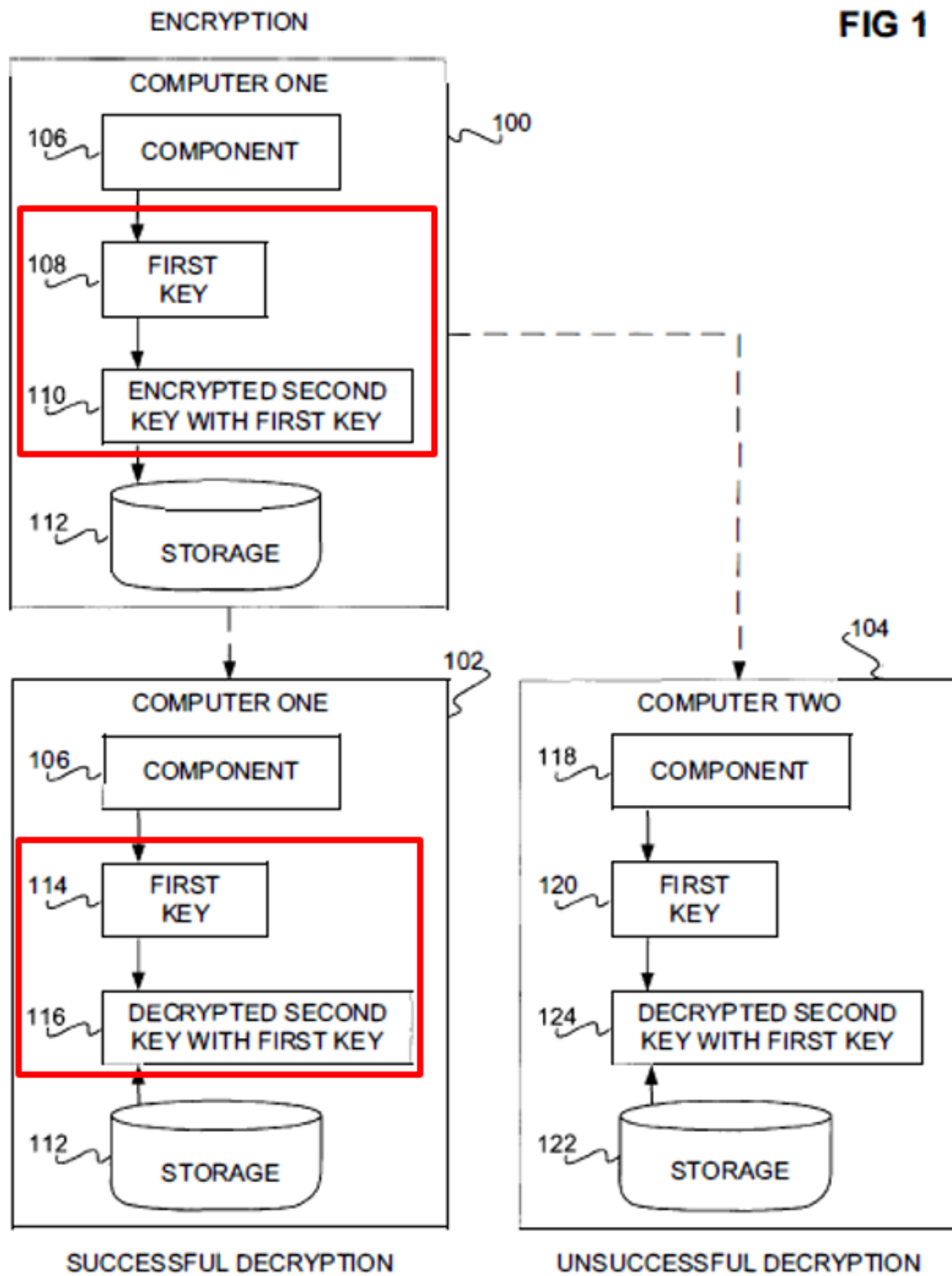
80. Because encrypting data using symmetric or asymmetric techniques was well-known and implementing user interfaces to query users for information was well-known, such a combination would have been predictable and a POSITA would have had a reasonable expectation of success in implementing the combination. Moreover, because only a finite number of encryption options exist and Ellison does not describe a specific encryption method, a POSITA would have

investigated and found obvious the use of a known option, such as Al-Salqan's. A POSITA would have been motivated to choose Al-Salqan's method to achieve the above-noted benefits that Al-Salqan explicitly mentions for using multiple layers of encryption using private information. Indeed, the practical limitations of using private keys, as explained in Al-Salqan and noted above, were commonly experienced and would have protected the EMC system from similar deficiencies, thereby making the combination of Ellison and Al-Salqan foreseeable.

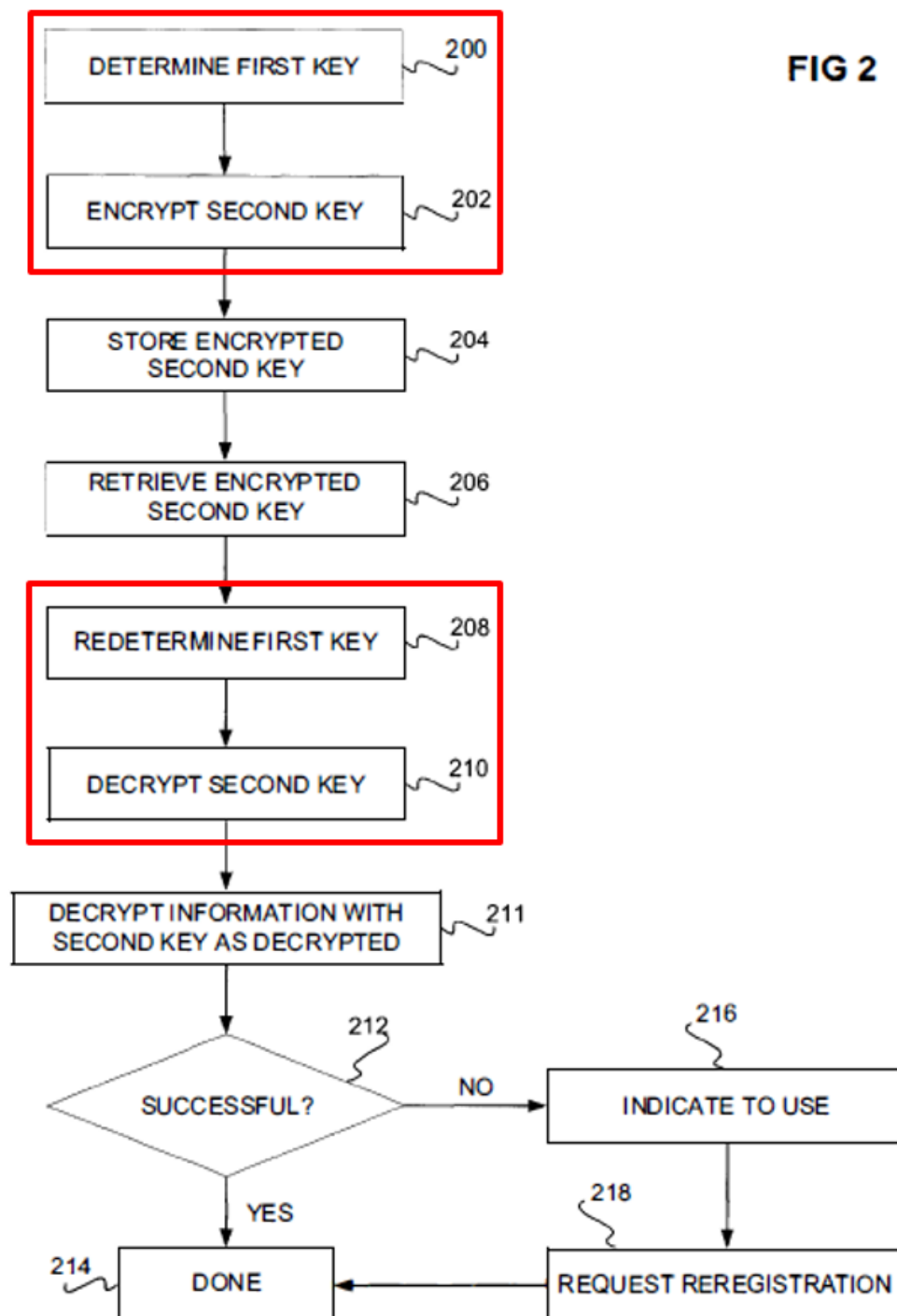
F. Searle

81. Searle discloses encrypting “a key [(second key)] using another key [(first key)] that is unique and particular to a given client, such as a desktop computer, a laptop computer, a portable electronic device, etc., for fraud prevention and other purposes.” SAMSUNG-1009, 2:14-18. An identifier of a computer component can be used as the first key. SAMSUNG-1009, 4:48-65. The component can be a processor “that has a unique serial number or other identifier,” or a network card “having a unique media access controller (MAC) address.” SAMSUNG-1009, 4:48-65. The first key can be “one or more of: a processor identifier, a network card address, ... serial numbers and/or the number of cylinders of attached hard disk drives, checksums of the read-only memory (ROM) or other system components, the Internet Protocol (IP) address of the computer or system, and a combination of installed cards.” SAMSUNG-1009, 2:13-33, 4:58-5:7.

82. The second key is encrypted using the first key. SAMSUNG-1009, 2:30-33, Abstract, 5:8-28, FIGS. 1, 2 (reproduced below). When decryption is desired, the second key is decrypted using the first key. SAMSUNG-1009, 2:33-40, FIG. 1, 2, 5:65-6:27. “Because the first key is specific to the underlying computer or device, if the encrypted second key is moved to another computer or device, it will not be decrypted successfully.” SAMSUNG-1009, 2:33-40.



SAMSUNG-1009 (Searle), FIG. 1



SAMSUNG-1009 (Searle), FIG. 2

G. Combination of Ellison, Muir, and Searle

83. Ellison teaches using an encrypted private key 204 but does not provide details of how the private key 204 is encrypted. SAMSUNG-1006, 8:33-65, 11:1-12:26, FIGS. 3C, 3D, 6:60-67, *see supra* [1b]. It would have been obvious to a POSITA that encrypting a private key 204 involves an encryption key to encrypt the private key 204. SAMSUNG-1021, 1:13-2:7 (describing the well-known use of a key for encryption in different types of cryptography and the strength of encryption being dependent, in part, on the length of the key). In this regard, and given the lack of details in Ellison, a POSITA would have been motivated to look to other references that describe key encryption, such as Searle, and would have found it obvious to apply the details of that key encryption to implement the key encryption in Ellison. Indeed, a POSITA would have seen Searle's key encryption as use of a known technique to improve similar devices (the EMC and Searle processing systems) in the same way.

84. A POSITA using Ellison's encrypted private key 204 would also have turned to Searle to perform encryption and decryption of the private key (Searle's second key) using a key associated with a hardware device serial number (Searle's first key) because doing so would have provided an enhanced level of security and protection from intruders. In particular, by encrypting the private key using a hardware device serial number associated with a user's electronic device, only a

device with the information on the hardware device serial number would be able to perform the decryption of the private key, which would be helpful in preventing hacking of the private key even if an intruder gets possession of the encrypted private key (and any accompanying public key). SAMSUNG-1010, Abstract, ¶[0023]. Such advantages were known in the art. For example, another reference, Shen explains that when a hardware serial number is used as part of the encryption/decryption technique (like in Searle), data to be protected cannot be encrypted/decrypted by other users whose user end device hav[e] different hardware serial numbers from that of the true user, even if they know the true user's public key and private key.” SAMSUNG-1010, ¶[0019].

85. To incorporate the teachings of Searle in the EMC system, a POSITA would have used or associated a hardware device serial number with the encryption key or decryption key (Searle's first key) when performing encryption/decryption of the private key (Searle's second key). Searle explains that various suitable schemes can be used to perform encryption and decryption. SAMSUNG-1009, 3:66-4:11. For instance, as noted above and described in Searle, when encrypting EMC's private key, a suitable encryption scheme would have been used to configure an encryption key that is associated with a hardware device serial number and subsequently encrypt the private key. When decrypting EMC's encrypted private key, a suitable decryption scheme would have been used to

obtain a decryption key that is associated with a hardware device serial number and subsequently decrypt the encrypted private key.

86. Combining the teachings of EMC with the teachings of Searle in this manner would have been obvious to a POSITA because doing so would merely involve combining prior art elements (e.g., known hardware device serial number encryption) according to known methods (e.g., suitable known encryption/decryption methods) to yield predictable results (private key encrypted or decrypted using a key associated with a hardware device serial number). And, because encrypting a private key or decrypting an encrypted private key using a “hardware identifier of [a] computer system,” such as a hardware serial number, was known to a POSITA, such a combination would have been predictable and a POSITA would have had a reasonable expectation of success in implementing the combination. SAMSUNG-1011, ¶[0042]. Moreover, because only a finite number of encryption and decryption options existed and Ellison does not describe a specific encryption method, a POSITA would have investigated and found obvious the use of a known option, such as Searle's. A POSITA would have been motivated to choose Searle's method to achieve the above-noted benefits that Searle explicitly mentions for encrypting a private key or decrypting an encrypted private key using a hardware identifier of a computer system. The demand for

better computer security and fraud prevention would have further rendered the combination of Ellison, Muir, and Searle predictable and foreseeable.

VII. MANNER IN WHICH THE PRIOR ART REFERENCES RENDER THE '480 CLAIMS UNPATENTABLE

A. Claim 1

[1pre-1] A method for performing an electronic transaction between a first transaction party and a second transaction party using an electronic device operated by the first transaction party,

87. Claim 1 is rendered obvious by EMC.

88. To the extent the preamble is limiting, EMC renders [1pre-1] obvious. As I explained above in Sections VI.A-VI.C, EMC discloses a method to perform an electronic transaction (e.g., e-commerce) between two portable electronic devices associated with two transaction parties. SAMSUNG-1008, pp. 1-¶¶2-3, 11-¶3, 12-¶1, FIG. 5. EMC teaches that cryptographic systems and methods can be used to protect sensitive electronic data used in an e-commerce transaction (“***method for performing an electronic transaction***”). SAMSUNG-1008, pp. 9-¶2, 1-¶¶2-3, FIG. 5; SAMSUNG-1006, FIGS. 1A-1C, 2, 3C-3D, 3:9-4:56, 8:13-13:3. As shown below in Muir’s FIG. 5, the transaction can be conducted ***between the electronic devices of two transaction parties*** (e.g., first and second transaction parties). SAMSUNG-1008, pp. 11-¶3, 12-¶1. The electronic devices may include, for example, cellular phones, personal data assistants, and smart cards.

SAMSUNG-1008, pp. 11-¶3, 8-¶1, 1-¶¶2-3. Although two devices in FIG. 5 below have been labeled as the transaction parties, other devices in FIG. 5 may also correspond to the transaction parties if involved in a transaction with each other.

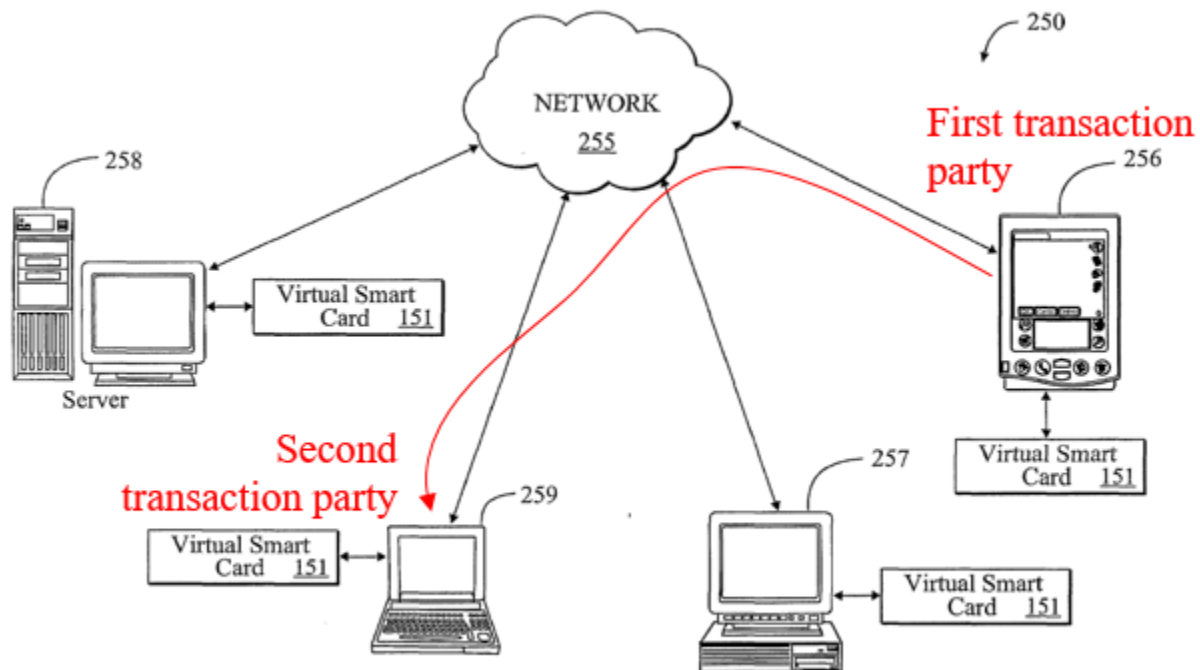


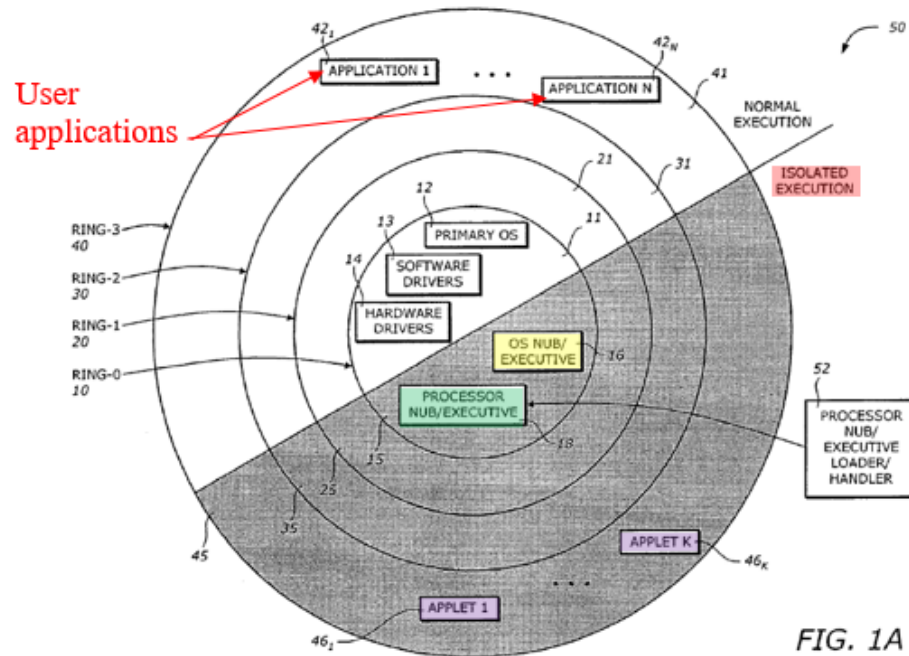
Fig. 5

SAMSUNG-1008 (Muir), FIG. 5

[1pre-2] the electronic device having an operating system creating a run-time environment for user applications, the method comprising:

89. As I noted above in Section VI.C, in EMC, Ellison’s “isolated execution architecture” would have been implemented in Muir’s electronic devices including the electronic device of the first transaction party. Thus, an architecture similar to

the one shown in Ellison's 615's FIGS. 1A-1C (reproduced below) would have been implemented in Muir's electronic device of the first transaction party.



SAMSUNG-1006 (Ellison), FIG. 1A

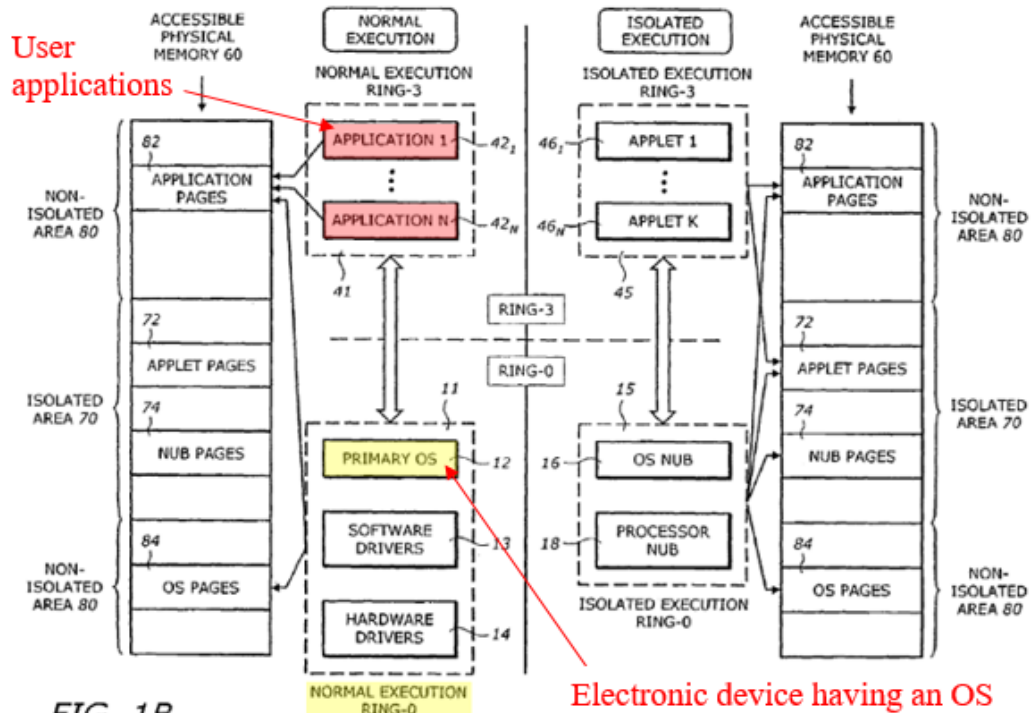


FIG. 1B

SAMSUNG-1006 (Ellison), FIG. 1B

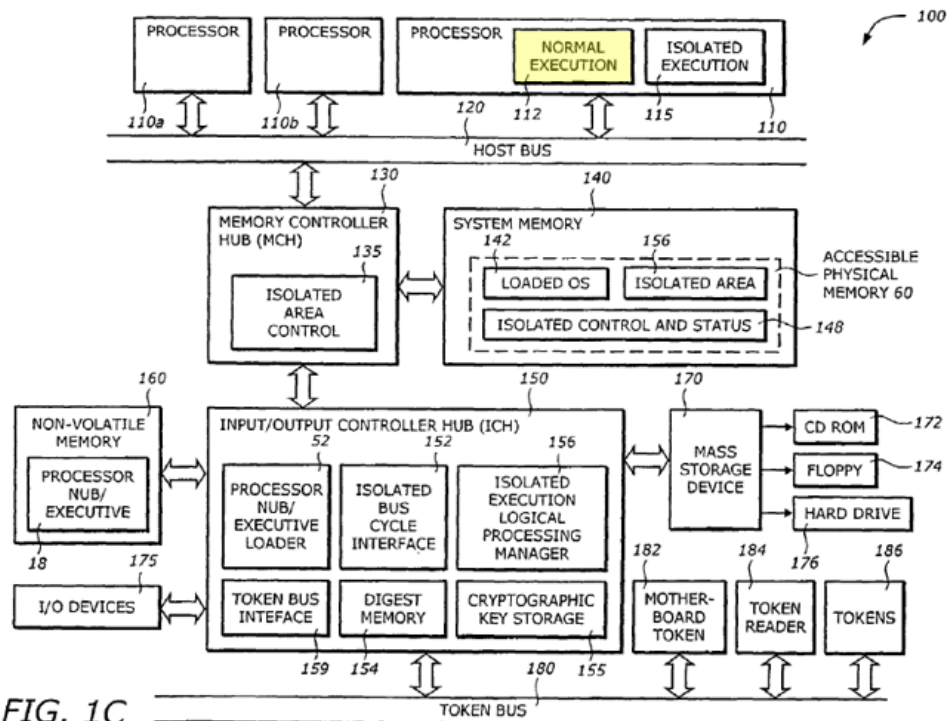


FIG. 1C

SAMSUNG-1006 (Ellison), FIG. 1C

90. FIGS. 1A and 1B depict *a primary operating system (OS) 12 that creates a run-time environment for user applications, such as applications 42₁-42_N* which can be executed in the normal execution mode and are isolated from components of the isolated execution rings. SAMSUNG-1006, 3:9-61, 4:1-29, 2:57-61.

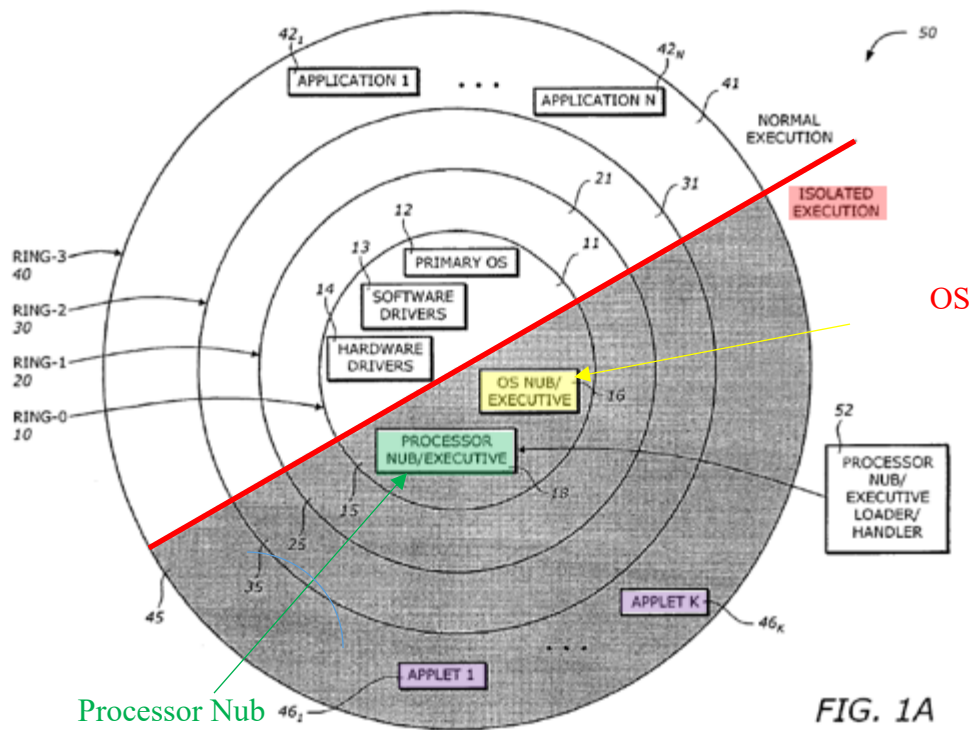
Applications 42₁-42_N are configured to operate in Ring-3, which “typically encompasses users or applications level and has the least privilege,” and would therefore be understood by a POSITA as encompassing user applications.

SAMSUNG-1006, 2:57-61. OS 12 operates in the normal execution mode 112 and may be executed by processor 110. SAMSUNG-1006, 4:23-29, 4:57-5:27, FIGS.

1A, 1C. Thus, in EMC, an electronic device associated with the first transaction party would have an OS 12 that creates a run-time environment for applications 42₁-42_N.

[1a] providing a private key in a memory of said electronic device, said memory being a secure part of a Basic In Out System or any other secure location in said electronic device, which private key is inaccessible to a user of said electronic device,

91. As I explained above in Section VI.A, Ellison’s isolation execution architecture includes an isolated execution region (*see* FIG. 1A below) in which entities such as OS nub 16 and processor nub 18 operate in a secure environment in an isolated execution mode. SAMSUNG-1006, 3:9-25, 2:41-61, FIG. 1B.



SAMSUNG-1006 (Ellison), FIG. 1A

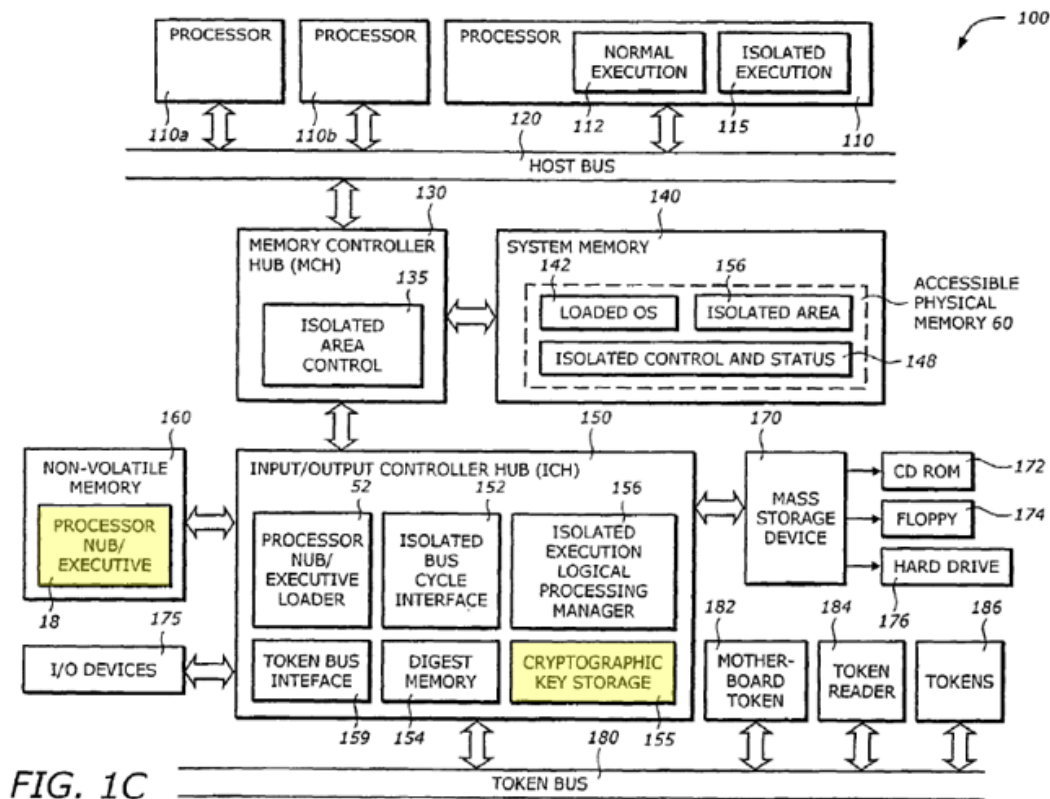
92. Ellison explains that the isolated execution region “is *accessible only to elements of the operating system and processor operating in isolated execution mode*” and that isolated mode applets 46₁ to 46_K and their data are tamper-resistant and monitor-resistant from all software attacks from other applets and OS 12.

SAMSUNG-1006, 4:1-22. *A user* is not an element of the operating system and processor operating in isolated execution mode, and therefore *cannot access elements within the isolated execution region*. The isolated execution region is protected by the processor and chipset in the computer system, and access to the isolated region is possible only through isolated read and write cycles which are

issued by a processor executing in an isolated execution mode. SAMSUNG-1006, 3:32-49. Effectively, the isolated execution mode and region are protected from user tampering and software outside of the isolated execution region.

93. “One task of [a] processor nub 18 is to verify and load the ring-0 OS nub 16 into the isolated area, and to generate the root of a key hierarchy unique to a combination of the platform, the processor nub 18, and the operating system nub 16.” SAMSUNG-1006, 3:46-55. “*The isolated area 70 is accessible only to elements of the operating system and processor operating in isolated execution mode.*” SAMSUNG-1006, 4:19-23. Thus, processors and OSs not operating in the isolated execution mode (e.g., OSs that can be interfered with by users) cannot interfere or access elements operating in the isolated execution mode such as OS nub 68 and processor nub 18.

94. The OS nub 16 has access to a *protected private key 204* that “can be stored in the multiple key storage ... *of the non-volatile flash memory 160*” and can be “programmed into the fuses of a *cryptographic key storage 155* of the ICH 150 or elsewhere in persistent storage within platform 200.” SAMSUNG-1006, 8:33-65, FIG. 1C (reproduced below).



SAMSUNG-1006 (Ellison), FIG. 1C

95. Platform 200 is depicted in FIG. 2 below and is described as a *secure platform* 200 that “includes the OS nub 16, the processor nub 18, a key generator 240, a hashing function 220, and a usage protector 250, *all operating within the isolated execution environment.*” SAMSUNG-1006, 8:25-32.

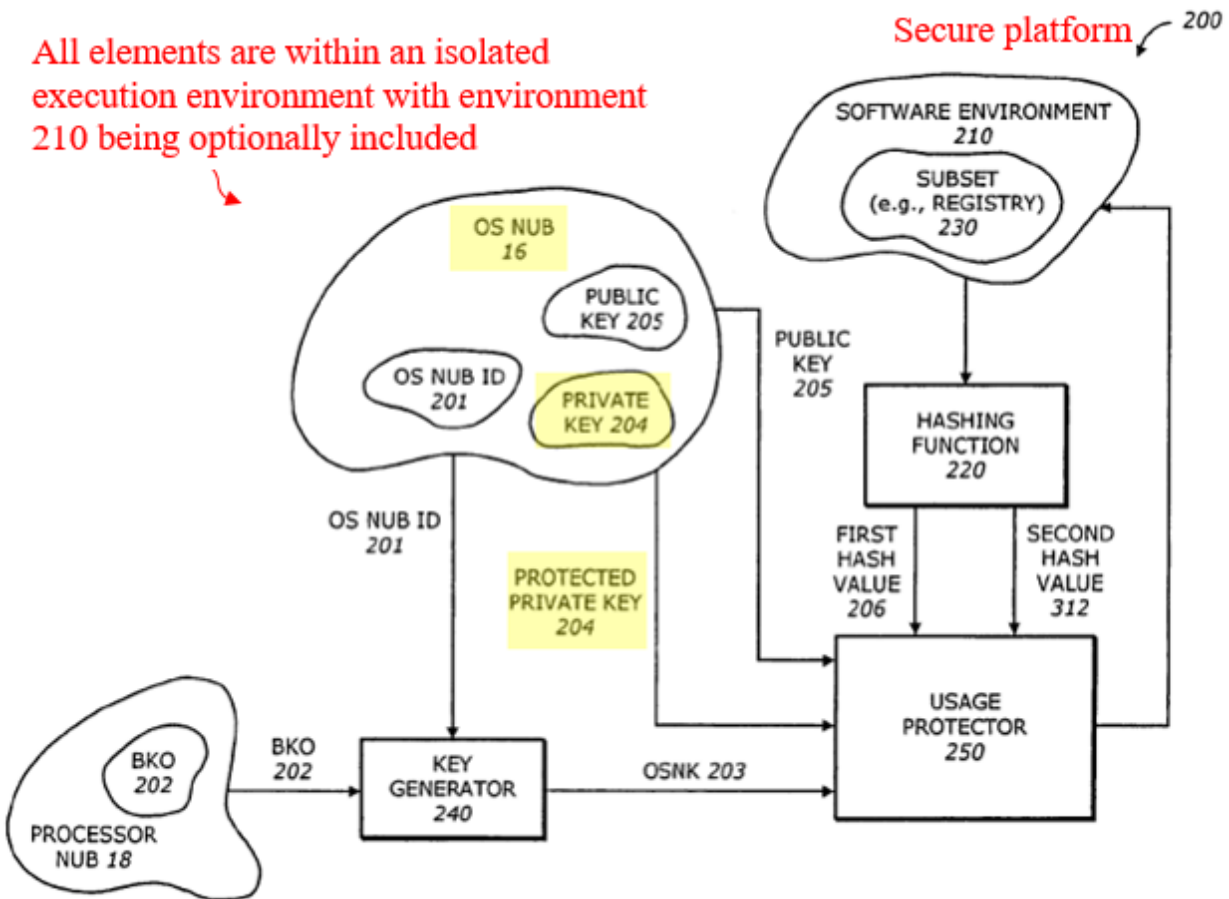


FIG. 2

SAMSUNG-1006 (Ellison), FIG. 2

96. As I noted above, the private key 204 can be stored in the cryptographic key storage 155 and in a multiple key storage of the non-volatile flash memory 160.

SAMSUNG-1006, 8:33-65. Moreover, “[o]nly the OS nub 16 can retrieve and decrypt the encrypted private key for subsequent use.” *Id.* Because only the OS nub 16 can retrieve the private key and the OS nub 16 is within the secure platform and isolated environment, it would have been obvious to a POSITA that the cryptographic key storage 155 and the non-volatile flash memory 160 are secure

storage locations so that the private key 204 can be accessed securely by OS nub 16 and at an area within non-volatile memory 160 that is inaccessible by the normal OS or the user of the device. *See also infra* [1e]. Ellison explicitly teaches that the OS nub 16 and private key 204 shown in platform 200 all ***operate within the isolated execution environment***, which would indicate to a POSITA that private key 204 is stored in a secure memory and is inaccessible to a user of said electronic device. Indeed, since a private key is an important element for maintaining security in a cryptographic process or more generally a secure electronic transaction, it would have been obvious to a POSITA to store the ***private key 204 in a secure memory location that is inaccessible to a user***. This obviousness is reinforced by the OS nub 16 and processor nub 18 being located in isolated execution Ring-0 15, which encompasses the most critical and highest degree of privilege—in contrast to Ring-3 which ***provides user level*** privilege. SAMSUNG-1006, 3:9-4:22, 2:45-61; *see supra* Section VI.A.

97. To the extent Patent Owner contends that Ellison does not store the private key 204 in a secure memory location, EMC also renders this feature obvious in view of Muir's disclosure, which explicitly teaches ***a private key 167 being stored in a restricted memory space 170 which is inaccessible to the operating system***. SAMSUNG-1008, pp. 7-¶1, 8-¶4, 9-¶3, 3-¶3. Since Muir's restricted memory space 170 can store a plurality of private keys and is not accessible to the operating

system to maintain security of the private keys, it would have been obvious to a POSITA that the restricted memory space 170 is not accessible to the user.

Moreover, given the use of private key 204 by an OS nub 16 that operates within the isolated execution environment in the EMC system, it would have been obvious to a POSITA to provide Ellison's private key 204 in a secure memory location (as revealed by Muir) and that the private key would have been inaccessible to a user of the electronic device.

[1b] wherein the private key is encrypted when the private key is stored in said memory,

98. As I noted above in [1a], ***the encrypted/protected private key 204 is stored at a secure location in multiple key storage*** of non-volatile flash memory 160, programmed into the cryptographic key storage 155, or another secure storage within secure platform 200 (e.g., Muir's restricted memory space 170).

SAMSUNG-1006, 8:33-65; SAMSUNG-1008, pp. 7-¶1, 8-¶4, 9-¶3. OS nub 16 retrieves the encrypted private key 204 and decrypts it for use in the isolated environment. SAMSUNG-1006, 8:33-65, 11:1-12:26. An example of this decryption is shown in Ellison's FIGS. 3C and 3D (reproduced below), which show decryptor 325 decrypting an encrypted private key 204 using an operating system nub key (OSNK) 203, and thereby yielding an unencrypted private key 328. *Id.*

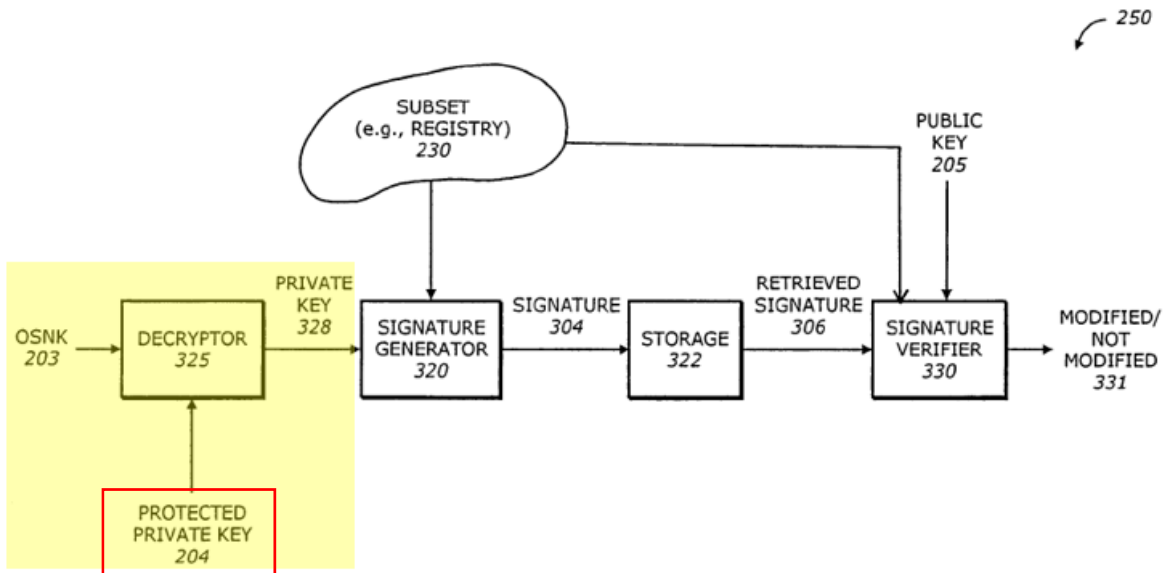


FIG. 3C

SAMSUNG-1006 (Ellison), FIG. 3C

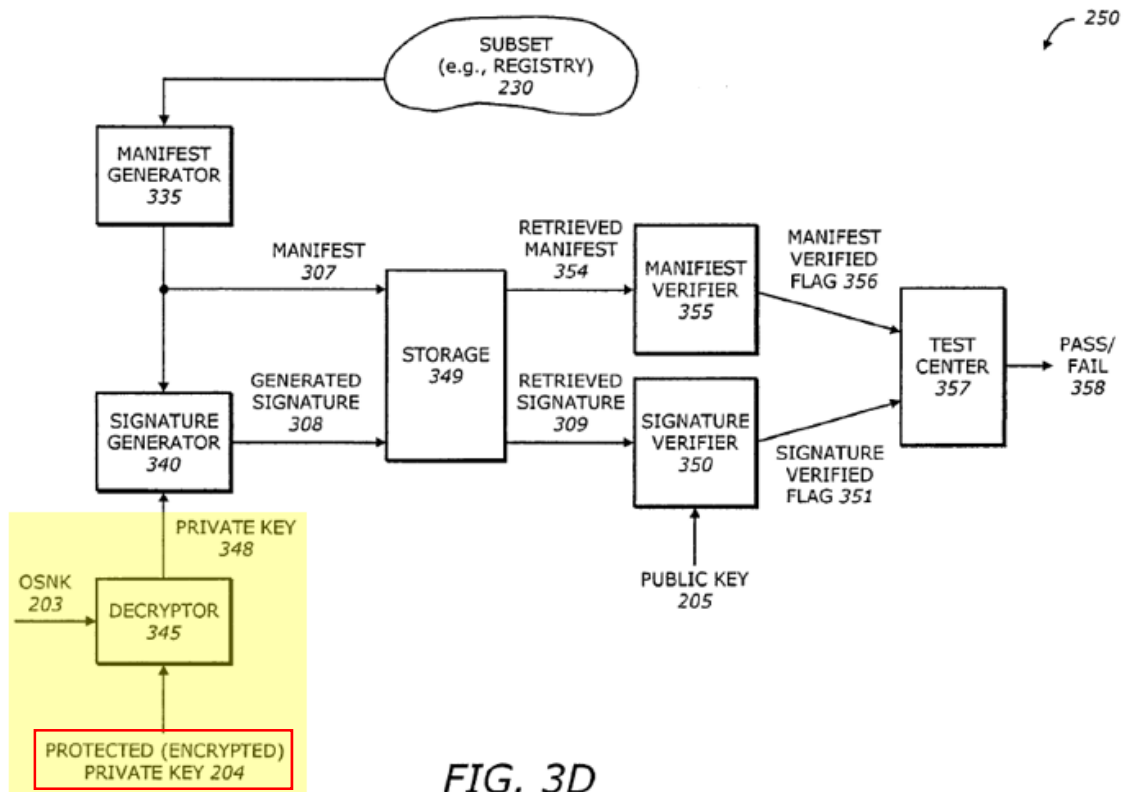


FIG. 3D

SAMSUNG-1006 (Ellison), FIG. 3D

99. Accordingly, Ellison and Muir render obvious *the private key 204 being encrypted when stored in a secure memory (e.g., memory 160 or Muir's restricted memory space 170)* and obtained therefrom by OS nub 16. *See also supra* [1a].

[1c] a decryption key for decrypting the private key being incorporated in the electronic device,

100. As I noted above in [1b], the *OSNK 203 is used as a decryption key for decrypting the private key 204*. SAMSUNG-1006, 8:33-65, 11:1-30, FIG. 3C.

The *OSNK 203 is incorporated in the electronic device of the first transaction party*. For example, the OSNK 203 is generated by the key generator 240, which is part of the isolated execution environment in the electronic device of the first transaction party. SAMSUNG-1006, 8:25-32, 9:1-14, FIG. 2 (reproduced below). As I explained above in [1a], the private key also is incorporated in the electronic device.

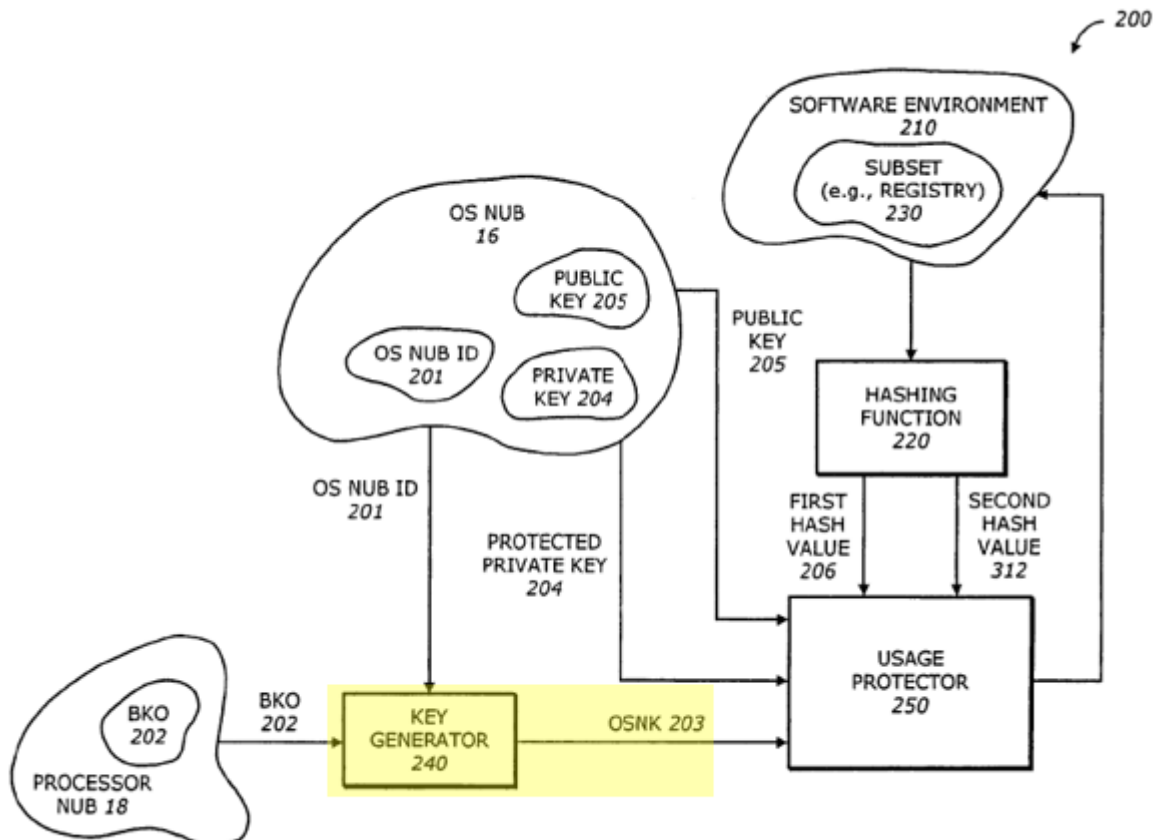


FIG. 2

SAMSUNG-1006 (Ellison), FIG. 2

[1d] said decryption key being inaccessible to said user, to any user-operated software and to said operating system,

101. As described above in [1a]-[1c], the OSNK 203 (“*decryption key*”) is generated and used within the *isolated environment region, which is not accessible to a user, any user-operated software, and the operating system (OS)* 12 of the electronic device in the normal execution mode. SAMSUNG-1006, 8:25-65, 11:1-12:26, 9:1-14, FIGS. 2, 3C, 3D. To prevent a user or malicious software controlling one or more operations of the operating system from corrupting sensitive or personal data that needs to be protected, such as the OSNK 203 or

private key, Ellison's system architecture provides the isolated environment region in which the usage protector 250, key generator 240, and nubs 16 and 18 can execute their operations in a secure environment that is free from interference and precludes access by the user, user-operated software, or OS 12. Ellison explicitly discloses that OSNK 203 "is provided only to the OS Nub 16," which may further "supply the OSNK 203 to *trusted agents, such as the usage protector 250.*"

SAMSUNG-1006, 9:1-14. Ellison teaches that the OS nub 16 and the processor nub 18 operate in a secure environment associated with the isolated area 70 and the isolated execution mode" and are present in Ring-0 10 of the isolated execution environment, which has the least amount of privileges. SAMSUNG-1006, 3:9-25, 4:8-22, FIG. 1A. Moreover, the key generator 240 and usage protector 250, which generate and use the OSNK 203, are each described as being part of the isolated execution environment in secure platform 200. SAMSUNG-1006, 8:12-32.

Accordingly, and as I explained above in [1a], a POSITA would have understood that the isolated execution region is inaccessible to the user, any user-operated software, and OS 12. Indeed, as shown in Ellison's FIG. 1A, the isolated execution portion of the operating architecture 50 contrasts with the normal execution portion in which OS 12 resides as an unprotected OS, and supports user-accessible applications 421-42N. SAMSUNG-1006, 3:55-56, 3:9-31.

[1e] wherein said memory is inaccessible to said operating system of said electronic device, thereby rendering the private key inaccessible to said user,

102. See *supra* [1a], [1pre-2]. For example, OS 12 is in Ring-0 11 of the normal execution region of architecture 50 and does not access components of the isolated execution region, as I explained above in [1a] and [1d]. SAMSUNG-1006, 3:9-4:29, FIG. 1A. The cryptographic key storage 155 is configured to store cryptographic keys, and is part of the ICH 150, which operates in the isolated execution mode. SAMSUNG-1006, 6:64-67, 6:27-43, 4:8-37. Non-volatile flash memory 160 is coupled to the ICH 150 and includes the processor nub 18, which also operates in the isolated execution mode (see FIGS. 1A, 1B, 1C). SAMSUNG-1006, 7:19-22. Accordingly, because cryptographic key storage 155 and memory 160 operate in the isolated execution mode, it would have been obvious to a POSITA that the cryptographic key storage 155 and the non-volatile flash memory 160 are secure storage locations that are inaccessible to the OS 12 and thereby the user. As I explained above, the protected private key 204 “can be stored in the multiple key storage ... of the non-volatile flash memory 160” and can be “programmed into the fuses of a cryptographic key storage 155 of the ICH 150 or elsewhere in persistent storage within platform 200.” SAMSUNG-1006, 8:33-65, FIG. 1C. Therefore, a POSITA would also have found it obvious that the private key is inaccessible to the user.

[1f] wherein the private key is decrypted in a secure processing environment inaccessible to said user and to any user-operated software;

103. As I noted in [1a]-[1d], Ellison's private key 204 is decrypted using the decryption key, OSNK 203, in the isolated execution environment of secure platform 200 that is inaccessible to a user or user-operated software. SAMSUNG-1006, 8:33-65, 11:1-30, FIG. 3C.

[1g] providing authentication software in said electronic device,

104. Ellison's isolation execution architecture ***provides authentication software in the electronic device in the form of the usage protector 250***, which is part of the secure platform 200 in the electronic device (e.g., Muir's device 256 in EMC) associated with the first transaction party and operates within the isolated execution environment. SAMSUNG-1006, 8:25-32, FIG. 2 (reproduced below).

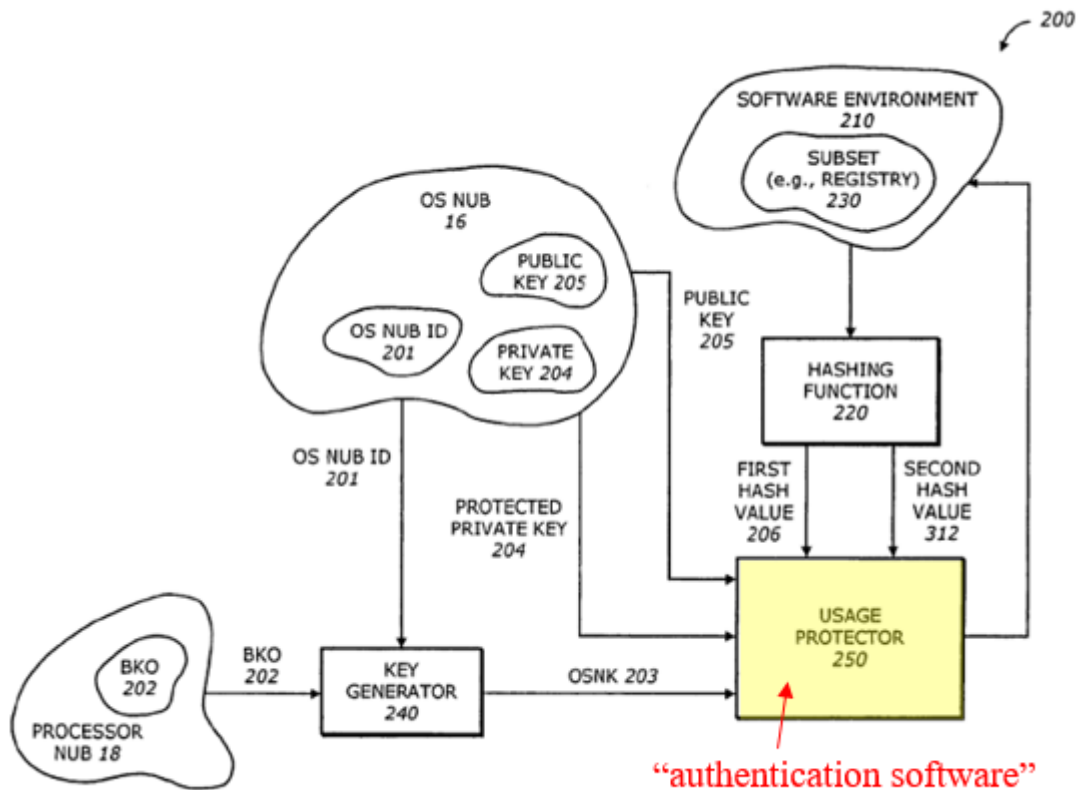


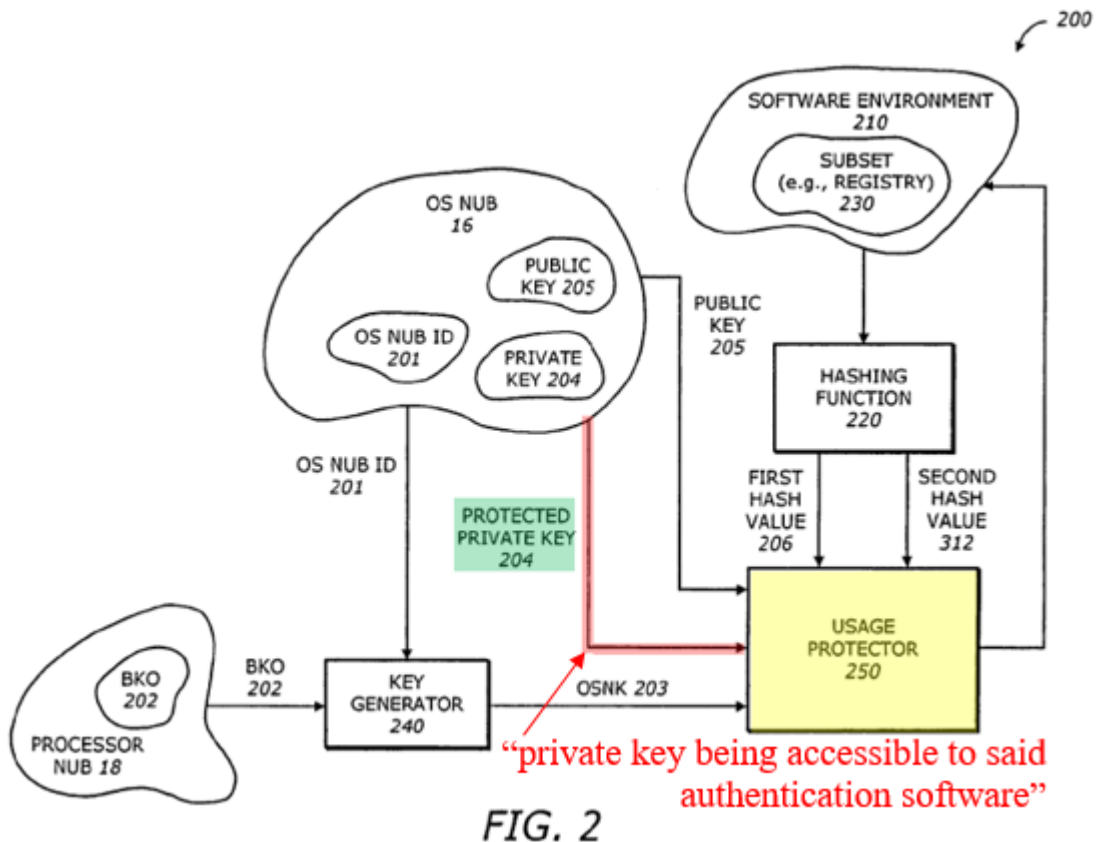
FIG. 2

SAMSUNG-1006 (Ellison), FIG. 2

105. One example, among several authentication operations of the usage protector 250, is that the usage protector 250 compares a first hash value 206 and a second hash value 312 after a time period to authenticate subset 230 and detect intrusion or modification of the subset 230. SAMSUNG-1006, 9:40-58. FIGS. 3A-3E depict various other implementations of the usage protector 250 and its operations (that also include authentication operations). SAMSUNG-1006, 10:4-13:3.

[1h] the private key being accessible to said authentication software,

106. As I explained above in [1a] and [1b] and shown clearly in Ellison's FIG. 2, private key 204 is accessible to the usage protector 250 ("**authentication software**"). SAMSUNG-1006, 10:1-3, 10:63-12:26, 8:40-65.



SAMSUNG-1006 (Ellison), FIG. 2

107. FIGS. 3C and 3D are two example implementations of the usage protector 250 that show the private key 204 being accessible to the usage protector 250 and used for decryption purposes. SAMSUNG-1006, 10:63-12:26.

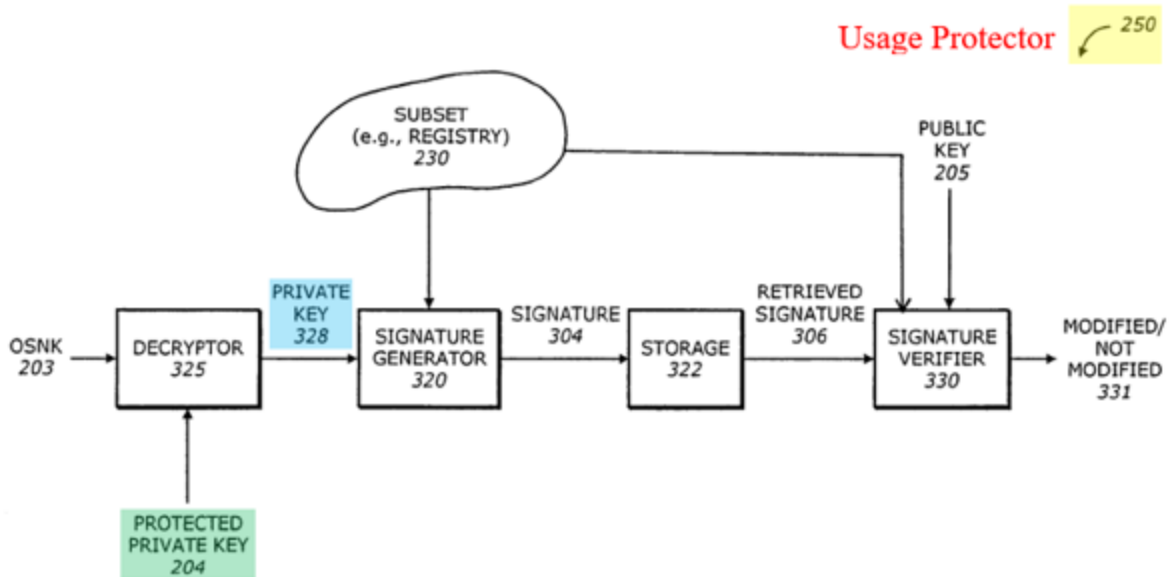


FIG. 3C

SAMSUNG-1006 (Ellison), FIG. 3C

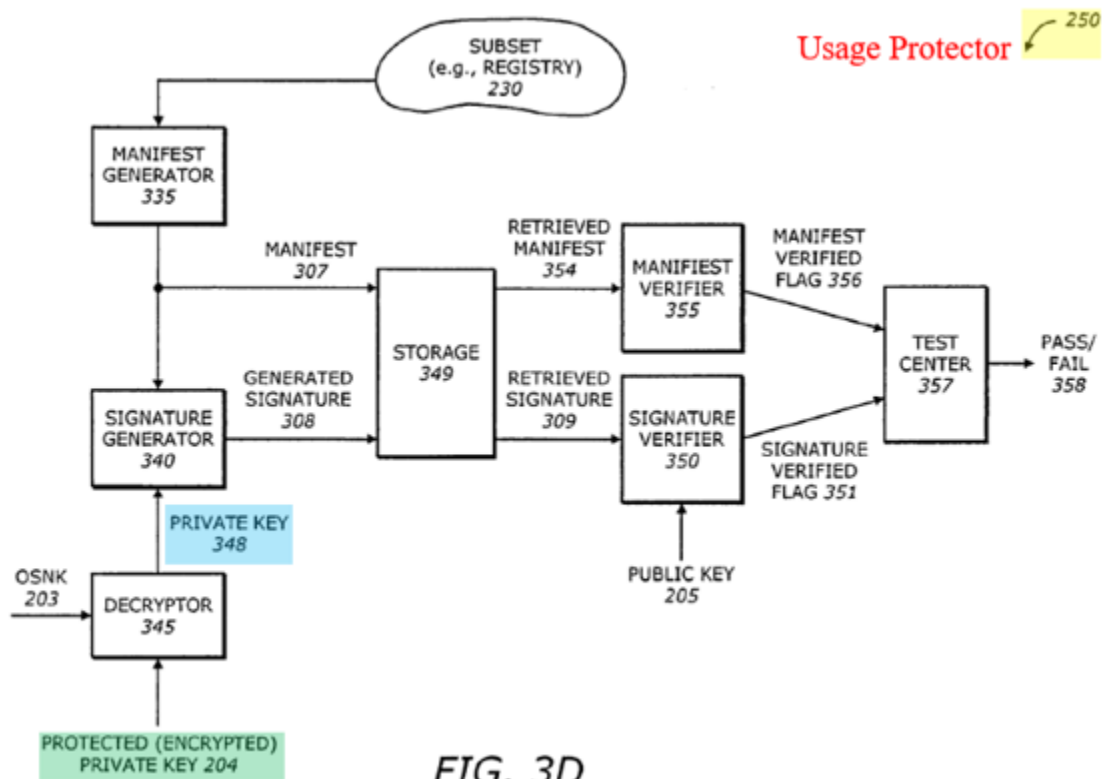


FIG. 3D

SAMSUNG-1006 (Ellison), FIG. 3D

[1i] wherein authentication software is stored in said secure memory inaccessible to said operating system;

108. As I explained above in [1a]-[1d], the ***usage protector 250 (“authentication software”)*** is located in a ***secure*** platform 200 and ***operates within the isolated execution environment*** (*note: the location of the usage protector 250 can be contrasted with the software environment 200, which Ellison explains can be inside or outside the isolated execution environment*). SAMSUNG-1006, 8:25-32. The ***usage protector 250 is a trusted agent***. SAMSUNG-1006, 9:2-8.

Furthermore, Ellison explains that the operating system (OS) nub 16 and processor nub 18 are located in Ring-0 15 of the isolated execution environment, and are operated in a secure environment associated with isolated area 70.” SAMSUNG-1006, 3:15-25. “The isolated area 70 is accessible only to elements of the operating system and processor operating in isolated execution mode.” SAMSUNG-1006, 4:19-22.

109. Based on the foregoing descriptions of the usage protector 250, the isolated execution mode, and the secure environment, it would have been obvious to a POSITA that the usage protector 250 would have been stored in secure memory (e.g., isolated area) inaccessible to elements outside of the isolated execution environment such as OS 12. And even if the usage protector 250 is loaded from another memory into the isolated execution environment, code for executing the usage protector 250 while in the isolated execution environment would have been

stored in a secure memory so as to prevent unauthorized access of sensitive data. For example, various implementations of the usage protector 250 shown in FIGS. 3A-3E include operations involving encryption and decryption using a private key. SAMSUNG-1006, 10:4-13:3. It would have been obvious to a POSITA that, to perform encryption and decryption using a private key—operations for which a high level of security are vital—all or at least the authentication portion of the usage protector 250 and its operational software/code would have been stored in a secure memory of the isolated execution environment such as isolated area 70.

[1j] activating the authentication software to generate a digital signature from the private key,

110. Ellison's FIG. 3C (reproduced below) depicts an illustration of the usage protector 250 in which ***the signature generator 320 generates a digital signature 304 for the subset 230 using private key 328***, which is a decrypted version of encrypted private key 204. SAMSUNG-1006, 10:63-11:30, 8:55-65.

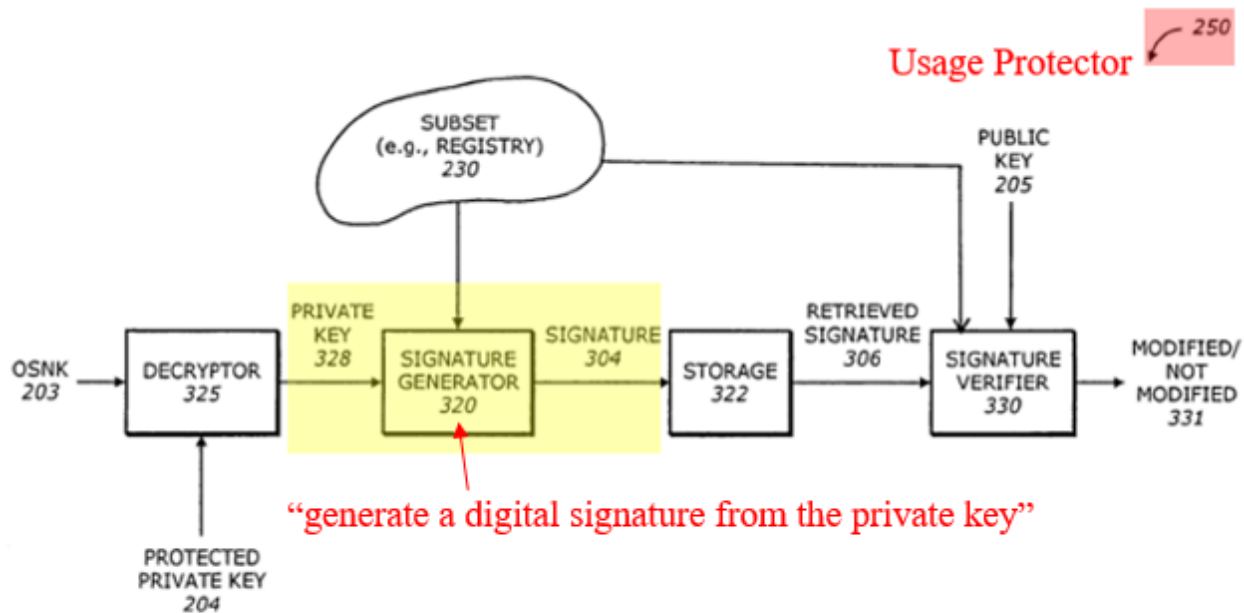


FIG. 3C

SAMSUNG-1006 (Ellison), FIG. 3C

[1k] wherein the authentication software is run in a secure processing environment inaccessible to said operating system;

111. As I noted above in Sections VI.A, [1pre-2], [1a]-[1d], and [1i], the usage protector 250 is run in an isolated environment (“*secure processing environment*”) that is not accessible to the OS 12 (“*said operating system*”). SAMSUNG-1006, 3:9-61, 4:1-29, 4:57-5:27, FIGS. 1A, 1B.

[1l] providing the digital signature to the second transaction party.

112. After the digital signature 304 is generated (as described above in [1j]), Muir teaches that the *generated digital signature is provided to another user* (“*second*

transaction party”) connected in the networking environment to the computer system 20 to allow the other user “to determine the identity” of the first transaction party. SAMSUNG-1008, pp. 9-¶3, 2-¶1, 3-¶¶1-2, 6-¶3, 10-¶¶1-2, 12-¶1, 18-¶6. As I explained in Section VI.C, doing so allows data to be securely transmitted, received, and decrypted by second transaction party.

B. Claim 3

[3] The method according to claim 1, wherein the authentication software has its own unique serial number, software ID or encryption key.

113. Claim 3 is rendered obvious by EMC.

114. Ellison’s *usage protector 250 has its own encryption key in the form of the operating system nub key (OSNK) 203*. As shown in FIG. 2 below, the “key generator 240 generates the OSNK 203 by combining the BK0 202 and the OS Nub ID 201 using a cryptographic hash function.” SAMSUNG-1006, 9:9-11.

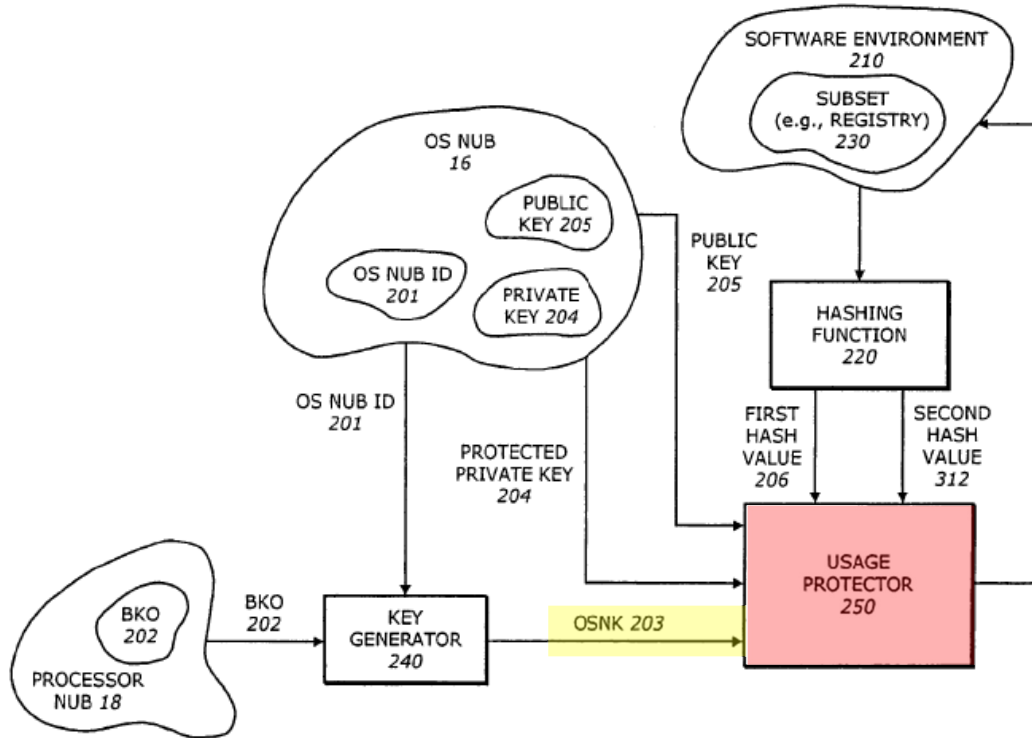


FIG. 2

SAMSUNG-1006 (Ellison), FIG. 2

115. The “cryptographic hash function is a one-way function, mathematical or otherwise, which takes a variable-length input string, called a pre-image and converts it to a fixed length, generally smaller, output string referred to as a hash value.” SAMSUNG-1006, 9:17-22. In one embodiment, “[t]he OS nub ID 201 can be the hash of the OS nub 16, or a hash of a certificate that authenticates the OS nub 16, or an ID value extracted from a certificate that authenticates the OS nub 16.” SAMSUNG-1006, 9:11-17. “The usage protector 250 uses the OSNK 203 to protect the usage of the subset 230.” SAMSUNG-1006, 9:31-35. For example, the usage protector 250 *uses the OSNK 203 as an encryption key to*

encrypt the first hash value 206 obtained from subset 230. SAMSUNG-1006, FIG. 3B, 9:48-10:3, 10:32-62 (“[t]he encryptor 305 *encrypts* the first hash value 206 *using the OSNK 203* to generate an encrypted first hash value 302”).

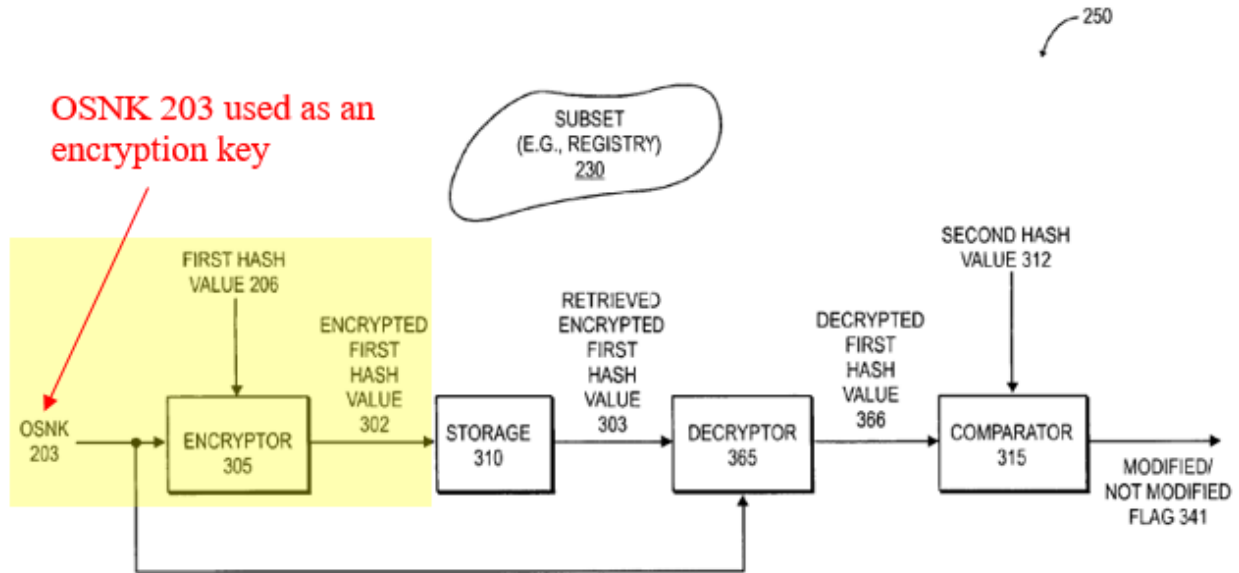


FIG. 3B

SAMSUNG-1006 (Ellison), FIG. 3B

C. Claim 6

[6] The method according to claim 1, wherein the authentication software is installed in an application environment in the secure area such that it may obtain the private key without passing through an unsecured part of said electronic device.

116. Claim 6 is rendered obvious by EMC.

117. As I explained above in [1a] and [1i], “the OS nub 16, the processor nub 18, a key generator 240, a hashing function 220, and a *usage protector 250, all*

operat[e] within the isolated execution environment” and form part of the *secure platform 200 (“secure area”)*. SAMSUNG-1006, 8:25-32. To operate securely within the isolated execution environment, it would have been obvious to a POSITA that the usage protector 250 (*“authentication software”*) is installed in an application environment in the secure area.

118. As shown in FIG. 2 below, since usage protector 250 and OS nub 16 are both part of the isolated execution environment, the usage protector 250 may obtain the private key without passing through an unsecured part of said electronic device.

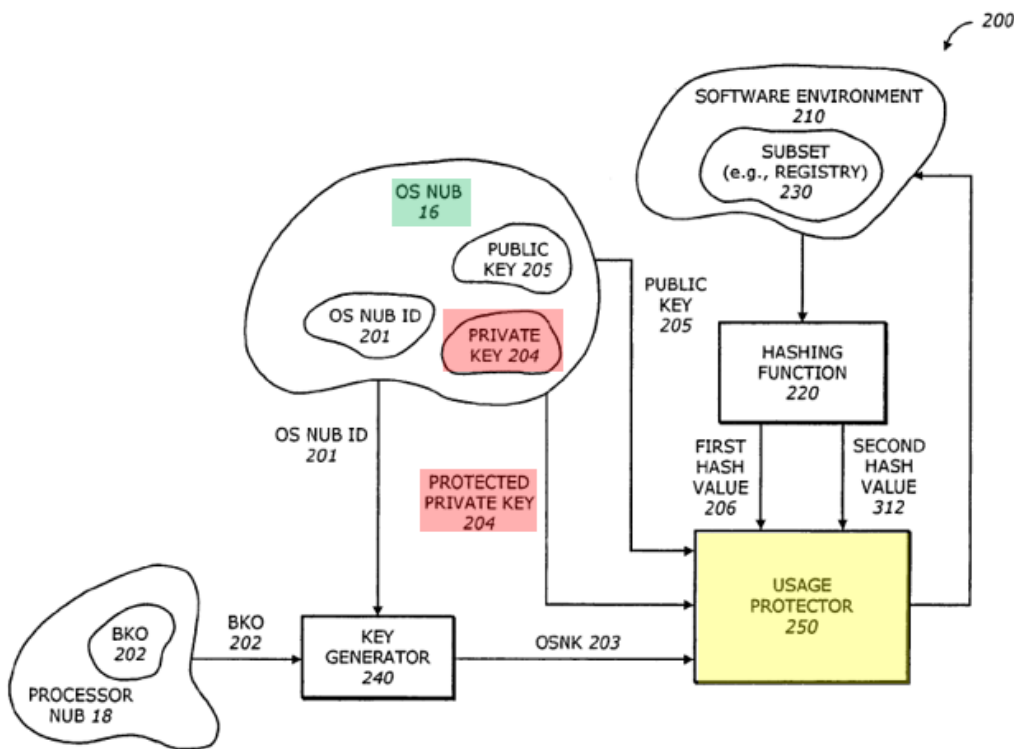


FIG. 2

SAMSUNG-1006 (Ellison), FIG. 2

119. In particular, Ellison teaches that “[i]n one embodiment, the protected private key 204 is generated by the platform 200 itself the first time the platform 200 is powered up.” SAMSUNG-1006, 8:48-52. “Only the OS nub 16 can retrieve and decrypt the encrypted private key [204] for subsequent use.” SAMSUNG-1006, 8:59-61. The subsequent use involves use by the usage protector 250 to perform operations such as decryption operations, as I explained above in [1b]. SAMSUNG-1006, 8:33-65, 11:1-12:26, FIGS. 3C, 3D. At least in such an embodiment, the usage protector 250 “may obtain the private key without passing through an unsecured part of said electronic device” since the usage protector 250, the OS nub 16, and the private key 204 are all within the isolated execution environment.

D. Claim 7

[7] The method according to claim 1, wherein the private key is encrypted using at least two encryption layers.

120. Claim 7 is rendered obvious by EMC and Al-Salqan.

121. As I explained above in Sections VI.D-VI.E, the combination of Ellison, Muir, and Al-Salqan (EMAS) renders obvious ***a private key that is encrypted using at least two encryption layers***. For instance, a first encryption layer would have been a symmetric encryption layer implemented when Al-Salqan’s symmetric encryptor 230 encrypts encoded private information, and a second encryption layer

would have been an asymmetric encryption layer implemented when Al-Salqan's asymmetric encryptor 242 encrypts the output from the symmetric encryptor 230. SAMSUNG-1007, 2:50-60, 4:4-5:9, FIG. 2.

E. Claim 8

[8] The method according to claim 1, wherein the private key is encrypted using an encryption key which is associated with a hardware device serial number.

122. Claim 8 is rendered obvious by EMC and Searle.

123. As I explained above in Sections VI.F-VI.G, the Ellison-Muir-Searle combination renders obvious [8] at least because Searle teaches encrypting a second key (“***the private key***”) using a first key (“***an encryption key***”) which is associated with “***a hardware device serial number***” (e.g., “a unique serial number or other identifier” of a processor, a network card address, serial numbers and/or the number of cylinders of attached hard disk drives). SAMSUNG-1009, 2:13-33, 4:58-5:7.

F. Claim 9

[9] The method according to claim 1, wherein the private key is encrypted using an encryption key which is associated with a user identifying number, in particular a personal identifying number, PIN, or a number or a template associated with a fingerprint of the user.

124. Claim 9 is rendered obvious by EMC and Al-Salqan.

125. As I explained above in Sections VI.D-VI.E, the EMAS combination renders obvious Ellison's private key 204 being encrypted using an encryption key which is associated with a user identifying number or personal identifying number such as the user's social security number. SAMSUNG-1007, Abstract, 2:49-63, 4:4-7. Accordingly, EMAS renders obvious "wherein *the private key is encrypted using an encryption key which is associated with a user identifying number*, in particular *a personal identifying number*, PIN, or a number or a template associated with a fingerprint of the user."

G. Claim 10

[10] The method according to claim 1, wherein the decryption key is associated with one or more serial numbers of hardware components of said electronic device.

126. Claim 10 is rendered obvious by EMC and Searle.

127. As I noted above in Sections VI.F- VI.G, Searle also teaches decrypting the second key using a first key ("*decryption key*") that is "associated with one or more serial numbers of hardware components of said electronic device" such as a serial number of a processor, network address card, cylinders of hard disk drives. SAMSUNG-1009, 2:13-40, 4:58-5:7, 5:65-6:27, FIG. 1.

H. Claim 11

[11] The method according to claim 1, wherein the private key is decrypted by the authentication software.

128. Claim 11 is rendered obvious by EMC.

129. As I explained in [1b], private key 204 is decrypted by decryptor 325, which is part of the usage protector 250 (“*authentication software*”). SAMSUNG-1006, 8:33-65, 11:1-12:26, FIGS. 3C, 3D (reproduced below).

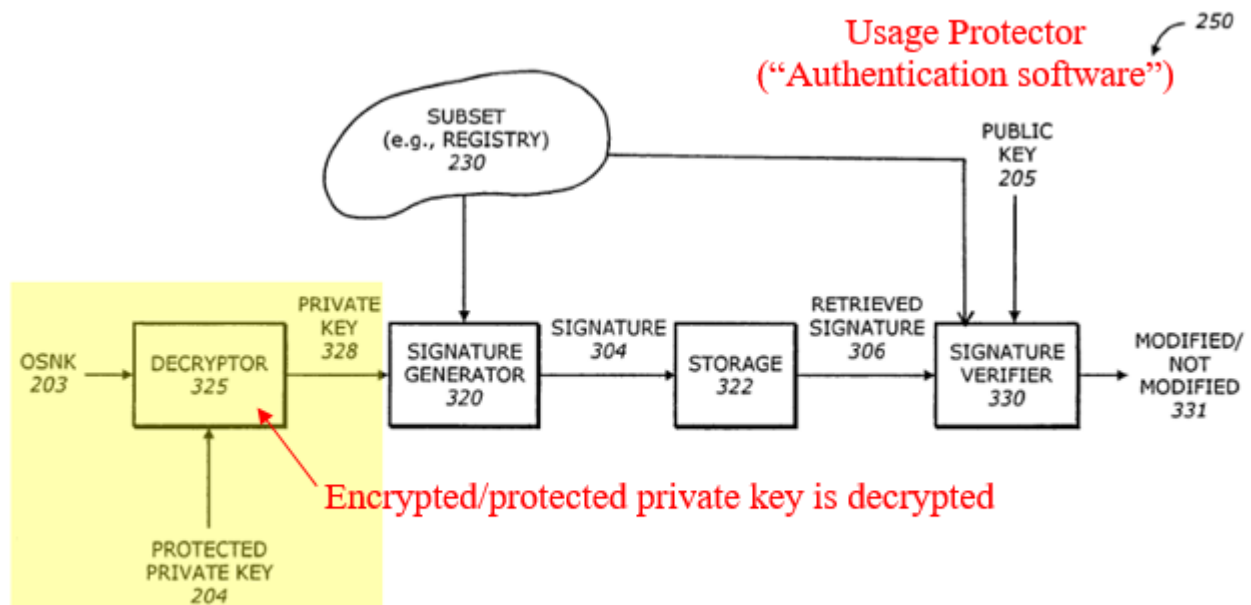


FIG. 3C

SAMSUNG-1006 (Ellison), FIG. 3C

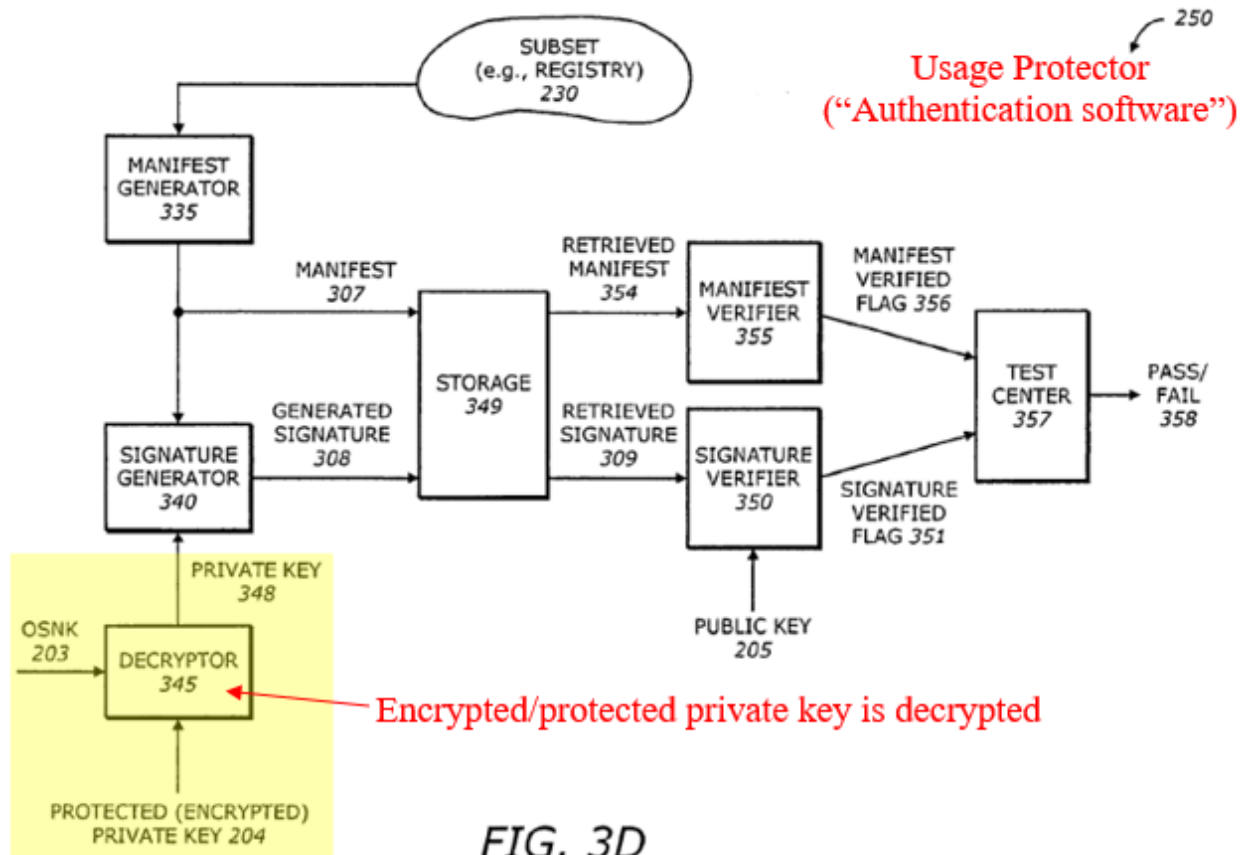


FIG. 3D

SAMSUNG-1006 (Ellison), FIG. 3D

I. Claim 12

[12] The method according to claim 1, wherein the private key is decrypted by an application stored in the electronic device using a device specific encryption key.

130. Claim 12 is rendered obvious by EMC and Searle.

131. For the reasons I explained above in [8] and [10], it would have been obvious to a POSITA that, in the Ellison-Muir-Searle combination, “the private key is decrypted ... using a device specific encryption key.”

132. It would also have been obvious to a POSITA that the decryption is performed “by an application stored in the electronic device. For example, Ellison’s FIGS. 3C and 3D depict a decryptor 325/345 that decrypts an encrypted private key 204 using OSNK 203. SAMSUNG-1006, 10:63-11:40. The decryptor 325/345 is part of the usage protector 250, which is described as being “a subset of a software environment.” *Id.*, 2:7-12, 10:63-11:40. Since an application would have generally been understood by a POSITA to be a computer program (software) that performs a specific task, and Ellison’s decryptor 325/345 is a program in Ellison’s software environment tasked with decrypting, it would have been obvious to a POSITA that decryptor 325 is an application stored in the electronic device of the Ellison-Muir-Searle combination. SAMSUNG-1022, 56-57.

J. Claim 13

[13] The method according to claim 1, wherein the private key is decrypted using a decryption key associated with any identifying characteristic, in particular a serial number, a personal identification number, PIN, or a fingerprint of a user.

133. Claim 13 is rendered obvious by EMC and Al-Salqan.

134. As I explained above in Sections VI.D-VI.E, in the EMAS combination, Al-Salqan’s asymmetric decryptor 314 uses the encoded ***private information*** received from private information encoder 324 ***as the decryption key.***” SAMSUNG-1007, 5:25-35. The private information includes the “social security number, mother's

maiden name, and other similar information.” SAMSUNG-1007, 4:4-7, Abstract, 2:49-63. Accordingly, EMAS renders obvious “wherein the private key is decrypted using *a decryption key associated with any identifying characteristic*, in particular a serial number, *a personal identification number*, PIN, or a fingerprint of a user.”

K. Claim 14

[14] The method according to claim 1, further comprising identifying the user of the electronic device before activating the authentication software.

135. Claim 14 is rendered obvious by EMC.

136. Ellison discloses using a symmetric key to protect the secrets of the processor nub 18 and the operating system nub 16, but does not provide details of how the symmetric key is generated. SAMSUNG-1006, 6:51-57, 7:19-32. Muir discloses that before performing encryption and decryption operations, a user provides a user ID/password to confirm an identification of the user. SAMSUNG-1008, 15-¶¶3-5, FIG. 8. It would have been obvious to a POSITA to combine the above-noted teachings of Ellison and Muir so that a user identity can be verified prior to conducting a secure transaction between two transaction parties. Indeed, providing login credentials, such as user ID/password, to identify a user is a well-known user authentication method and would have been obvious to implement in EMC prior to conducting encryption/decryption operations and/or a secure

transaction with another party so that the user of the electronic device can be authenticated. SAMSUNG-1019, 1:53-66 (“[m]ost login sequences begin with the host prompting the user for an identification name or number and a password”); SAMSUNG-1020, FIGS. 2, 7, 2:49-65, 7:40-52 (“[a] buyer manually performs authentication by entering their user identification and password”). An additional benefit here is that the information identifying the user can be used to generate a symmetric key, thereby enabling a user-specific symmetric key to protect electronic device elements such as the processor nub 18 and the operating system nub 16. Combining the teachings of Ellison and Muir would have involved combining prior art elements (Ellison’s system and Muir’s user identification mechanism) according to known methods (obtaining user id/password information from a user) to yield predictable results (generation of a user id/password-based symmetric key). Accordingly, EMC renders [14] obvious.

L. Claim 15

[15] The method according to claim 1, wherein the authentication software has an encryption key for encrypting digital data.

137. Claim 15 is rendered obvious by EMC.

138. As I explained above in [3], the usage protector 250 (“***authentication software***”) has an encryption key in the form of OSNK 203. The ***OSNK 203 (“encryption key”) is used for encrypting various data*** such as a compressed

subset 377 (*see* SAMSUNG-1006, FIG. 3A, 10:4-31), a first hash value 206, and a second hash value 312 (*see* SAMSUNG-1006, FIGS. 3B, 3E, 4, 10:32-62, 12:27-13:20).

139. A POSITA reading Ellison would have understood that Ellison's disclosure relates to encrypting and decrypting digital data. For example, when the usage protector 250 decrypts an encrypted private key 204 by using OSNK 203, a ***digital*** signature algorithm is applied by the signature generator 320 on the resulting private key 204 suggesting to a POSITA that the data being processed is digital.

SAMSUNG-1006, 11:1-16. Ellison discloses that the "signature algorithm used by the signature generator 320 may be public-key ***digital*** signature algorithm" and that "[e]xample algorithms include ... ***Digital*** Signature Algorithms [sic] schemes."

SAMSUNG-1006, 11:1-16, 8:59-62. Moreover, a POSITA would have understood that, with the inclusion and use of "a compressor 370, an encryptor 375, ... a decryptor 385, and a decompressor 390," the usage protector 250 is configured to process ***digital data*** (e.g., encrypt digital data). SAMSUNG-1006, 10:4-8. Thus, a POSITA would have found it obvious that Ellison encrypts digital data. For at least these reasons, EMC renders [15] obvious. *Id.*

M. Claim 16

[16pre] An electronic device comprising

140. Claim 16 is rendered obvious by EMC. *See supra* [1pre-1].

[16a] a network interface configured for an electronic transaction between a first transaction party, and a second transaction party, wherein the electronic device is operated by the first transaction party;

141. *See supra* [1pre-1]. In addition, Muir's FIG. 5 shown below discloses the network interface for an electronic transaction between a first transaction party and a second transaction party. SAMSUNG-1008, pp. 9-¶2, 11-¶3, 1-¶2.

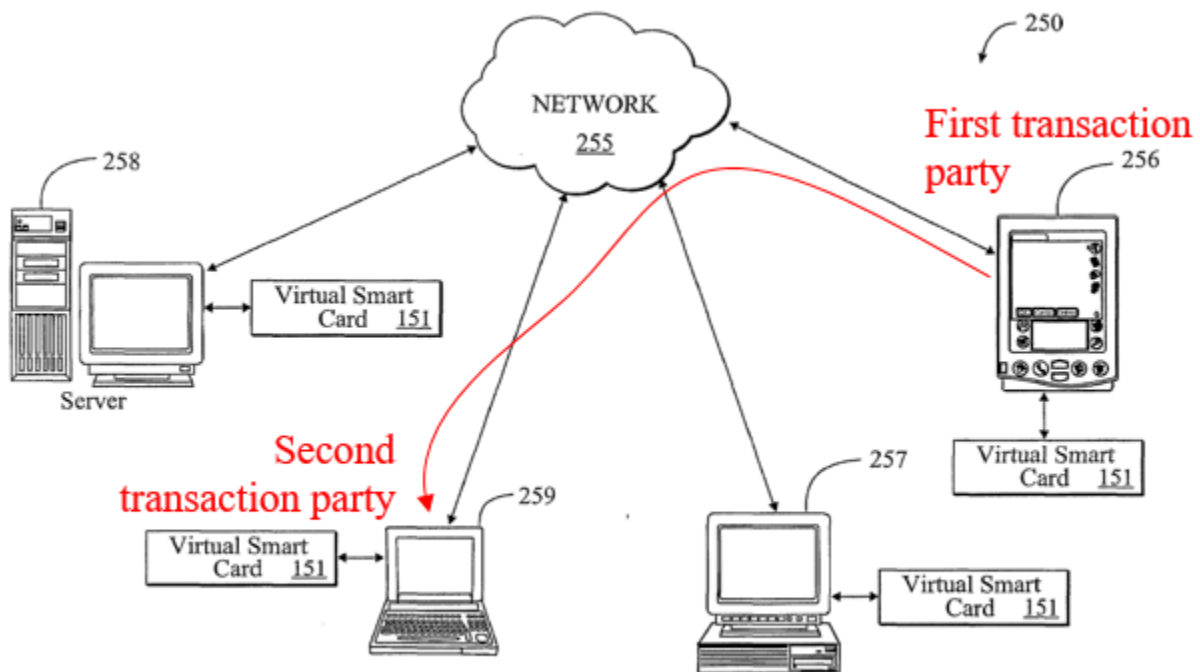
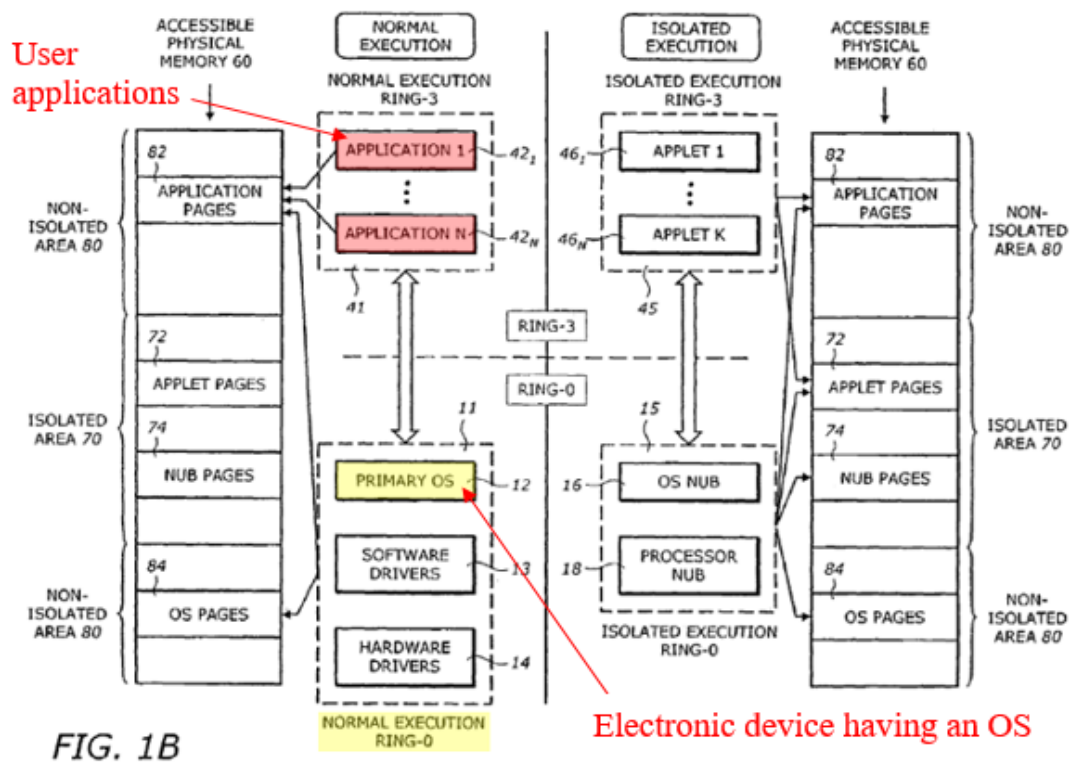


Fig. 5

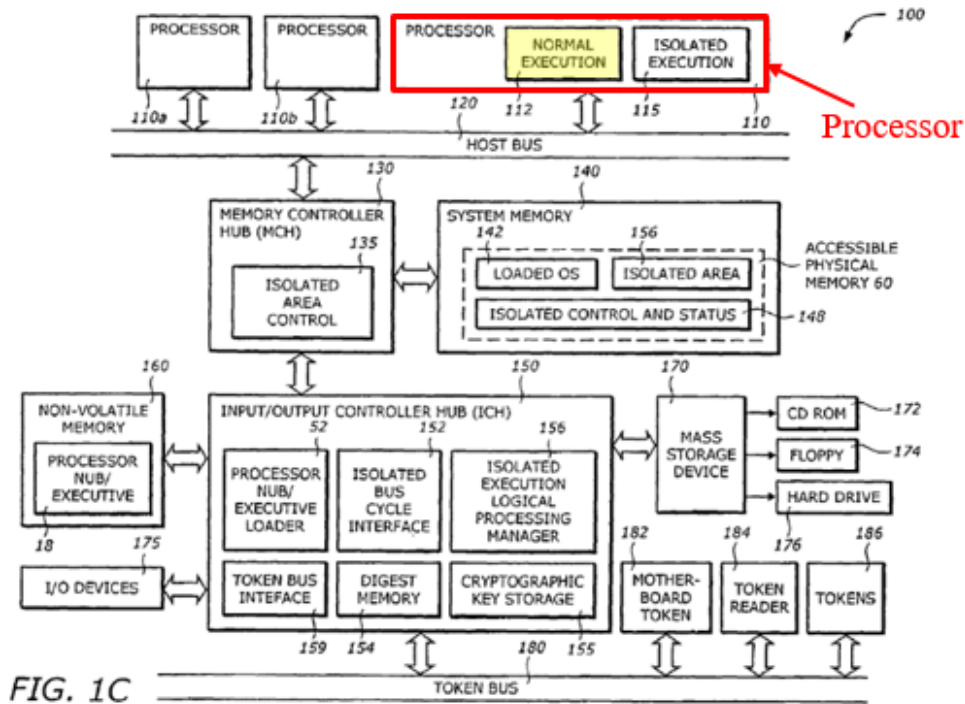
SAMSUNG-1008 (Muir), FIG. 5

[16b] a processor configured for an operating system creating a run-time environment for user applications;

142. As I explained in [1pre-2], Ellison discloses an operating system, OS 12, “creating a run-time environment for user applications.” Processor 110 is configured for OS 12 in the normal execution mode 112. SAMSUNG-1006, 4:23-29, 4:57-5:27, FIGS. 1A, 1C.



SAMSUNG-1006 (Ellison), FIG. 1B



SAMSUNG-1006 (Ellison), FIG. 1C

[16c] a memory storing a private key, said memory being a secure part of a Basic In Out System or any other secure location in said electronic device, which private key is inaccessible to a user of the electronic device,

143. See *supra* [1a].

[16d] wherein the private key is encrypted when the private key is stored in said memory,

144. See *supra* [1b].

[16e] a decryption key for decrypting the private key being incorporated in the electronic device, said decryption key being inaccessible to said user, to any user-operated software and to said operating system;

145. See *supra* [1c], [1d].

[16f] wherein said memory is inaccessible to said operating system of said electronic device, thereby rendering the private key inaccessible to said user,

146. See *supra* [1e].

[16g] wherein the private key is decrypted in a secure processing environment inaccessible to said user and to any user-operated software; and

147. See *supra* [1f].

[16h] a memory storing authentication software, the private key being accessible to said authentication software wherein the authentication software is stored in secure memory location inaccessible to said operating system;

148. See *supra* [1g], [1h], [1i].

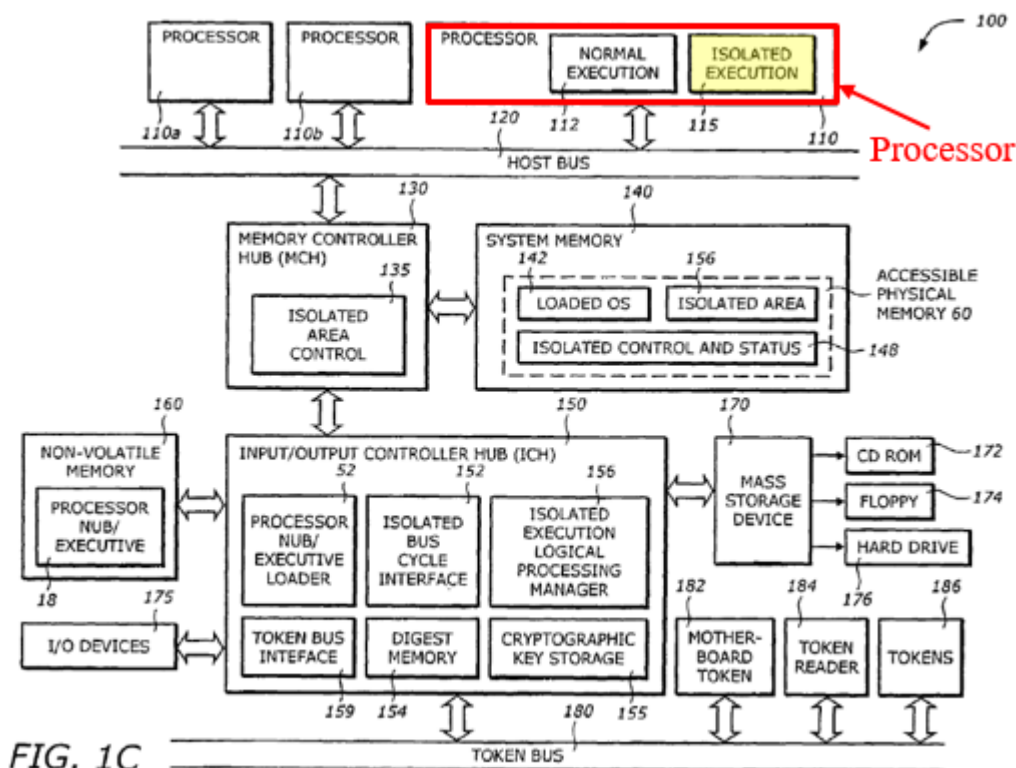
[16i] wherein the processor is configured for activating the authentication software to generate a digital signature from the private key,

149. As I explained in [1j], Ellison discloses “activating the authentication software to generate a digital signature from the private key.” Ellison also discloses that the processor 110 is configured for the activating. In particular, Ellison teaches that processor 110 also includes an “isolated execution circuit 115

[that] provides hardware and software support for the isolated execution mode.

This support includes configuration for isolated execution, definition of an isolated area, definition (e.g., decoding and execution) of isolated instructions, generation of isolated access bus cycles, and generation of isolated mode interrupts.”

SAMSUNG-1006, 5:1-10, FIG. 1C (reproduced below).



SAMSUNG-1006 (Ellison), FIG. 1C

150. In addition, as I explained above in [1i], it would have been obvious to a POSITA that, to perform encryption and decryption using a private key, all or at least a portion of the usage protector 250 and its operational software/code would have been stored in a secure memory of the isolated execution environment such as

isolated area 70. Ellison teaches that processor 110 can define and control access to the isolated area 70. SAMSUNG-1006, 6:1-26. It would have been obvious to a POSITA that because (i) processor 110 configures the isolated execution mode and controls access to the isolated area 70, and (ii) the usage protector 250 operates within the isolated execution environment and is stored in a secure memory such as isolated area 70, the processor 110 would have also been configured to activate the usage protector 250 (“*authentication software*”) as part of its control and configuration of the isolated execution mode and/or the isolated area 70.

[16j] wherein the authentication software is run in a secure processing environment inaccessible to said operating system, for providing the digital signature to the second transaction party.

151. See *supra* [1k], [11].

N. Claim 17

[17pre-1] Method for performing an electronic transaction between a first transaction party and a second transaction party using an electronic device operated by the first transaction party,

152. Claim 17 is rendered obvious by EMC. See *supra* [1pre-1].

[17pre-2] the electronic device having an operating system creating a run-time environment for user applications, the method comprising:

153. See *supra* [1pre-2].

[17a] providing a private key in a memory of said electronic device, wherein the private key is encrypted,

154. See *supra* [1a], [1b].

[17b] when the private key is stored in said memory, a decryption key for decrypting the private key being incorporated in the electronic device, said decryption key being inaccessible to said user, to any user-operated software and to said operating system,

155. See *supra* [1a]-[1d].

[17c] wherein the private key is decrypted in a secure processing environment inaccessible to said user and to any user-operated software;

156. See *supra* [1f].

[17d] providing authentication software in said electronic device, the private key being accessible to said authentication software;

157. See *supra* [1g], [1h].

[17e] activating the authentication software to generate a digital signature from the private key, wherein the authentication software is run in a secure processing environment inaccessible to said operating system;

158. See *supra* [1j], [1k].

[17f] providing the digital signature to the second transaction party.

159. *See supra* [11].

O. Claim 18

[18pre] An electronic device comprising

160. Claim 18 is rendered obvious by EMC. *See supra* [1pre-1], [16pre].

[18a] a network interface configured for an electronic transaction between a first transaction party, and a second transaction party, wherein the electronic device is operated by the first transaction party;

161. *See supra* [1pre-1], [16a].

[18b] a processor configured for an operating system creating a run-time environment for user applications;

162. *See supra* [1pre-2]-[16b].

[18c] a memory storing a private key, wherein the private key is encrypted, when the private key is stored in said memory,

163. *See supra* [1a], [1b], [16c], [16d].

[18d] a decryption key for decrypting the private key being incorporated in the electronic device said decryption key being inaccessible to said user, to any user-operated software, and to said operating system,

164. *See supra* [1c], [1d], [16e].

[18e] wherein the private key is decrypted in a secure processing environment inaccessible to said user and to any user-operated software; and

165. See *supra* [1f], [16g].

[18f] a memory storing authentication software, the private key being accessible to said authentication software;

166. See *supra* [1g], [1h], [1i], [16h].

[18g] wherein the processor is configured for activating the authentication software to generate a digital signature from the private key, wherein the authentication software is run in a secure processing environment inaccessible to said operating system, for providing the digital signature to the second transaction party.

167. See *supra* [1j], [1k], [1l], [16i], [16j].

P. Claim 19

[19] The electronic device according to claim 18, wherein the electronic device is a personal computer, a cellular phone, a hand-held personal digital assistant with wireless communication capabilities, or a device containing a BIOS.

168. See *supra* [1pre-1]. Muir's electronic devices may include, for example, cellular phones, personal data assistants, and smart cards. SAMSUNG-1008, pp.

11-¶3, 8-¶1, 1-¶¶2-3.

VIII. THE '480 CLAIMS DO NOT HAVE WRITTEN SUPPORT

A. The '480 Patent does not describe a private key in the manner recited in the claims

169. As I noted above in Section V.B, to get the application allowed, Patent Owner amended the claims to replace “authentication data” with “private key.” SAMSUNG-1002, 91-96. However, the '480 Patent's specification does not describe the “private key” in the manner that it is claimed. The '480 Patent discloses that the authentication software may have its own private encryption key. SAMSUNG-1001, 5:39-40. In addition to this sentence, *only one paragraph* (13:31-44, reproduced below) in the '480 Patent mentions a “private key.”

In a further embodiment, the authentication software may be provided with a system for signing a digital signature according to the present invention using at least a part of a software identification number (software ID) or any other application identifying character string, hereinafter referred to as a private key. The private key is registered at a third party, for example the authentication software provider which supplied said private key. The private key may be used to uniquely sign the digital signature and append the signed digital signature to the digital signature, which is sent to the second transaction party. The second transaction party is not capable of generating the signed digital signature without the private key. The second transaction party only stores the signed digital signature.

170. In contrast to this description of a private key, the claims recite numerous features that are not supported by the paragraph above or the four corners of the '480 Patent's specification. Examples of the private key features recited in claim 1

that are not supported by the specification are highlighted in red below and discussed in more detail below in subsequent sections.

1. A method for performing an electronic transaction between a first transaction party and a second transaction party using an electronic device operated by the first transaction party, the electronic device having an operating system creating a run-time environment for user applications, the method comprising:

providing a private key in a memory of said electronic device, said memory being a secure part of a Basic In Out System or any other secure location in said electronic device, which private key is inaccessible to a user of said electronic device, wherein the private key is encrypted when the private key is stored in said memory, a decryption key for decrypting the private key being incorporated in the electronic device, said decryption key being inaccessible to said user, to any user-operated software and to said operating system, wherein said memory is inaccessible to said operating system of said electronic device, thereby rendering the private key inaccessible to said user, wherein the private key is decrypted in a secure processing environment inaccessible to said user and to any user-operated software;

providing authentication software in said electronic device, the private key being accessible to said authentication software, wherein authentication software is stored in said secure memory inaccessible to said operating system;

activating the authentication software to generate a digital signature from the private key, wherein the authentication software is run in a secure processing environment inaccessible to said operating system;

providing the digital signature to the second transaction party.

1. The '480 Patent does not describe providing a private key in a memory that is a secure part of the BIOS or any other secure location in the electronic device operated by the first transaction party (independent claims 1, 16-18)

171. The '480 Patent discloses a secure area 62 that includes a secure storage location 60, authentication software 54, an authentication table, and a memory location 58 that stores a log file of all actions performed by the BIOS 44.

SAMSUNG-1001, 8:62-9:34, FIG. 3 (reproduced below).

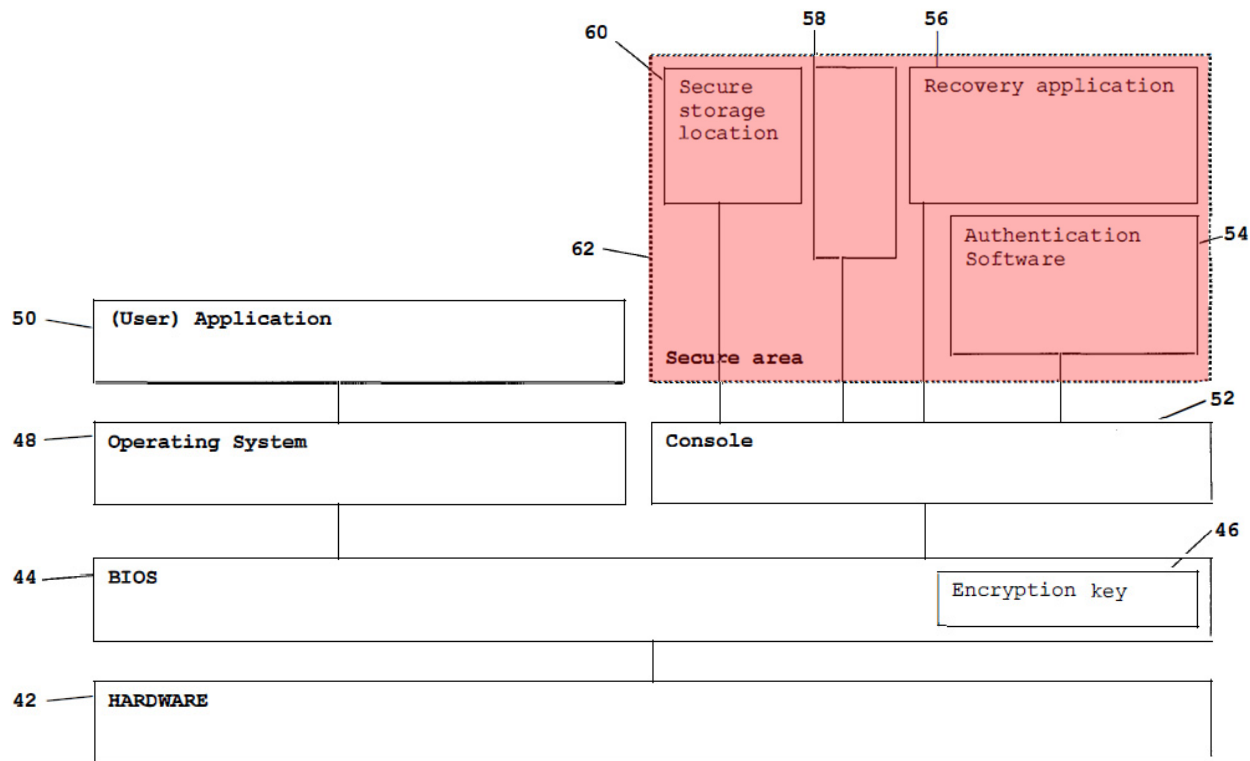


FIG. 3

SAMSUNG-1001 ('480 Patent), FIG. 3

172. Although the '480 Patent explains that the secure area 62 is “[i]n or behind the console 52” and is “only accessible to the BIOS” 44, the '480 Patent fails to describe where the private key is stored in the electronic device and certainly does

not teach that the secure area 62 stores the private key. Claim 1 also recites that the private key is provided in a memory of the electronic device ***that is a secure part of the BIOS or any other secure location*** in the electronic device. Nothing in the '480 Patent describes providing the private key in a memory that is “a secure part of the BIOS or any other secure location” in the electronic device described.

173. The '480 Patent teaches that authentication software for signing a digital signature uses a private key and that “the digital signing algorithm is preferably stored in a secure part of the BIOS 44” and is “protected like the authentication table in the secure storage location 60.” SAMSUNG-1001, 9:27-34, 13:31-44.

However, this disclosure describes the storage of the ***digital signing algorithm*** and ***not of the private key***, which even if used by the digital signing algorithm to generate a digital signature could be obtained from another storage location. The '480 Patent does not teach that the private key is stored in a secure location when running the digital signing algorithm or that the private key can be stored in “any other secure location in said electronic device.” In fact, the '480 Patent only reveals that the private key is “***registered at a third party***, for example the authentication software provider ***which supplied said private key***.” SAMSUNG-1001, 13:31-44. Accordingly, the '480 Patent does not provide written description to support “providing a private key in a memory of said electronic device, said memory being a secure part of a Basic In Out System or any other secure location

in said electronic device,” as recited in claim 1 and similarly in independent claim 16.

174. Claim 17 recites “providing a private key in a memory of said electronic device,” and claim 18 recites “[a]n electronic device comprising: ... a memory storing a private key,”—features that are not described by the ’480 Patent for the same reasons I noted above.

2. The ’480 Patent does not describe the private key being inaccessible to the user (independent claims 1 and 16)

175. As I noted above in Section VIII.A, only *one* paragraph (13:31-44, reproduced below) in the ’480 Patent mentions a “private key.”

In a further embodiment, the authentication software may be provided with a system for signing a digital signature according to the present invention using at least a part of a software identification number (software ID) or any other application identifying character string, hereinafter referred to as a private key. The private key is registered at a third party, for example the authentication software provider which supplied said private key. The private key may be used to uniquely sign the digital signature and append the signed digital signature to the digital signature, which is sent to the second transaction party. The second transaction party is not capable of generating the signed digital signature without the private key. The second transaction party only stores the signed digital signature.

176. This passage does not describe the private key being inaccessible to the user. In fact, since the private key is described as being part of a software ID or an

application identifying character string, it could have been accessible to the user because the '480 Patent does not teach that the software ID or application identifying character string are inaccessible to the user. Moreover, since there is no teaching in the '480 Patent of where the private key is stored or maintained in user's electronic device (*see supra* Section VIII.A.1) or how the private key may be protected from the user when provided by the third party, the '480 Patent fails to provide written description in support of the private key being inaccessible to a user of the electronic device.

177. The description of the authentication software having its own “private encryption key” (*see* SAMSUNG-1001, 5:39-40) also does not provide the needed written description support at least because the '480 Patent does not provide any explanation that such a key is used to perform the operations related to the private key in claim 1 such as being used to generate a digital signature. At best, based on its name, the private encryption key would be understood by a POSITA as an encryption key that is kept private, but not necessarily inaccessible to the user.

3. The '480 Patent does not describe the private key
being encrypted when the private key is stored in the
memory (claims 1, 16-18)

178. The '480 Patent explains that the encryption key 46 in BIOS 44 may be used to encrypt data, but the encrypted data is not a private key. SAMSUNG-1001, 8:29-30. The '480 Patent does not teach “the private key is encrypted when the

private key is stored in said memory” because it does not disclose 1) *a temporal element* that the private key is encrypted *when* it is stored in the memory, and 2) that the private key is stored in the secure memory, as explained above in Section VIII.A.1.

179. The '480 Patent does disclose a “private encryption key.” SAMSUNG-1001, 5:39-40. However, a private encryption key is not the same as an encrypted private key. As I noted above, a private encryption key would have been understood by a POSITA as an encryption key that is kept private. In contrast, an encrypted private key is a private key that has been encrypted— a limitation that is not required by a private encryption key, which may not be encrypted. Rather, the private encryption key is used to encrypt other data, not itself. Thus, the “private encryption key” disclosure in the '480 Patent does not provide written description support for the features related to encryption of the “private key” in the claims.

4. The '480 Patent does not describe the private key being decrypted in a secure processing environment inaccessible to said user, to any user-operated software, and to said operating system (claims 1, 16-18)

180. First, as I noted above in Section VIII.A.1, the paragraph (13:31-44) and sentence (5:39-40) in the '480 Patent that describe a private key do not provide any disclosure related to decryption of the private key.

181. Second, to the extent that the '480 Patent describes a secure processing environment that is inaccessible to said user, to any user-operated software, and to

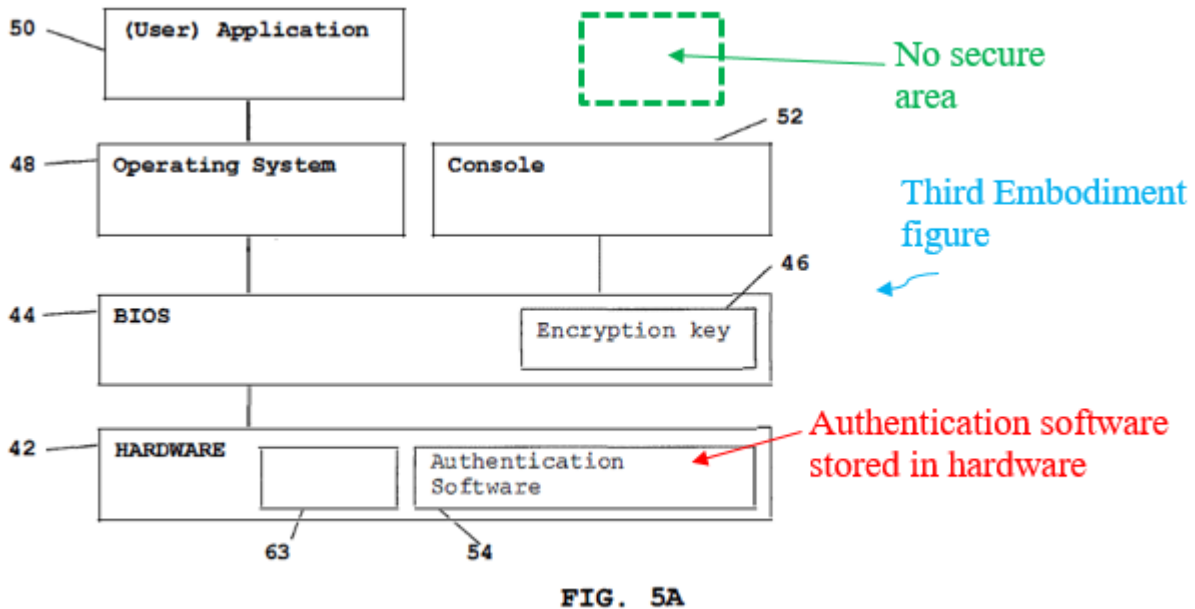
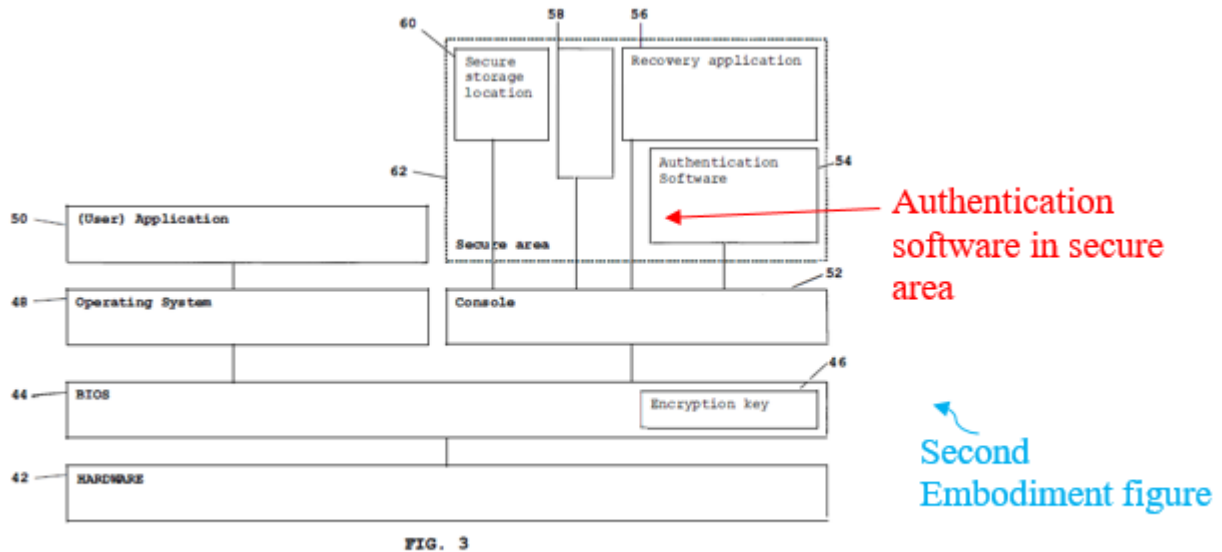
said operating system, such a description is provided with respect to the third embodiment disclosed in the '480 Patent, which is not combinable with the embodiment describing the '480 Patent's second method. SAMSUNG-1001, 9:47-53.

182. For example, as shown in the below snippet of the '480 Patent (11:23-45), the description of the secure processing environment is applicable to the third embodiment in which the authentication data is stored *in a memory that is accessible to an operating system* and possibly to a user of the device.

In a third preferred installation method illustrated in FIG. 4, the authentication data, e.g. authentication table, is stored in a memory that is accessible to an operating system of the device and possibly to a user of said device. To prevent that the user may obtain the authentication data, which should be prevented as described above, the authentication data are encrypted. Thus, the user may only obtain encrypted authentication data. To prevent that the authentication data becomes accessible to a user, the authentication data should only be decrypted in a secure processing environment such as a 'Ring Zero' processing environment, which is embedded in a processing unit of commonly used personal computers. Such a secure processing environment may only be accessible to selected software applications, preferably not to user-operated software applications. Further, a decryption algorithm and/or a decryption key should not be obtainable to any user.

SAMSUNG-1001 ('480 Patent), 11:23-45

183. In contrast, the '480 Patent claims recite “said *memory is inaccessible to said operating system* of said electronic device” which is described with respect to the embodiment illustrated in FIG. 3 and in 8:12-10:2. A comparison of FIG. 3 (second embodiment) and FIG. 5A (third embodiment) immediately reveals to those skilled in the art that the system architecture of these two embodiments are not combinable.



SAMSUNG 1001 ('480 Patent), FIGS 3 (top), 5A (bottom)

184. For example, the second method embodiment (FIG. 3) has a secure area behind the console 52 which the third method embodiment (FIG. 5A) lacks. In the third method embodiment, the authentication software 54 is stored in hardware 42 and accessible to the operating system and the user. In contrast, in the second

method embodiment, the authentication software 54 is stored in a secure area—a key part of all of the '480 Patent independent claims. Thus, the third embodiment directly contradicts the features recited in all the '480 Patent independent claims and cannot provide support for the claims.

B. The '480 Patent does not describe claim 2

185. Claim 2 recites, in part, “the private key is provided by the second transaction party, which stores the private key together with data identifying the first transaction party.” SAMSUNG-1001, 18:30-34.

186. The '480 Patent discloses that “[t]he private key is registered *at a third party*” and that “[t]he second transaction party is not capable of generating the signed digital signature without the private key. *The second transaction party only stores the signed digital signature*” and is forced to communicate with the third party to verify the digital signature because it does not have the private key. SAMSUNG-1001, 13:31-56. Because the second transaction party is not the third party and does not store the private key (and must therefore communicate with the third party), the '480 Patent clearly fails to provide written description support of claim 2’s requirement that the private key be provided by the second transaction party, much less stored by the second transaction party together with data identifying the first transaction party.

187. Moreover, one skilled in the art would have understood that storing the private key at both the first transaction party (claim 1) and the second transaction party (claim 2 which depends from claim 1) would not make sense because then both parties of a transaction would have the private key. Specifically, providing the private key to both parties defeats the purpose of having a “private” key and can prevent the first transaction party from being able to perform its encryption and decryption securely. SAMSUNG-1021, 1:41-45. Thus, the features of dependent claim 2 are inconsistent with the features of claim 1 and are not described in the specification of the ‘480 Patent. Further, note that the specification specifically states that “[t]he second transaction party is not capable of generating the signed digital signature without the private key. The second transaction party only stores the signed digital signature.” ‘480 Patent, 13:40-44. Claim 2 requires that the private key is *provided* by the second transaction party, yet the disclosure just cited makes it clear the second transaction party is not in possession of the private key, making it unclear how the second transaction party could provide an object it does not possess. If the second transaction party somehow has an encrypted or transformed version of the private key that is provided to the first transaction party, this concept and the methods required to make it work are not found anywhere in the specification of the ‘480 patent.

C. The '480 Patent does not describe claims 4 and 5

188. Claim 4 recites, in part, “wherein the private key is encrypted by the second transaction party using an encryption key before the private key is provided to the first transaction party.” SAMSUNG-1001, 18:37-40.

189. As I noted above in Section VIII.B, in the '480 Patent, the second transaction party is forced to communicate with the third party to verify the digital signature because it does not have the private key. SAMSUNG-1001, 13:31-56. Accordingly, claim 4 fails the written description requirement because the second transaction party does not encrypt the private key or provide it to the first transaction party.

190. Moreover, having two parties of a transaction that have knowledge of a private key would run contrary to the goals of a secure exchange. If both parties have possession of a private key, the party for whom the private key is not associated with, could use the private key in an unauthorized manner—which is why private keys are not provided from one party to another in the manner claimed by the '480 Patent. SAMSUNG-1021, 1:41-45. As discussed above in Section VIII.B, the patent specification teaches that the second transaction party is unable to generate the signed digital signature. If the second transaction party were to be in possession of the private key, it of course *could* generate a signed digital signature, contradicting the plain teaching of the specification.

191. Claim 5 also fails the written description requirement because it depends on claim 4 and its features do not cure the deficiencies in claim 4.

D. The '480 Patent does not describe claims 16, 18, and 19

192. In addition to the reasons I noted above in Section VIII.A.1, the '480 Patent fails to provide written description support for claims 16, 18, and 19 (by virtue of its dependency on claim 18) for the follow reasons.

193. Claim 16 recites, in part:

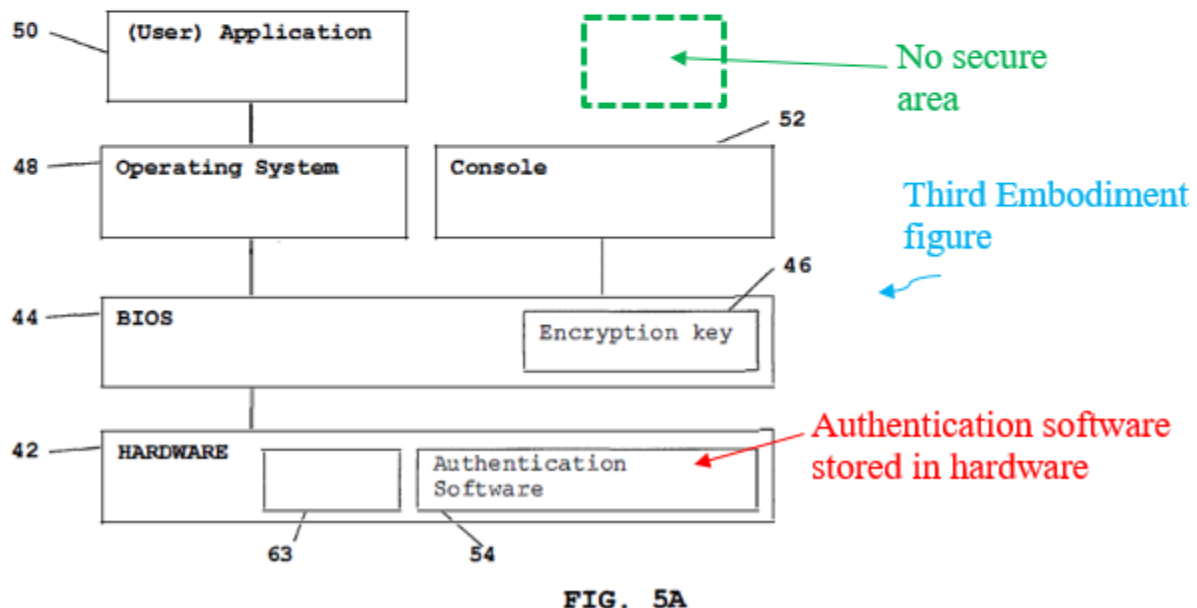
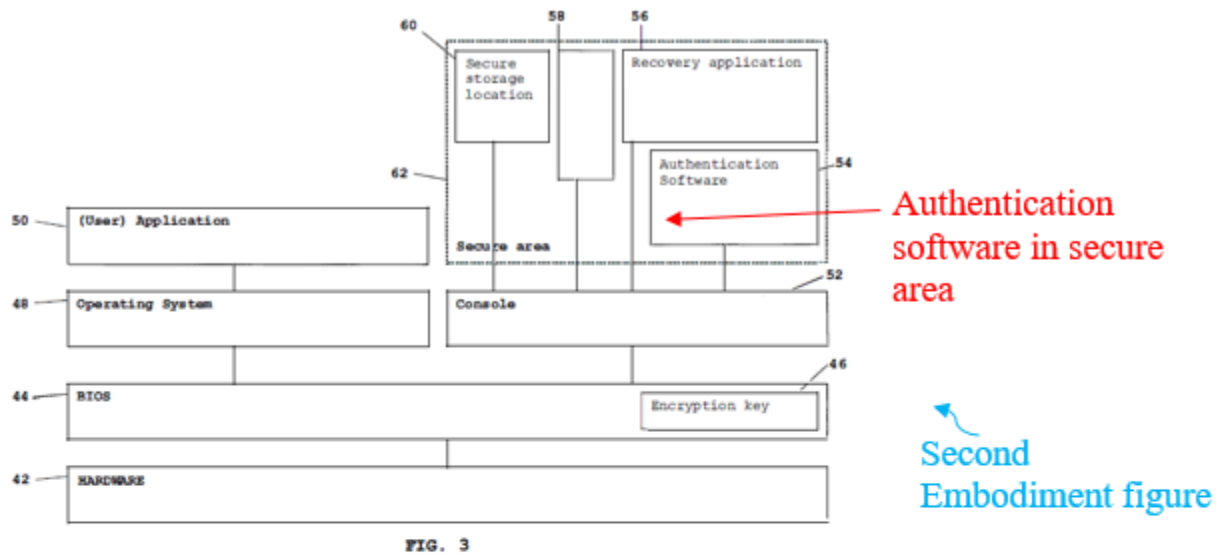
a processor configured for an operating system creating a
run-time environment for user applications;

...

wherein the processor is configured for activating the
authentication software to generate a digital signature
from the private key, wherein the authentication soft-
ware is run in a secure processing environment inac-
cessible to said operating system, for providing the
digital signature to the second transaction party.

194. However, the '480 Patent does not disclose a processor that is configured for an operation system **and** for activating the authentication software ... wherein the authentication software is run in a secure processing environment inaccessible to said operating system. Recall from Section VIII.A.4, the authentication software is in a secure processing environment in the second method embodiment shown in FIG. 3, not in the third method embodiment. However, in the second method embodiment, the processor is **not** configured for activating the authentication

software. SAMSUNG-1001, FIG. 3, 8:12-9:34. The processor is configured for activating the authentication software in the third method embodiment, but that embodiment is *not* compatible with the second method embodiment for the reasons I noted above in Section VIII.A.4.



SAMSUNG 1001 ('480 Patent), FIGS 3 (top), 5A (bottom)

Thus, claim 16 (and similarly claim 18 which recites similar features) are not supported by the '480 Patent because these claims attempt to capture features from two different embodiments, which are not combinable in the manner claimed by the '480 Patent. For instance, a processor/hardware 42 that is forbidden from accessing the secure area 62 or contents therein such as the authentication software 54 cannot be combined with a processor/hardware 42 that can access an authentication software with no secure area. Indeed, the same processor is claimed as being configured for both the operating system and the authentication software. Yet, the authentication software is “stored in a secure memory *inaccessible to said operating system*” and “run in a secure processing environment *inaccessible to said operating system.*” A POSITA would have found it unusual and perhaps inoperative to use of the same processor to handle operating system functionality and functionality of authentication software that is *inaccessible* to the operating system, and such a processor is certainly not described in the '480 Patent.

E. The priority applications and original claims do not cure the written description deficiencies in the '480 Patent

195. The '480 Patent is a continuation of U.S. Patent Application No. 10/560,579, filed as PCT application No. PCT/NL2004/000422 (“SAMSUNG-1014”) on June 14, 2004, and now issued as U.S. Patent No. 11,063,766. SAMSUNG-1001, cover; SAMSUNG-1012, cover. Counsel has informed me that the European Patent Office and the Hague Court have found that the PCT/NL2004/000422

disclosure (which supports European patents EP 1634140 (“EP 140”), EP 3229099 (“EP 099”), and the ’480 Patent) fails to provide support for the claimed features that are similarly recited across all three patents.

196. PCT/NL2004/000422 also claims priority to PCT application PCT/NL2003/000436 (“SAMSUNG-1013”), filed on June 13, 2003. *Id.* The parent PCT/NL2003/000436 application has a more limited disclosure and does not include FIGS. 4, 5, and 10 of the ’480 Patent. SAMSUNG-1013. A table showing the link between the figures of the PCT/NL2003/000436 application and the ’480 Patent is provided below.

PCT/NL2003/000436	US
FIGS. 1-3	FIGS. 1-3
FIG. 4	FIG. 6
FIG. 5	FIG. 7
FIG. 6	FIG. 8
FIG. 7	FIG. 9
---	FIGS. 4, 5, 10

197. The parent PCT/NL2003/000436 application lacks the description of FIGS. 4, 5, and 10 and also does not include the sole paragraph included in the ’480 Patent that discloses a private key. As I explained in detail in Section VIII.A,

claim features related to the private key are not supported by the '480 Patent disclosure. Because disclosure in the parent PCT/NL2003/000436 application is a subset of the disclosure in the '480 Patent and does not provide any additional description related to the claimed features found to lack written support in Europe and in the '480 Patent, PCT/NL2003/000436 also fails to cure the written description deficiencies in the '480 Patent.

198. The original claims submitted during prosecution of the application corresponding to '480 Patent do not recite private key. SAMSUNG-1002, 743-748. The claims were amended by replacing “authentication data” with “private key” in order to reach allowance. *See supra* Section V.B. Thus, the original claims do not provide the required written description support for the issued claims.

IX. CONCLUSION

199. For all the reasons I have noted in the foregoing paragraphs, the Challenged claims of the '480 Patent are obvious in view of the references discussed above.

APPENDIX A

John R. Black, Jr.

Department of Computer Science
430 UCB
University of Colorado
Boulder, CO 80309-0430 USA

office: +1 303 492-0573
FAX: +1 303 492-2844
secretary: +1 303 492-7514

Email: jrblack@cs.colorado.edu
WWW: <http://www.cs.colorado.edu/~jrblack/>

Position	Assoc. Prof. Computer Science, University of Colorado at Boulder	<i>7/08–present</i>
Research	Cryptography, Network Security, Computer Security.	
Past Employment	SecureSet LLC Vice President of Education University of Colorado at Boulder Assistant Professor of Computer Science. University of Nevada, Reno Assistant Professor of Computer Science. University of California, Davis Research Assistant University of California, Davis Teaching Assistant Ingres Corporation Senior Member of Technical Staff	<i>9/15–7/18</i> <i>7/02–7/08</i> <i>7/00–6/02</i> <i>7/97–7/00</i> <i>8/95–6/97</i> <i>6/88–4/94</i>
Education	University of California, Davis Ph.D. in Computer Science. Thesis: Message Authentication Codes. Advisor: Phillip Rogaway. California State University at Hayward B.S. in Computer Science and Mathematics, 1988. Honors: Summa Cum Laude	<i>9/95–9/00</i> <i>9/84–6/88</i>
Awards	NSF CAREER Award, 2002 Chancellor's Teaching Fellowship, UC Davis, 1998 Outstanding Teaching Assistant, UC Davis, 1998 Outstanding Teaching Assistant, UC Davis, 1997 A Check for \$2.56 from Don Knuth, 1996	

**Journal
Publications**

1. P. Rogaway, M. Bellare and J. Black, “OCB: A Block-Cipher Mode of Operation for Efficient Authenticated Encryption.” *ACM Transactions on Information and System Security (TISSEC)*, Volume 6, Issue 3, pp. 365–403, August, 2003.
2. J. Black, and P. Rogaway, “CBC MACs for Arbitrary Length Messages: The Three-Key Constructions.” *Journal of Cryptology*, Volume 18, Number 2, pp. 111–132, Spring, 2005.
3. J. Black, “The Impossibility of Technology-Based DRM and a Modest Suggestion.” *Journal of Telecommunications and High-Technology Law —JTHTL*, Volume 3, Number 2, pp. 387–396, Spring, 2005.
4. J. Black, M. Cochran and R. Gardner, “An Analysis of the Internet Chess Club.” *IEEE Security and Privacy*, Volume 4, Number 1, pp. 46–52, January, 2006.
5. J. Black, M. Cochran and T. Shrimpton, “On the Impossibility of Highly-Efficient Blockcipher-Based Hash Functions.” *Journal of Cryptology*, Volume 22, Number 3, pp. 311–329, Fall, 2009.
6. J. Black, P. Rogaway, T. Shrimpton, and M. Stam “An Analysis of the Blockcipher-Based Hash Functions from PGV.” *Journal of Cryptology*, Volume 23, Number 4, pp. 519–545, Fall, 2010.
7. C. Wilks, M. Cline, F. Weiler, M. Diehkans, B. Craft, C. Martin, D. Murphy, H. Pierce, J. Black, D. Nelson, B. Litzinger, T. Hatton, L. Maltbie, M. Ainsworth, P. Allen, L. Rosewood, E. Mitchell, B. Smith, J. Warner, J. Groboske, H. Telc, D. Wilson, B. Sanford, H. Schmidt, D. Haussler, D. Maltbie “The Cancer Genomics Hub (CGHub): overcoming cancer through the power of torrential data.” *Database*, Volume 2014, doi: 10.1093/database/bau093.

**Book
Chapters**

1. J. Black, “Cryptography.” Invited article for the Encyclopedia of Life Support Systems under the auspices of UNESCO. See <http://www.eolss.net>. 14 pages, March, 2004.
2. J. Black, “Authenticated Encryption.” Invited article for the *Encyclopedia of Cryptography and Security*, Springer-Verlag. 12 pages. August, 2005.

**Conference
Publications
(Refereed)**

1. J. Black, C. Martel, and H. Qi, “Graph and Hashing Algorithms for Modern Architectures: Design and Performance.” *Workshop on Algorithm Engineering — WAE ’98*, Saarbrücken, Germany. Second Workshop on Algorithm Engineering, proceedings, pp. 37–48. Full version of this paper available at theory.cs.ucdavis.edu.
2. J. Black, S. Halevi, H. Krawczyk, T. Krovetz, and P. Rogaway, “UMAC: Fast and Secure Message Authentication.” *Advances in Cryptology — CRYPTO ’99*, Lecture Notes in Computer Science, Vol. 1666, Springer-Verlag, pp. 216–233, 1999. Full version and updated version of this paper available at www.cs.ucdavis.edu/~rogaway/umac.
3. J. Black and P. Rogaway, “CBC MACs for Arbitrary-Length Messages: The Three-Key Constructions.” *Advances in Cryptology — CRYPTO 2000*, Lecture Notes in Computer Science, Vol. 1880, Springer-Verlag, pp. 197–215, 2000.
4. P. Rogaway, M. Bellare, J. Black and T. Krovetz, “OCB: A Block-Cipher Mode of Operation for Efficient Authenticated Encryption.” *Eighth ACM Conference on Computer and Communications Security (CCS-8)*, ACM Press, pp. 196–205, 2001.
5. J. Black and P. Rogaway, “Enciphering Finite Sets of Arbitrary Size.” *RSA Data Security Conference, Cryptographer’s Track (RSA-CT)*, Lecture Notes in Computer Science, Vol. 2271, Springer-Verlag, pp. 114–130, 2002.
6. J. Black and P. Rogaway, “A Block-Cipher Mode of Operation for Parallelizable Message Authentication.” *Advances in Cryptology — EUROCRYPT 2002*, Lecture Notes in Computer Science, Vol. 2332, Springer-Verlag, pp. 384–397, 2002.
7. J. Black and H. Urtubia, “Side-Channel Attacks on Symmetric Encryption Schemes: The Case for Authenticated Encryption.” *USENIX Security Symposium — Security ’02*. 10 pages, 2002.
8. J. Black, P. Rogaway, and T. Shrimpton, “Black-Box Analysis of the Block-Cipher-Based Hash-Function Constructions from PGV.” *Advances in Cryptology — CRYPTO 2002*, Lecture Notes in Computer Science, Vol. 2442. 16 pages, 2002.
9. J. Black, P. Rogaway, and T. Shrimpton, “Encryption Scheme Security in the Presence of Key-Dependent Messages.” *Selected Areas in Cryptography — SAC 2002*, Lecture Notes in Computer Science, Vol. 2595, 14 pages, 2002.
10. R. Motwani, J. Breidenbach and J. Black, “Collocated Dataglyphs for Large Message Storage and Retrieval.” *Security, Steganography, and Watermarking of Multimedia Contents VI*, Society for Imaging Science and Technology (I&ST) jointly with International Society for Optical Engineering (SPIE), Vol. 5306, 19 pages, 2004.
11. J. Black, M. Cochran and T. Shrimpton, “On the Impossibility of Highly-Efficient Blockcipher-Based Hash Functions.” *Advances in Cryptology — EUROCRYPT 2005*, Lecture Notes in Computer Science, Vol. 3494, Springer-Verlag, pp. 526–541, 2005.
12. J. Black, M. Cochran and R. Gardner, “Lessons Learned: A Security Analysis of the Internet Chess Club.”, *Annual Computer Security Applications Conference — ACSAC 2005*, Tucson AZ, USA, pp. 220–228, 2005.
13. J. Black and M. Cochran and T. Highland, “A Study of the MD5 Attacks: Insights and Improvements”, *Fast Software Encryption — FSE 2006*, Lecture Notes in Computer Science, Vol. 4047, Springer-Verlag, pp. 262–277, 2006.

**Conference
Publications
(cont.)**

14. J. Black, “The Ideal-Cipher Model, Revisited: An Uninstantiable Blockcipher-Based Hash Function.”, *Fast Software Encryption — FSE 2006*, Lecture Notes in Computer Science, Vol. 4047, Springer-Verlag, pp. 328–340, 2006.
15. J. Black, “Compare-by-Hash: A Reasoned Analysis”, *USENIX Annual Technical Conference — USENIX 2006*, 8 pages, 2006.
16. J. Black and M. Cochran, “MAC Reforgeability”, *Fast Software Encryption — FSE 2009*, Lecture Notes in Computer Science, Vol. 5665, Springer-Verlag, pp. 345–362, 2009.
17. J.H. Huang, J. Black, and S. Mishra, “Security and Privacy in a Sensor-Based Search and Rescue System,” *1st ICST/CREATE-NET International Conference on Ad Hoc Networks — ADHOCNETS 2009*, Vol. 28, Springer.
18. A. Sayler, D. Grunwald, J. Black, E. White, M. Monaco, “Supporting CS education via virtualization and packages: tools for successfully accommodating “bring-your-own-device” at scale,” *SIGCSE 2014*, pp. 313–318, 2014.

**Workshop
Publications
(Non-
Refereed)**

1. J. Black and P. Rogaway, “A Suggestion for Handling Arbitrary-Length Messages with the CBC MAC.” *NIST Symmetric Key Block Cipher Modes of Operation Workshop—2000*, 4 pages, Sep 2000.
2. J. Black and P. Rogaway, “OCB: Proposal to NIST.” *2nd NIST Symmetric Key Block Cipher Modes of Operation Workshop—2001*, 36 pages, Aug 2001.
3. J. Black and P. Rogaway, “PMAC: Proposal to NIST.” *2nd NIST Symmetric Key Block Cipher Modes of Operation Workshop—2001*, 27 pages, Aug 2001.

Selected Talks

1. Data Structures for Fast Graph Algorithms. Presented at the 1997 UC Davis Workshop on Computing, Davis, USA, October 1997. (See Conference Publication #1.)
2. UMAC: Fast and Secure Message Authentication. Presented at CRYPTO '99, Santa Barbara, USA, August 1999. (See Conference Publication #2)
3. CBC MACs for Arbitrary-Length Messages: The Three-Key Constructions. Presented at CRYPTO 2000, Santa Barbara, USA, August 2000. (See Conference Publication #3)
4. A Suggestion for Handling Arbitrary-Length Messages with the CBC MAC. Presented at NIST Symmetric Key Block Cipher Modes of Operation Workshop—2000, October, 2000; also presented at the 2nd NIST Modes Workshop in Santa Barbara, USA, August 2001.
5. Enciphering Finite Sets of Arbitrary Size. Presented at RSA-CT '02, San Jose, USA, February 2002. (See Conference Publication #5)
6. A Block-Cipher Mode of Operation for Parallelizable Message Authentication. Presented at EUROCRYPT 2002, Amsterdam, The Netherlands, May 2002. (See Conference Publication #6)
7. Side-Channel Attacks on Symmetric Encryption Schemes. Presented at USENIX Security 2002, San Francisco, USA, August 2002. (See Conference Publication #7)
8. Practical Cryptography and Autonomic Web Computing. Invited talk at the 47th meeting of the IFIP Working Group 10.4. Rincon, Puerto Rico, January 2005.
9. On the Impossibility of Highly-Efficient Blockcipher-Based Hash Functions. Presented at EUROCRYPT 2005, Aarhus, Denmark, May 2005. (See Conference Publication #11)
10. The Ideal-Cipher Model, Revisited: An Uninstantiable Blockcipher-Based Hash Function. Presented at FSE 2006, Graz, Austria, March 2006. (See Conference Publication #14)
11. Compare-by-Hash: A Reasoned Analysis. Presented at USENIX Technical Conference 2007, Boston, MA, June 2006. (See Conference Publication #15)

Patents

1. T. McSheery, J. Black, S. Nollet, J. Johnson, and V. Jivan. Distributed-Processing Motion Tracking System for Tracking Individually Modulated Light Points. US Patent 6,324,296 B1. November 2001.

Funding

1. NCHIA E-Team Grant. “Entrepreneurship for Undergraduates.” PI: John Black. Period: 2000-2001. Amount: \$6,000.
2. University of Nevada Junior Faculty Research Grant. “Fast, Provably-Secure Cryptography.” PI: John Black. Period: 2001-2002. Amount: \$10,000.
3. NSF CAREER Award. “Highly-Optimized Provably-Secure Cryptography.” PI: John Black. Period: 2002-2007. Amount: \$469,925.
4. NSF NeTS Grant. “NeTS ProWIN: Topology And Routing With Steerable Antennas.” PI: Dirk Grunwald. Co-PIs: John Black, Douglas Sicker. Period: 2005–2008. Amount: \$745,215.
5. NSF Cybertrust Grant. “Cryptography for Constrained Environments.” PI: John Black. Period: 2005–2008. Amount: \$294,887.

Teaching History

ECS 122A — Design and Analysis of Algorithms (UC Davis). Co-taught once with Professor Rogaway; subsequently taught the course independently.

CMPSC 290G — Intro to Cryptanalysis (UCSB).

CS 365 — Discrete Mathematics (UNR).

CS 425 — Software Engineering (UNR).

CS 426 — Senior Projects (UNR). Supervised 11 group projects in topics ranging from fingerprint recognition to audio editing to GUI design.

CS 432 — Computer Networks (UNR). Introduction to low-level networking concepts with an emphasis on network security.

CS 665 — Graduate Analysis of Algorithms (UNR). A typical algorithms course with emphasis on complexity theory.

CS 709 — Modern Cryptography (UNR). A graduate course introducing cryptography and visiting some of the research front.

CS 791G — Computer Network Security (UNR). A seminar course covering various topics related to network security.

CSCI 2270 — Data Structures (CU); Program design, Object orientation, Java, Linked lists, Arrays, Stacks and Queues, Hash tables, Trees, Balanced Binary Trees, Multi-core programming.

CSCI 3104 — Algorithms (CU); Divide-and-conquer, Greedy, Graph Algorithms, NP-Completeness, Quantum Algorithms.

CSCI 3753 — Operating Systems (CU); Scheduling, Virtual Memory, Filesystems, Multi-core systems, Pthreads, Kernel data structures, virtualization, Security.

CSCI 4830 — Network Security (CU); a new course developed to introduce basics of cryptography and network security. Covers SSL, PKI, DDOS attacks, wireless security, buffer overruns, and more.

CSCI 4900 — Solving Puzzles with Computers (CU); a one-unit undergraduate course describing some hard combinatorial puzzles and how computers can be used to attack them.

CSCI 5413 — Ethical Hacking (CU); Network security, nmap, netcat, Kali Linux, Buffer overruns, Format string vulnerabilities, Race Conditions, Web Security, SQL Injections, XSS, CSRF, Wireless.

CSCI 6268 — Foundations of Computer and Network Security (CU); an introductory course covering basic cryptography, cryptographic protocols, attacks, and principles, as well as core network security attack and defense.

CSCI 7000 — Cryptography Seminar (CU); A graduate course introducing basic cryptographic definitions and then making some forays to the research front.

CSCI 7000 — Cryptanalysis Seminar (CU); A graduate course introducing students to cryptanalysis. Differential and linear cryptanalysis, square attack, RSA basics, factoring, protocol errors, lattices, Coppersmith's algorithm.

CSCI 7000 — Quantum Computing (CU); Introduction to quantum circuits, number theory, Shor's Algorithm, Grover's Algorithm.

Graduate Students

Rakhi Motwani, M.S., Completed: Spring 2002.

Scott Fritzinger, M.S., Completed: Summer 2002.

Hector Urtubia, M.S., Completed: Spring 2003.

Hiba Fayoumi, M.S., Completed: Summer 2004.

Mary Hedges, M.S., Completed: Spring 2007.

Joesph Dunn, Ph.D., co-advisor with John Bennett, Completed: Summer 2007.

Martin Cochran, Ph.D., Completed: Spring 2008.

Jared Nishikawa, Ph.D., Completed: Spring 2016.

Undergraduate Students Troy Trimble, University of California at San Diego. REU Student, Summer 2003.
Gagan Sekhon, California State University at Hayward. REU Student, Summer 2003.
Ryan Gardner, University of Colorado at Boulder. REU Student, Summer 2004.
Trevor Highland, University of Texas at Austin. REU Student, Summer 2005.

**External
Service**

Secretary, International Association for Cryptologic Research, 2005–2007.
Program Committee, ACNS 2015.
Program Committee, CT-RSA 2015.
Program Committee, FSE 2014.
Program Committee, CT-RSA 2014.
Program Committee, CT-RSA 2013.
Program Committee, Eurocrypt 2012.
Program Committee, FSE 2011.
Program Committee, PKC 2011.
Program Committee, CANS 2010.
General Chair, CRYPTO 2009.
Program Committee, CRYPTO 2008.
Program Committee, ACNS 2008.
Program Committee, RSA-CT 2007.
Program Committee, ISC 2007.
Program Committee, ACNS 2007.
Program Committee, ACM CCS 2006.
Program Committee, CANS 2006.
Program Committee, ICISC 2006.
Program Committee, SECURE 2006.
Program Committee, ACSAC 2006.
Program Committee, CRYPTO 2005.
Program Committee, SAC 2005.
Program Committee, ICISC 2005.
Program Committee, CANS 2005.
Program Committee, IEEE SISW 2005.
Program Committee, CRYPTO 2004.
Program Committee, EUROCRYPT 2004.
Program Committee, RSA-CT 2003.
NSF CISE Panelist, 2001, 2003, 2005, 2006, 2007, 2009.
Referee for Journal of Cryptography, 1999–2006.
Referee for Software: Practice and Experience, 2005.
Referee for IEEE Communications Magazine, 2005.
Referee for IEEE Transactions on Circuits and Systems I, 2005.
Referee for IEEE Computer, 2005.
Referee for Journal of Computer Security, 2004.
Referee for IEEE Transactions on Information Theory, 2003.
Referee for IEEE Transactions on Computers, 2002.
Reviewer for CRYPTO 1999–2002, SODA 1998, SPAA 2002, Asiacrypt 2004, EURO-CRYPT 2006.

Developed **CryptoStats** web site: an application which tracks publication rates by year, by author, by conference in the two main cryptography conferences. It was heavily used in my community (on average 240 hits per month), 2003–2009.

ACM Programming Contest problem composer, 2003–2007.
ACM Programming Contest site administrator, 2005.

Graduate Student Mixer organizer, CRYPTO 2005.

**Internal
Service**

Chair, Computing Committee, 2010–2011, 2012–2015.

Chair, Space Committee, 2012–2015.

Liaison, Casey Feldman Foundation, 2010–2015.

Member, Departmental Executive Committee, 2003–2005, 2007–2009, 2013–2015.

Member, Executive Committee, Computer and Communications Security Center, 2003–2006.

Member, Departmental Search Committee, 2003–2006, 2012–2014.

Chair, Departmental Search Committee, 2008–2009.

Member, Graduate Committee, 2005–2006.

Developed departmental voting software, now used for all departmental votes and college votes.