IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE PATENT TRIAL AND APPEAL BOARD

CISCO SYSTEMS, INC.,
Petitioner,

v.

LIONRA TECHNOLOGIES LTD.,
Patent Owner.

U.S. Patent No. 7,302,708

Title: ENFORCING COMPUTER SECURITY UTILIZING AN ADAPTIVE LATTICE MECHANISM

Inter Partes Review Nos.: IPR2023-00669 and IPR2023-00670

DECLARATION OF DR. PAUL C. CLARK IN SUPPORT OF PETITIONS FOR *INTER PARTES* REVIEW OF U.S. PATENT NO 7,302,708

Exhibit 1003
Cisco Systems, Inc. v. Lionra Technologies Ltd.

Table of Contents

I. INTRODUCTION 1

II. QUALIFICATIONS 3

III. RELEVANT LEGAL **STANDARDS** 7

IV. OVERVIEW 10

V. BACKGROUND 11

A. Summary of the '708 Patent 11

B. Prosecution History 19

C. Claim Construction 21

i. [10e] Function-1: comparing said access request to said temporal access table to determine if said access request completes a prohibited temporal access pattern for said entity 23

ii. [10e] Function-2: comparing a minimum access level established for said first base node to said predetermined access level.... 27

iii. [10f] Function-1: denies said request if said access request completes a prohibited temporal access pattern for said entity 29

iv. [10f] Function-2: grants said access request only if it does not complete a prohibited temporal access pattern for said entity, and said minimum access level for said first base node does not exceed said predetermined access level 31

v. [11] Function: denies said request if said minimum access level for said first base node exceeds said predetermined access level for said entity 34

vi. [13] Function: identifies within said hierarchy any higher-level nodes that are aggregations comprising said first base node 35

vii. [14] Function: identifies within said hierarchy any nodes that comprise children of any generation of said higher-level nodes that are aggregations comprising said first base node 36

D. Priority Date of the Challenged Claims 37

E. Person of Ordinary Skill in the Art 37

F. State of the Art 38

1. Well-known sensitivity-based (i.e., "access levels") access control methods 39

2. Well-known method of organizing data: aggregation of nodes in a tree hierarchy 44

3. Updating information among nodes 45

4. Well-known use of patterns of behavior/temporal access patterns in computer security and access control 46

VI. IDENTIFICATION OF CHALLENGES 48

A. Challenged Claims 48

B. Statutory Grounds for Challenges 48

VII. IDENTIFICATION OF HOW THE CHALLENGED CLAIMS ARE UNPATENTABLE 49

A. Ground 1: Claims 1-5 and 9-14 Are Obvious Over Gladstone (EX-1004) and Salazar (EX-1009) 49

1. Gladstone 49

a. Analogous Art 49

b. Overview of Gladstone 51

2. Salazar 53

a. Analogous Art 53

b. Overview of Salazar 55

3. Motivation to combine Gladstone and Salazar 58

4. Detailed Application to the Challenged Claims 59

a. Claim 1 59

i. [1pre] A method for secure access to a computer system, comprising the steps of:.....59

ii. [1a] receiving in said computer system a request from an entity with a predetermined access level for access to a first base node representing at least one of an information type and a computer system function; 61

iii. [1b] determining if said access request completes a prohibited temporal access pattern for said entity; 70

iv. [1c] comparing a minimum access level established for said first base node to said predetermined access level;.....74

v. [1d] granting said access request only if it does not complete a prohibited temporal access pattern for said entity, and said minimum access level for said first base node does not exceed said predetermined access level; and.....77

vi. [1e] denying said request if said access request completes a prohibited temporal access pattern for said entity. 78

b. Claim 2 81

i. 2. The method according to claim 1, further comprising the step of denying said request if said minimum access level for said first base node exceeds said predetermined access level for said entity.....81

c. Claim 3 81

i. [3pre] The method according to claim 1, further comprising the steps of: 81

ii. [3a] logically organizing said computer system in the form of a tree hierarchy having a plurality of leaf nodes and higher-level nodes;...81

iii. [3b] defining a plurality of said base nodes as comprising respectively a plurality of leaf nodes of said tree hierarchy; and 86

iv. [3c] defining said higher-level nodes as aggregations of said base nodes. 86

d. Claim 4 86

i. 4. The method according to claim 3 further comprising the step of identifying within said hierarchy any higher-level nodes that are aggregations comprising said first base node. 86

e. Claim 5 87

i. 5. The method according to claim 4, further comprising the step of identifying within said hierarchy any nodes that comprise children of any generation of said higher-level nodes that are aggregations comprising said first base node. 88

f. Claim 9 88

i. [9pre] A method for restricting access to a computer system having a plurality of logical base nodes representing at least one of an information type and a computer system function, and a plurality of higher-level nodes arranged together with said base nodes in the form of a tree hierarchy, comprising the steps of: 88

ii. [9a] receiving in said computer system a request from an entity with a predetermined access level for access to a first base node; 90

iii. [9b] determining if said access request completes a prohibited temporal access pattern for said entity; 90

iv. [9c] comparing a minimum access level established for said first base node to said predetermined access level; 90

v. [9d] granting said access request only if it does not complete a prohibited temporal access pattern for said entity, and said minimum access level for said first base node does not exceed said predetermined access level; and 90

vi. [9e] denying said request if said access request completes a prohibited temporal access pattern for said entity. 90

g. Claim 10.....90

i. [10pre] A secure computer system comprising:.....90

ii. [10a] a plurality of logical base nodes representing at least one of an information type and a computer system function;.....92

iii. [10b] a plurality of higher-level nodes arranged together with said base nodes in the form of a tree hierarchy;.....93

iv. [10c] a computer system interface capable of receiving a request from an entity with a predetermined access level for access to a first base node;.....98

v. [10d] a temporal access table;.....106

vi. [10e] processing means programmed for comparing said access request to said temporal access table to determine if said access request completes a prohibited temporal access pattern for said entity, and for comparing a minimum access level established for said first base node to said predetermined access level; and.....109

vii. [10f] wherein said processing means denies said request if said access request completes a prohibited temporal access pattern for said entity and grants said access request only if it does not complete a prohibited temporal access pattern for said entity, and said minimum access level for said first base node does not exceed said predetermined access level.....119

h. Claim 11.....122

i. 11. The secure computer system according to claim 10, wherein said processing means denies said request if said minimum access level for said first base node exceeds said predetermined access level for said entity. 122

i. Claim 12.....123

i. 12. The secure computer system according to claim 10 wherein said higher-level nodes are aggregations of said base nodes.....123

j. Claim 13.....123

i. 13. The secure computer system according to claim 12 wherein said processing means identifies within said hierarchy any higher-level nodes that are aggregations comprising said first base node. 124

k. Claim 14..... 125

i. 14. The secure computer system according to claim 13 wherein said computer processing means identifies within said hierarchy any nodes that comprise children of any generation of said higher-level nodes that are aggregations comprising said first base node. 125

B. Ground 2: Claim 6 Is Obvious Over Gladstone in View of Salazar and Hada (EX-1005)..... 126

1. Hada 126

a. Analogous Art..... 126

b. Overview of Hada..... 126

2. Motivation to Combine Gladstone, Salazar, and Hada..... 130

3. Detailed Application to the Challenged Claims 130

a. Claim 6 130

i. 6. The method according to claim 5, further comprising the step of updating a minimum required entity access level for any base nodes that comprise children of any generation of said higher-level nodes that are aggregations comprising said first base node. 130

CONCLUSION AND DECLARATION..... I

APPENDIX A – MATERIALS RELIED UPON..... A

APPENDIX B – *CURRICULUM VITAE*..... B

I, Dr. Paul C. Clark, declare as follows:

I. INTRODUCTION

1. I am over the age of eighteen (18) and otherwise competent to make this Declaration.

2. I have been retained by Cisco Systems, Inc. (Petitioner) as an independent, technical expert consultant in this proceeding before the United States Patent and Trademark Office (USPTO). I am being compensated at my rate of $590 per hour for my services. No part of my compensation is dependent on my opinions or on the outcome of this proceeding. I have no financial interest in any of the parties to this proceeding.

3. This Declaration is in support of the Petitions for *inter partes* review involving U.S. Patent 7,302,708 (the "'708 Patent" or "the Patent") (EX-1001), entitled "ENFORCING COMPUTER SECURITY UTILIZING AN ADAPTIVE LATTICE MECHANISM" and listing Vincent Joseph Kovarik as the inventor.

4. I understand that, according to USPTO records, the '708 Patent is currently assigned to Lionra Technologies Ltd. ("Lionra" or "Patent Owner"). I assume for purposes of this proceeding that Lionra is the patent owner.

5. For the purposes of this *inter partes* review, as I discuss later, I have been instructed to assume that the effective filing date of the Claims of the '708

Patent challenged by the Petitioner in this *inter partes* review is no earlier than March 11, 2004, the filing date of the '708 Patent.

6. The '708 Patent is generally directed to computer and information security and more particularly to security measures that enforce security based on logical implementation methods. I am familiar with the technology described in the '708 Patent as of the earliest possible priority date of March 11, 2004.

7. In preparing this Declaration, I have reviewed the '708 Patent (EX-1001) and its prosecution history (EX-1002), as well as the patents and documents cited herein, and I have considered these documents in light of the general knowledge in the art as of March 11, 2004. In formulating my opinions, I have relied upon my experience in the relevant art. I have also considered the viewpoint of a person of ordinary skill in the art ("POSA") in the relevant field, as of March 11, 2004, who I describe below (§V.E). For convenience, the materials I considered in arriving at my opinions are listed in Appendix A.

8. The opinions and comments formulated during this assessment are based on observations and information available at the time of the investigation. The findings presented herein are made to a reasonable degree of scientific certainty. I have made every effort to accurately and completely investigate all areas of concern identified during our investigation.

9. I have been asked to provide my technical expertise, analysis, insights and opinions regarding the '708 Patent and relevant references that form the basis of the arguments set forth in the accompanying Petitions for *inter partes* review of the '708 Patent. As described in detail below, I offer the following opinions in this Declaration:

- A POSA would have found Claims 1-5, and 9-14 are unpatentable over *Gladstone* in view of *Salazar* and a POSA would have been motivated to combine the teachings of these references.
- A POSA would have found Claim 6 are unpatentable over *Gladstone* and *Salazar* in further view of *Hada* and a POSA would have been motivated to combine the teachings of these references.

II. QUALIFICATIONS

10. My qualifications for forming the opinions set forth in this Declaration are summarized here and explained in more detail in my *curriculum vitae*, which is included as Appendix B. Appendix B also includes a list of my publications and patents.

11. I received a Bachelor of Science in Mathematics from the University of California Irvine, a Master of Science in Electrical Engineering and Computer Science from the University of Southern California, and a Doctor of Science in Electrical Engineering and Computer Science from The George Washington

University with concentrations in Computer and Network Security, Graphics and Intellectual Property Law. My doctoral dissertation and professional work for the last 35 years included advanced secure solutions to access control.

12. From the late 1980's to the mid-1990's I was a Senior Security Engineer at Trusted Information Systems. In that role, I participated in the design and implementation of several networked systems providing integrity, authentication, access control and encryption services.

13. Between January 1994 and July 1999, I was Chief Scientist for DynCorp Network Solutions. In that role, I designed and directed the implementation of several cryptographically enabled network systems including the IRS's Secure Submission and Retrieval System. That secure system received three Al Gore Hammer Awards for improving Government.

14. In the mid 1990's I served as a member of the Federal Advisory Committee for Key Management Infrastructure and as Chairman of the Interoperability Working Group for Cryptographic Key Recovery.

15. Also, in the mid 1990's, I served as a Cooperative Research and Development Agreements partner in a joint effort between the National Institute of Standards and Technology and several companies to begin deployment of the key elements of a public key infrastructure.

16. Since 1999, I have been President and Chief Technology Officer of SecureMethods, Inc. and Paul C. Clark LLC. SecureMethods specializes in the design, implementation, and deployment of advanced network applications for commercial and government clients, including the United States Department of Defense ("DoD.") SecureMethods provides a comprehensive scalable, COTS-based secure network architecture, implemented through the use of the SM Gateway. The SM Gateway is a next-generation security appliance developed by SecureMethods that is available on UNIX-based platforms using commercial, government, and Type I cryptography, implemented in both hardware and software. In my capacity as President and Chief Technology Officer of SecureMethods, I have technical and operational oversight of all projects and corporate technical operations. I provide guidance to senior technical personnel relating to design, implementation, and troubleshooting for a wide range of systems both internal and external. My work includes network systems and security, cryptographic applications, certification, key management, authentication, and integrity strategies for network applications. My firm specializes in complex software and hardware systems for commercial and DoD clients.

17. I have published several articles on network security and encryption, including "BITS – A Smartcard Protected Operating System," with Lance Hoffman, Communications of the ACM, November 1994; "Service Layering

Promotes Secure Data Exchange in Diverse Environments," Computer News, October 23, 1995; and "Threats Posed to Cryptographic Applications by Random Numbers," presented to the RSA Data Security Conference, January 1996.

18. I have also given several presentations at technical conferences relating to security architecture including wireless access, and key distribution protocol topics concerning the subject matter of the '708 patent.

19. I have also been called to provide expert testimony before Congress on issues related to encryption, access control and secure cellular technology, and asked to speak at various conferences and consult on topics related to the '708 Patent long before its priority date.

20. My academic and professional background are closely related to the subject matter of the '708 Patent, and include extensive experience with methods of computer security access control models and associated data. I have served as an adjunct professor in the Electrical Engineering and Computer Science Department at The George Washington University, teaching doctoral level cryptography and security courses. Since receiving my doctorate in 1994, I have worked in the computer and networking field specializing in the design, implementation, and deployment of advanced secure network applications for commercial, Department of Defense, and government clients.

21. I am the named inventor on five security related U.S. patents: 5,448,045; 5,892,902; 8,695,066; 9,391,957 and 10,129,214.

22. Based on my experience described above, and as indicated in my *curriculum vitae*, I am qualified to provide the following opinions with respect to the '708 Patent. Additionally, I am at least a person having ordinary skill in the art as of the priority date of the '708 Patent.

III. RELEVANT LEGAL STANDARDS

23. I do not provide any legal opinions in this Declaration. I have been informed and understand that certain legal standards are to be applied by technical experts in forming opinions regarding the meaning and validity of patent claims. I have been asked to provide my opinions regarding whether the claims of the '708 Patent would have been obvious to a person having ordinary skill in the art at the time of the alleged invention, in light of the prior art.

24. Further, I have been informed and understand that a patent claim is not patentable as obvious (a requirement I understand to be governed by a statute, 35 U.S.C. § 103) if the differences between the patent claim and the prior art are such that the claimed subject matter as a whole would have been obvious at the time the claimed invention was made to a person having ordinary skill in the relevant art. Obviousness, as I have been informed and understand, is based on the scope and content of the prior art, the differences between the prior art and the

claim, the level of ordinary skill in the art, and, to the extent that they exist, certain objective indicia of non-obviousness.

25. I understand that objective indicia can be important evidence regarding whether a patent is obvious or nonobvious, if it has an appropriate nexus to the claimed invention, i.e., is a result of the merits of a claimed invention (rather than the result of design needs or market-pressure advertising or similar activities). Such indicia include: commercial success of products covered by the patent claims; a long-felt need for the invention; failed attempts by others to make the invention; copying of the invention by others in the field; unexpected results achieved by the invention as compared to the closest prior art; praise of the invention by the infringer or others in the field; the taking of licenses under the patent by others; expressions of surprise by experts and those skilled in the art at the making of the invention; the patentee proceeded contrary to the accepted wisdom of the prior art, and, the contemporaneous development of the subject matter claimed by others.

26. I have been informed that whether there are any relevant differences between the prior art and the claimed invention is to be analyzed from the view of a person of ordinary skill in the relevant art at the time of the invention. As such, my opinions below as to a person of ordinary skill in the art are as of the time of the invention, even if not expressly stated as such; for example, even if stated in the present tense.

27. In analyzing the relevance of the differences between the claimed invention and the prior art, I have been informed that I must consider the impact, if any, of such differences on the obviousness or non-obviousness of the invention as a whole, not merely some portion of it. The person of ordinary skill faced with a problem is able to apply his or her experience and ability to solve the problem and also look to any available prior art to help solve the problem.

28. I have been informed that a precise teaching in the prior art directed to the subject matter of the claimed invention is not needed. I have been informed that one may take into account the inferences and creative steps that a person of ordinary skill in the art would have employed in reviewing the prior art at the time of the invention. For example, if the claimed invention combined elements known in the prior art and the combination yielded results that were predictable to a person of ordinary skill in the art at the time of the invention, then this evidence would make it more likely that the claim was obvious. On the other hand, if the combination of known elements yielded unexpected or unpredictable results, or if the prior art teaches away from combining the known elements, then this evidence would make it more likely that the claim that successfully combined those elements was not obvious.

29. I have been informed and understand that there are recognized, exemplary, rationales for combining or modifying references to show obviousness

of claimed subject matter. Some of the rationales include the following: combining prior art elements according to known methods to yield predictable results; simple substitution of one known element for another to yield predictable results; use of a known technique to improve a similar device (method or product) in the same way; applying a known technique to a known device (method or product) ready for improvement to yield predictable results; choosing from a finite number of identified, predictable solutions, with a reasonable expectation of success; known work in one field of endeavor may prompt variations of it for use in either the same field or a different one based on design incentives or other market forces if the variations are predictable to one of ordinary skill in the art; and some teaching, suggestion, or motivation in the prior art that would have led one of ordinary skill to modify the prior art reference or to combine prior art teachings to arrive at the claimed invention.

IV. OVERVIEW

30. The Claims 1-6 and 9-14 of the '708 Patent (the "Challenged Claims") are unpatentable as obvious over the prior art. The claims are directed to prior art systems and combinations of conventional components that perform a conventional function of identifying/using patterns of behavior including, "temporal access patterns," in determining whether to grant access requests to a computer system, which was well known in computer and information security

before the priority date of the '708 Patent. The '708 Patent admits that the prior art employed "usage patterns" in computer security, but faults the art, incorrectly, for merely "flagging suspicious activity" and not "enforc[ing]" "a security policy based on such flagging." EX-1001 1:46-51; §V.F.4.

V. BACKGROUND

A. Summary of the '708 Patent

31. The '708 Patent is directed to methods and systems for ensuring secure access to computer systems. EX-1001 1:7-11, 1:55-56, 10:16-17, 11:15-16, 11:34. It describes controlling access to various "nodes" of a computer system organized in the form of a "tree hierarchy" as shown in Figure 1:

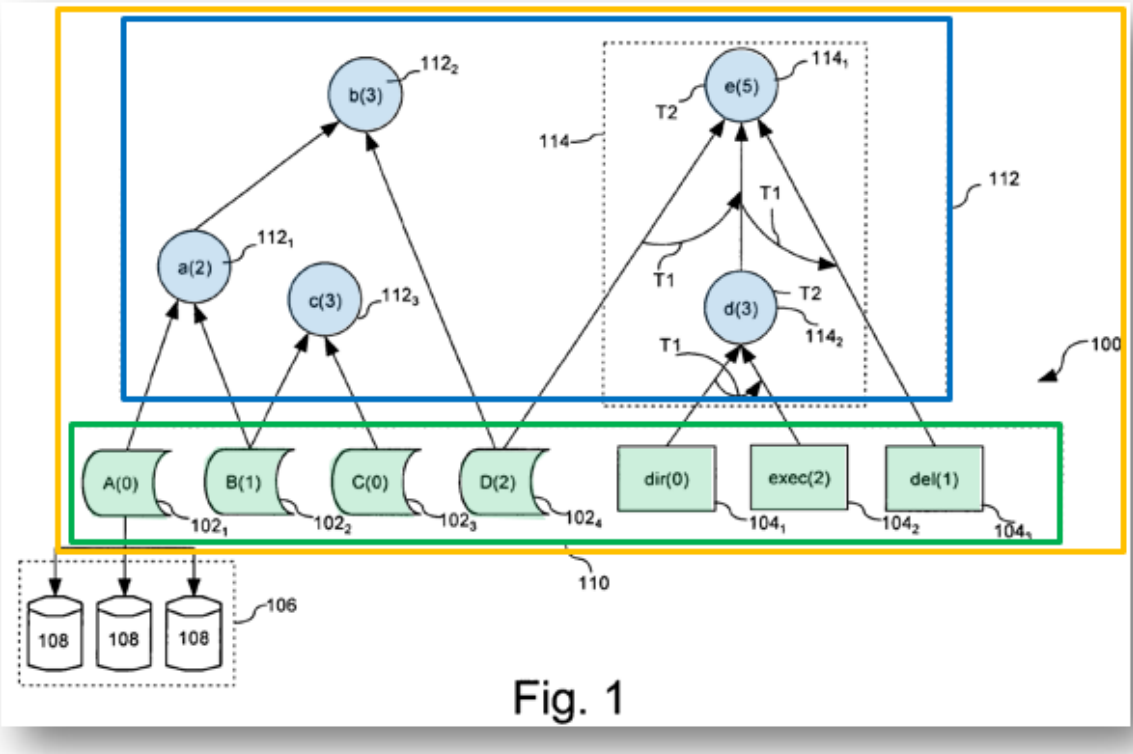


Fig. 1

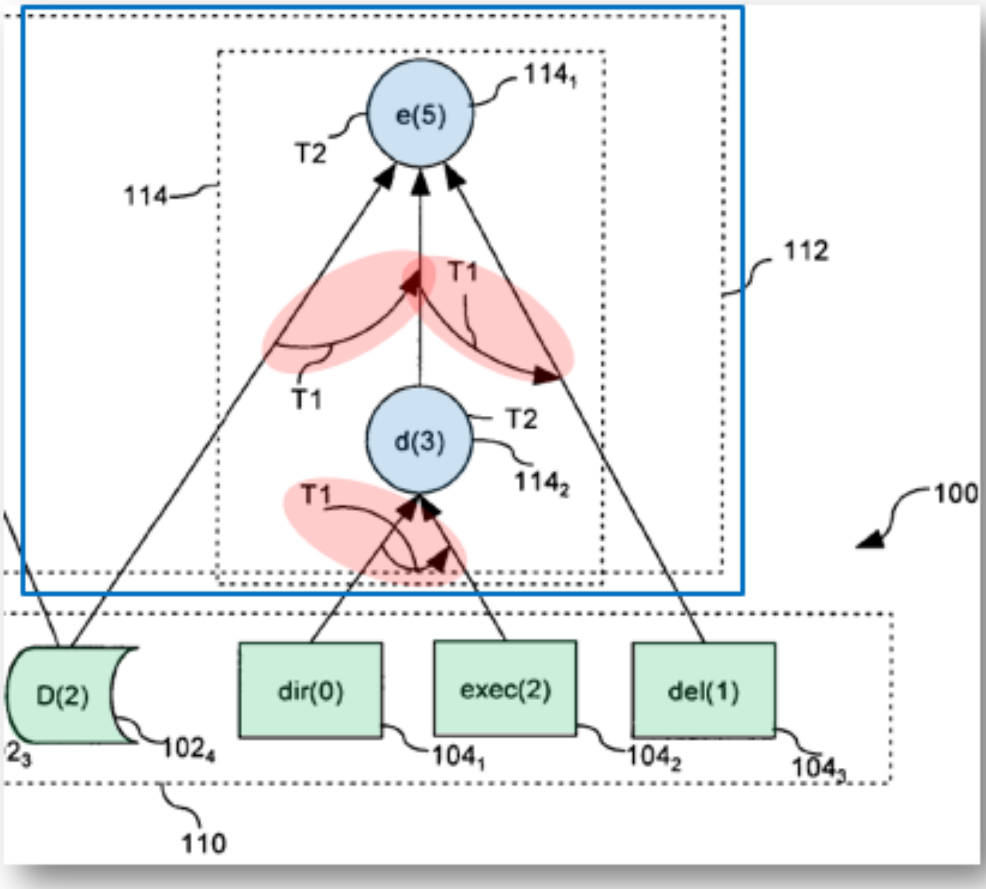
EX-1001 1:59-62, 2:67-3:6, 10:18-21, FIG. 1 (annotated).[1] Figure 1 illustrates two types of nodes: **base (or leaf) nodes 110** and **higher level (or aggregation) nodes 112**. EX-1001 3:44-49. **Base nodes 110** can be formed from "security access primitives" and can represent, or define, "various object types" 102_1-102_4 (*e.g.*, types of information), as well as system functions 104_1-104_3 (*e.g.*, directory (dir), execute (exec), and delete (del) functions). EX-1001 2:67-3:6, 3:10-47. Each node 110 can have a minimum required[2] security access level that is indicated using a parenthetical. In Figure 1, the number "2" in node "**D(2)**" indicates that "information type D" has a minimum required security access level of two. *Id.* If an "access authorization level" or "security access level" of a user making the request to the node is less than the minimum required security access level specified for that node, the request is rejected. EX-1001 1:59-62, 6:4-10.

32. Figure 1 also shows "**higher-level" or "aggregation" nodes 112** that "represent information aggregation sets and/or **temporal patterns of access**" of **base nodes 110** (or other **higher-level nodes**), *e.g.*:

---

[1] Emphasis, including coloring, made in this Declaration is added unless stated otherwise.

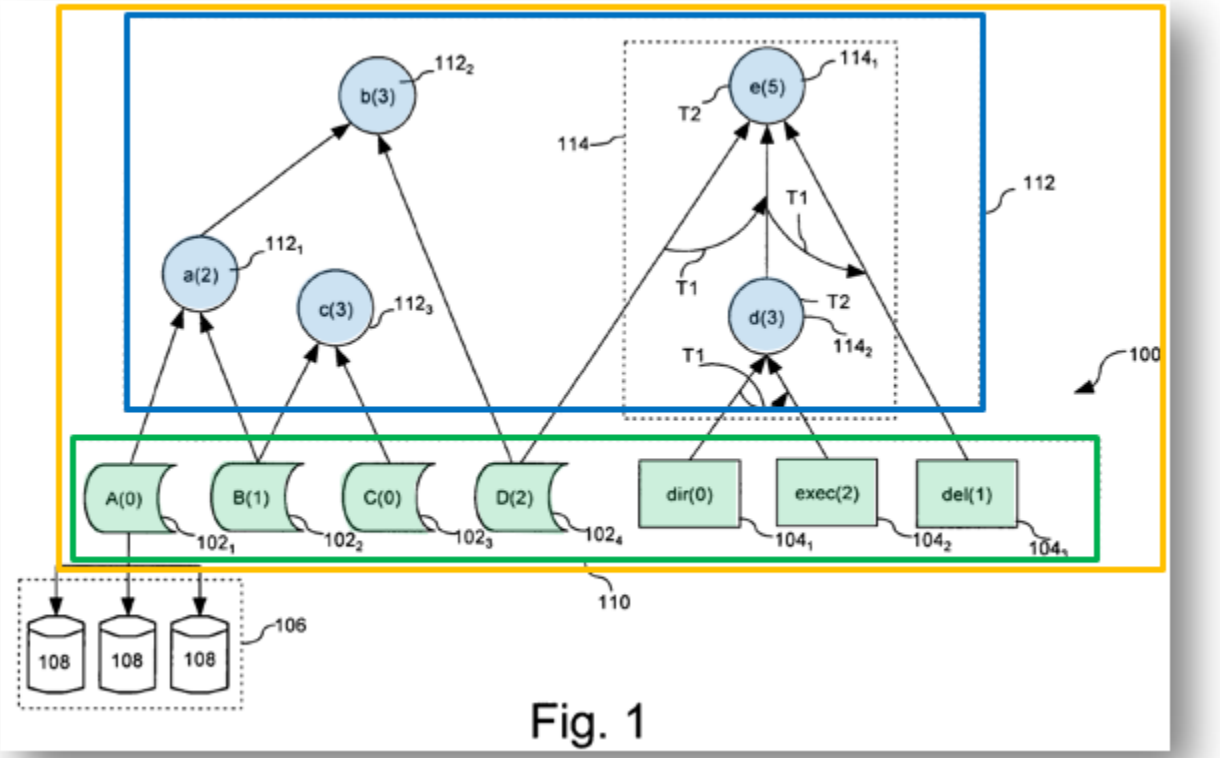
[2] As demonstrated in §§ V.B and V.F.1, this subject matter was well known, even according to the '708 Patent.



EX-1001 3:2-6, FIG. 1 (annotated). **Aggregation nodes 112** are comprised of two or more of **base nodes 110**, other **aggregation nodes 112**, or some combination of both. EX-1001 6:13-16. The '708 Patent states that "**aggregation nodes** will often have a higher minimum required security level" compared to **base nodes 110** "because aggregated data is often of a more sensitive nature." EX-1001 6:16-23. The '708 Patent acknowledges that "[c]onsequently **aggregated information** will inevitably be of greater interest to unauthorized users and system administrators

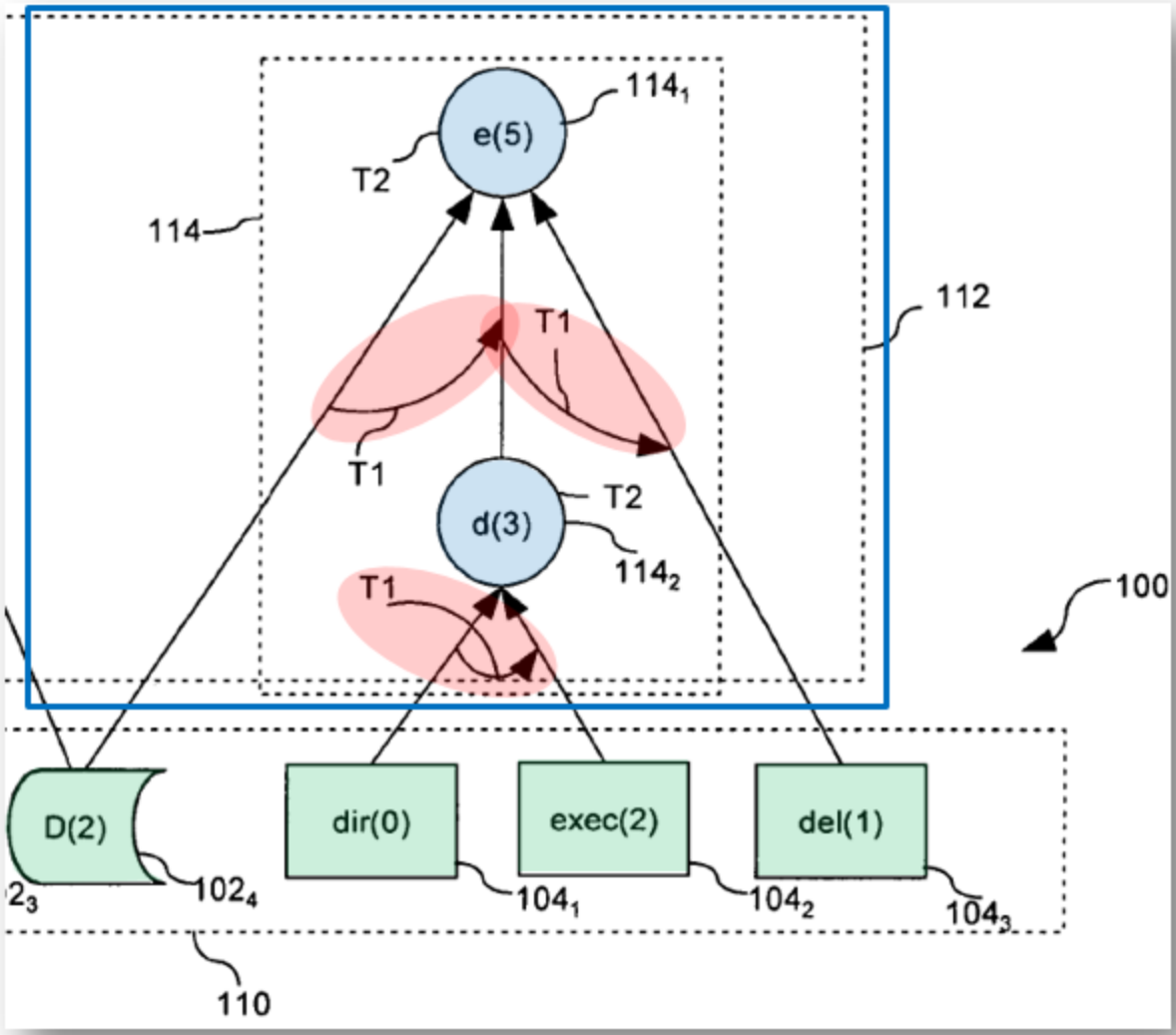
will naturally wish to impose higher level restrictions on its access." EX-1001 6:23-26.

33. The '708 Patent describes that **aggregation nodes 112** and **base nodes 110** are logically organized into "**hierarchical tree 100**":



EX-1001 3:44-49, FIG. 1 (annotated).

34. **Aggregation nodes 112** can represent "**temporal patterns of access**" that can be used to determine whether a user's access request completes a temporal access pattern as part of its access control method. EX-1001 1:62-64, 4:12-16. A temporal access pattern "denote[s] a **temporal ordering**" by which the nodes are accessed. EX-1001 4:3-15, FIG. 1. For example, **higher-level node "d(3)" 114_2** defines temporal ordering **T1** of the functions **104_1** and **104_2:**



Id. For the security policy associated with node **d(3) 114_2** to be activated, not only must both **104_1** and **104_2** be accessed but they must be accessed in the order indicated—*i.e.*, **104_1 before 104_2** (**$104_1 \rightarrow 104_2$**). *Id*.

35. In addition to Figure 1's illustration of a temporal access pattern, the '708 Patent also describes determining whether the order (temporal access pattern) required is met by comparing the time stamps in a Temporal Access Table, an

example of which is provided in Figure 4B, to a Temporal Order Table, an example of which is provided in Figure 3:

Temporal Order Table (TOT)

TOT	A	B	C	D	dir	exec	del
a							
b							
c							
d					1	2	
e				1	2.1	2.2	3

EX-1001 FIG. 3. This Temporal Order Table captures the relative temporal order of actions and allows relative or approximate temporal pattern matching. EX-1001 4:30-33.

Temporal Access Table (TAT)

Usr	A	B	C	D	dir	exec	del
1(1)					102	112	
2(1)							
3(2)					111	103	
4(1)							
5(3)							

EX-1001 FIG. 4B. This Temporal Access Table maintains a history of what operations are performed by a user and when those operations are performed. EX-1001 5:55-57. The order of the performance of these operations may be compared to a defined **temporal access pattern**. EX-1001 5:59-61. As shown in Figure 4B, user 1 performs a **"dir" operation 104_1** at time 102 and then requests an **"exec" operation 104_2** at time 112. Since time 112 is after time 102, this request matches the **temporal access pattern** illustrated for **node "d" 114_2** of Figure 1 and, hence, the security policy associated with **node "d" 114_2** would be evaluated. EX-1001 5:62-66, FIG. 1. In this example, this request would be denied because user 1's access level of "1" (*see* parenthetical following user 1 in Figure 4B, above) is less than the minimum access level required by node d—level "3." *Id*. 5:62-66, FIGs. 1, 4B.

36. The '708 Patent issued with 17 claims. Claims 1 and 9 are independent method claims and Claim 10 is an independent system claim. Claims 1 and 10 are representative:

[1pre]. A method for secure access to a computer system, comprising the steps of:

[1a] receiving in said computer system a request from an entity with a predetermined access level for access to a first base node representing at least one of an information type and a computer system function;

[1b] determining if said access request completes a prohibited temporal access pattern for said entity;

[1c] comparing a minimum access level established for said first base node to said predetermined access level;

[1d] granting said access request only if it does not complete a prohibited temporal access pattern for said entity, and said minimum access level for said first base node does not exceed said predetermined access level; and

[1e] denying said request if said access request completes a prohibited temporal access pattern for said entity.

[10pre] A secure computer system comprising:

[10a] a plurality of logical base nodes representing at least one of an information type and a computer system function;

[10b] a plurality of higher-level nodes arranged together with said base nodes in the form of a tree hierarchy;

[10c] a computer system interface capable of receiving a request from an entity with a predetermined access level for access to a first base node;

[10d] a temporal access table;

[10e] processing means programmed for comparing said access request to said temporal access table to determine if said access request completes a prohibited temporal access pattern for said entity, and for comparing a minimum access level established for said first base node to said predetermined access level; and

[10f] wherein said processing means denies said request if said access request completes a prohibited temporal access pattern for said entity and grants said access request only if it does not complete a prohibited temporal access pattern for said entity, and said minimum access level for said first base node does not exceed said predetermined access level.

37. However, controlling access to computer systems based on the temporal access pattern based on the order of a user's requests was well known in the art and commonly used long before the '708 Patent as demonstrated by this Declaration.

B. Prosecution History

38. I have been informed by counsel that the '708 Patent issued from U.S. 10/798,074, filed on March 11, 2004. Ex-1001 (21, 22, 45); Prosecution history of the '708 Patent ("File History") (Ex-1002). I have been informed by counsel that that the Examiner subjected U.S. 10/798,074 to a single office action, rejecting the

claims under 35 U.S.C. §§101 and 103 over U.S. 2004/0044655 to Cotner ("Cotner"). EX-1002 129-140.

39. I have been informed by counsel that the Applicant addressed the §101 rejection by adding the following limitations:

- Limitations [1e] and [9e]: "denying said request if said access request completes a prohibited temporal access pattern for said entity."
- Limitation [10f]: "wherein said processing means denies said request if said access request completes a prohibited temporal access pattern for said entity and …."

EX-1002 115-119, 120.[3] I have been informed by counsel that the Applicant argued against the §103 rejection, stating Cotner disclosed only "controlling access to a relational database based solely on security levels." EX-1002 123. These "security levels" are the same as the "access levels" described by the '708 Patent. I have been informed by counsel that the Applicant stated that Cotner's "security levels" did not disclose:

[3] I have been informed by counsel that original Claims 1, 10, and 11, to which these amendments were made, issued as Claims 1, 9, and 10, respectively. EX-1002 019.

- "determining if an access request completes a **prohibited temporal access pattern** for an entity,"
- "granting the access request only if it does not complete a **prohibited temporal access pattern** for the entity," or
- "a secure computer system comprising a processing means programmed for comparing an access request to the temporal access table to determine if the access request completes a **prohibited temporal access pattern** for an entity."

Ex-1002 122-124. I have been informed by counsel that the '708 Patent subsequently issued without further substantive comments from the Examiner. EX-1002 003-016.

40. However, as demonstrated by this Declaration, determining temporal access patterns and enforcing security policies based on those patterns was well understood and widely implemented prior to the subject matter claimed in the '708 Patent.

C. Claim Construction

41. I have been advised by counsel that, in order to properly evaluate the claims of the '708 Patent, the terms of the claims must first be interpreted. I have been asked to give each claim term in the Challenged Claims its plain and ordinary meaning in view of the specification and prosecution history as would have been

understood by a person of ordinary skill in the art ("POSA," defined below in §V.E) as of March 11, 2004 in this proceeding. It is my opinion that no specific construction of any claim term is required because the prior art I have relied upon meets each of the claim terms under any reasonable construction.

42. I have been informed by counsel that the Board does not construe claim terms unless the terms are in controversy, and then only to the extent necessary to resolve the controversy.

43. I have been informed by counsel to address the "processing means" term of claim elements [10e-f], and claims 11 and 13-14 to the extent the Patent Owner argues or the Board finds it is governed by 35 U.S.C. § 112 paragraph 6 ("means-plus-function").

44. I have been informed by counsel that when claim language is "governed by means-plus-function," it is given a special meaning in patent law. I am instructed by counsel and understand that a means-plus-function element does not cover all of the structures that could perform the function set forth in the claim. Instead, it covers a structure or a set of structures that performs that function and that is either identical or "equivalent" to at least one of the sets of structure(s) described in the patent specification and that is necessary for performing the claimed function. I have been informed by counsel that structure taught by the prior art, alone or in combination, is an "equivalent" of the corresponding structure

in the patent specification if a POSA either would have considered the differences between them to be insubstantial at the time of the invention or if that person would have found that the structures performed the identical function in substantially the same way to accomplish substantially the same result. I was informed by counsel that in deciding whether the differences would be insubstantial, I should consider whether a POSA would have known of the interchangeability of the structures. I further was informed by counsel that even if the prior art, alone or in combination, is not identical or an equivalent to the corresponding structure(s), that the claimed subject matter is still unpatentable if it would have been obvious to a POSA.

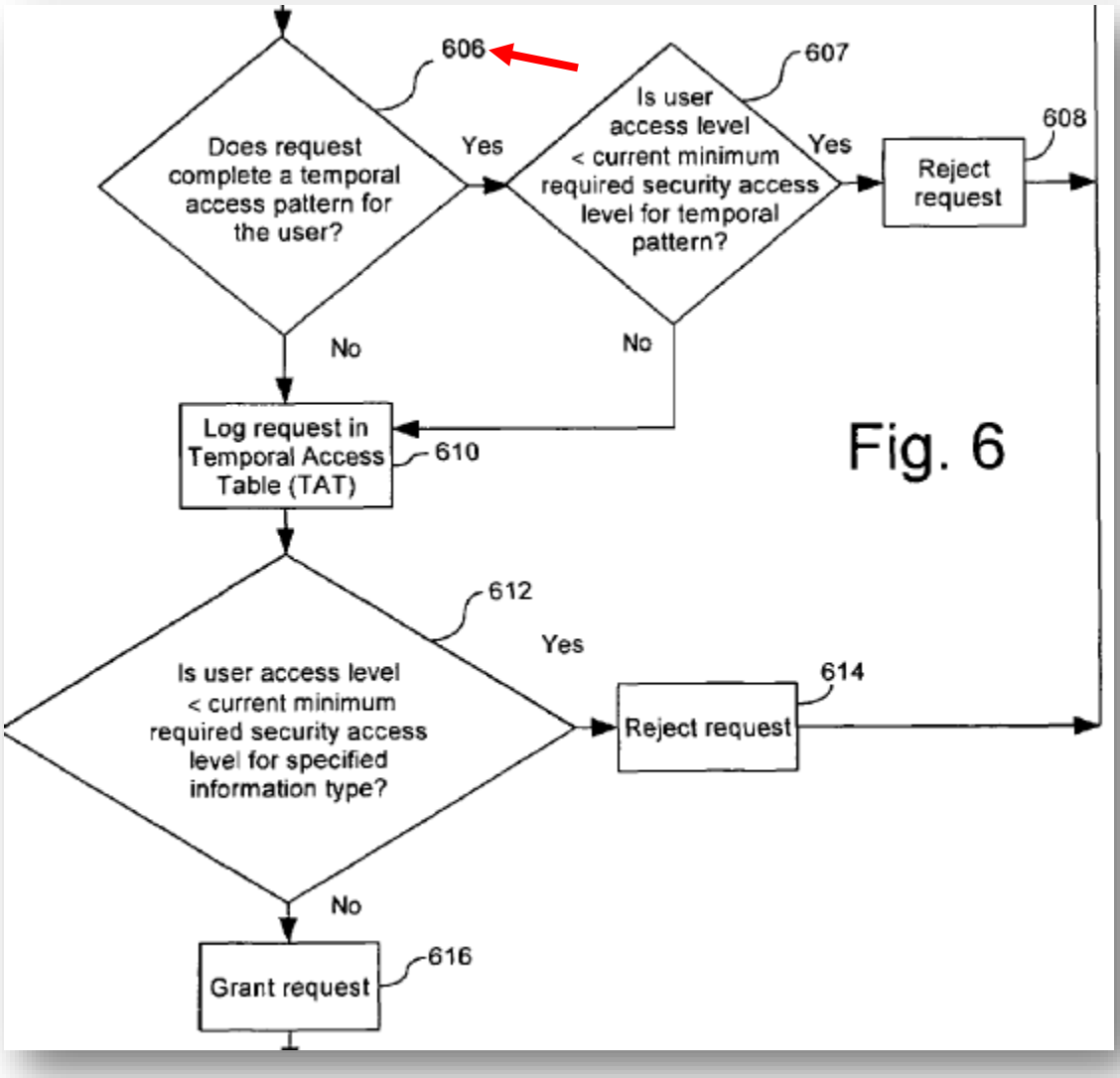
45. I have been instructed by counsel that, for each of the following functions of the "processing means," the recited structure is a general purpose computer system adapted to carry out the algorithm(s) listed in the following sections (EX-1001 9:37-10:9; FIG. 10) and equivalents thereof:

i. [10e] <u>Function-1</u>: comparing said access request to said temporal access table to determine if said access request completes a prohibited temporal access pattern for said entity

46. **<u>Algorithm</u>**:

"The computer system determine if the access request completes a prohibited temporal access pattern for the entity." EX-1001 1:62-64.

"[T]he system continues on to step 606 to determine if the request completes a temporal access pattern for the particular user":



EX-1001 5:33-35, FIG. 6 (annotated).

"If the request does complete a temporal access pattern, this means that the sequence of operations performed matches an identified pattern." EX-1001 5:36-37.

"Note that the temporal ordering of primitive access operations mandates that the operations occur in the order specified for the security policy to be enforced. Thus, in FIG. 1, if 104_2 is accessed first followed by item 104_1, the security policy associated with node d, item 114_2, will not be activated because the temporal order was not satisfied." EX-1001 5:42-47, *see also* FIG. 1.

"As shown in FIG. 4B, the Temporal Access Table maintains a history of the primitive operations performed by a user":

Temporal Access Table (TAT)

Usr	A	B	C	D	dir	exec	del
1(1)					102	112	
2(1)							
3(2)					111	103	
4(1)							
5(3)							

Fig. 4B

EX-1001 5:54-56, FIG. 4B.

"As a user is granted authorization to perform a primitive access operation, the operation is time stamped and stored within the TAT. The time stamps are compared against the temporal patterns identified in the Temporal Order Table, FIG. 3, to check for matches":

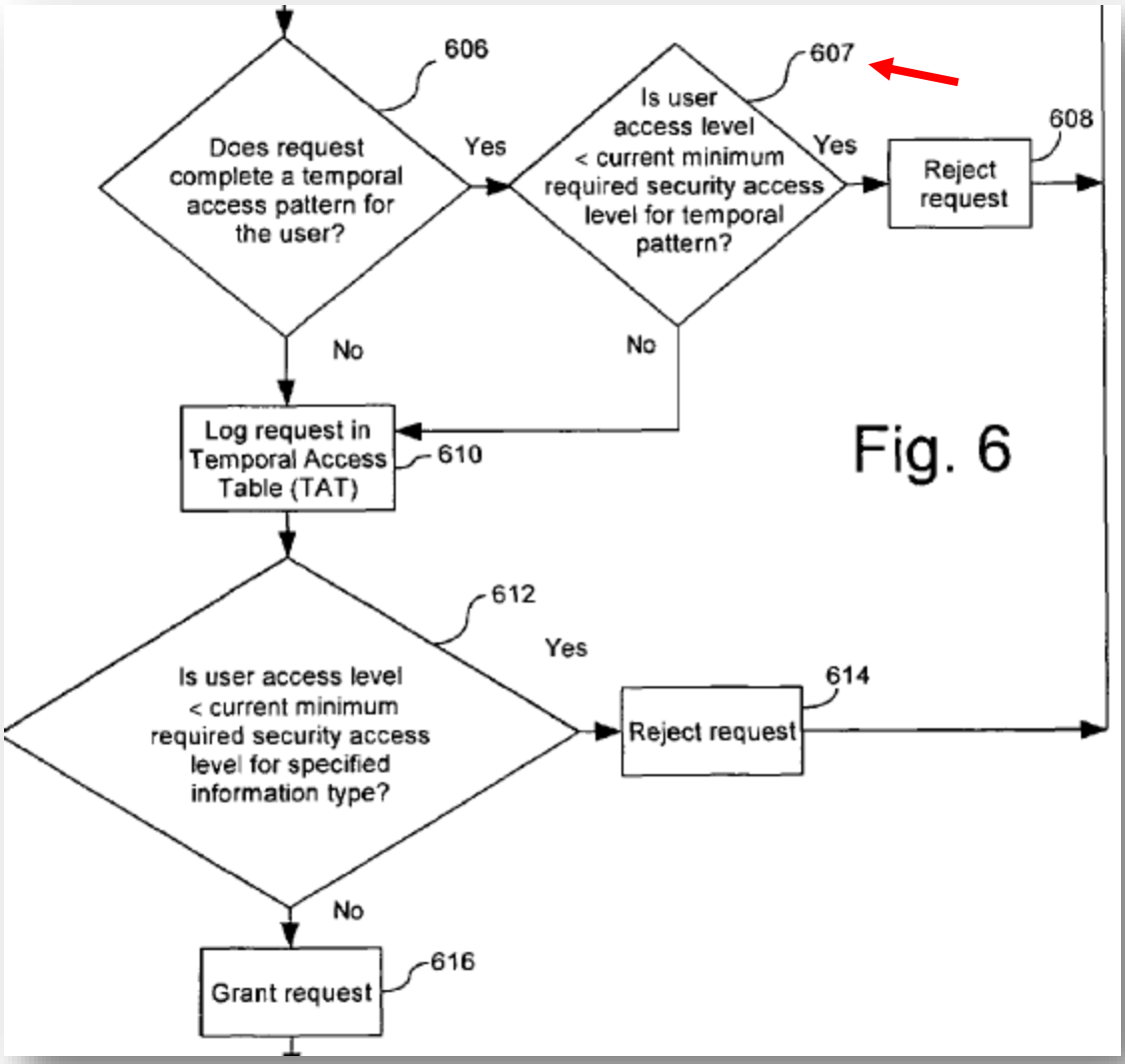
Temporal Order Table (TOT)

TOT	A	B	C	D	dir	exec	del
a							
b							
c							
d					1	2	
e				1	2.1	2.2	3

EX-1001 5:56-60, FIG. 3.

To extent that the *a prohibited temporal access pattern* requires a comparison of *access levels*:

"If a temporal pattern is completed, the user's access level is compared to the access level required for the requested access action, illustrated in step 607":



EX-1001 5:39-42, FIG. 6 (annotated).

"If the access level of the user is not sufficient for the requested action, then the request is rejected in step 608." EX-1001 5:47-49.

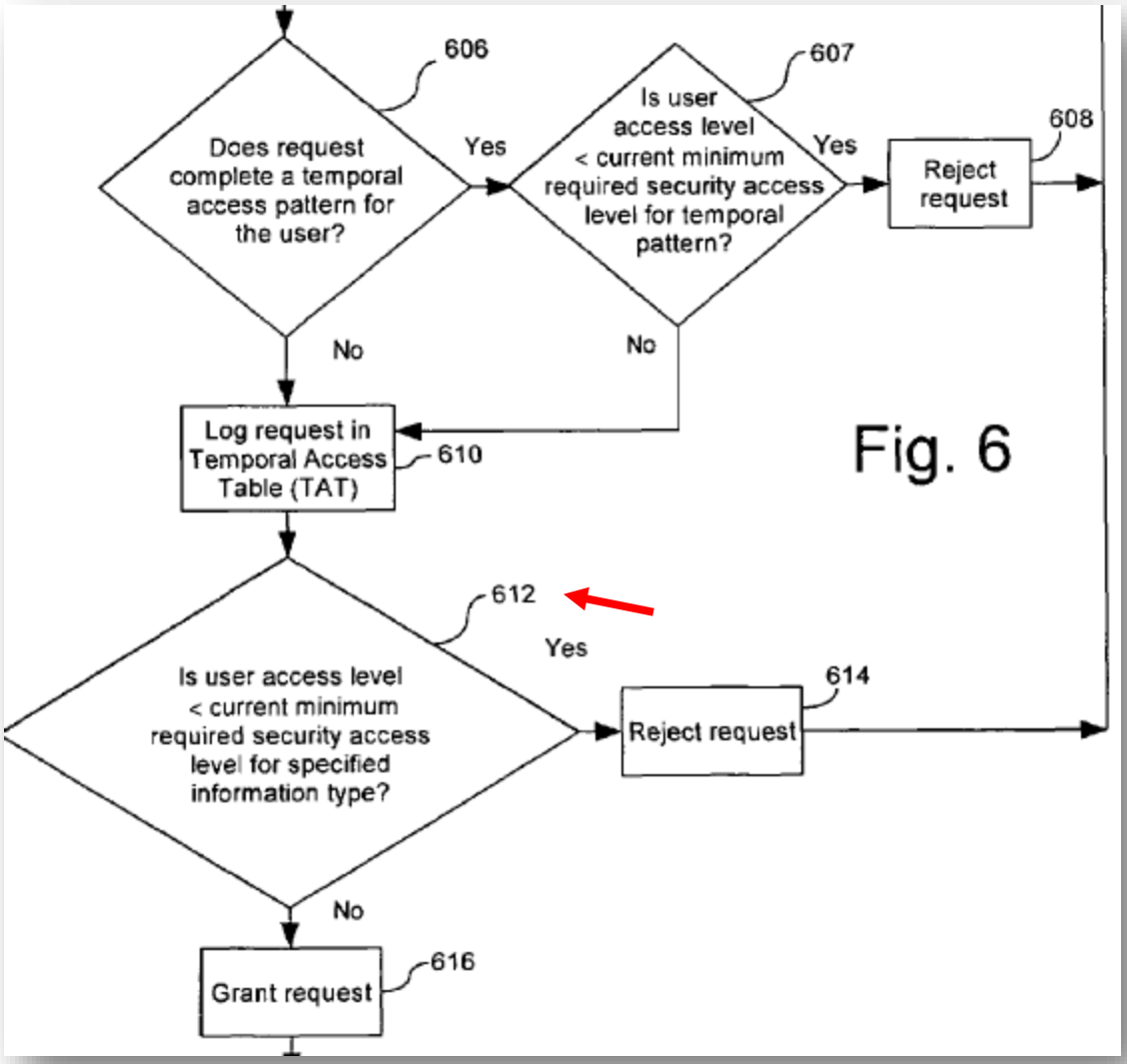
ii. [10e] <u>Function-2</u>: comparing a minimum access level established for said first base node to said predetermined access level

47. **<u>Algorithm</u>**:

"The system also compares a minimum access level established for the first base node to the predetermined access authorization level assigned to the entity." EX-1001 Abstract.

"[T]he system can compare a minimum access level established for the first base node to the predetermined access authorization level assigned to the entity." EX-1001 1:65-67.

"In step 612," of Figure 6 (below), "the computer system makes a determination as to whether the security access level of the user making the request is less than the current minimum required security access level for the specified information type." EX-1001 6:2-8.



EX-1001 FIG. 6 (annotated).

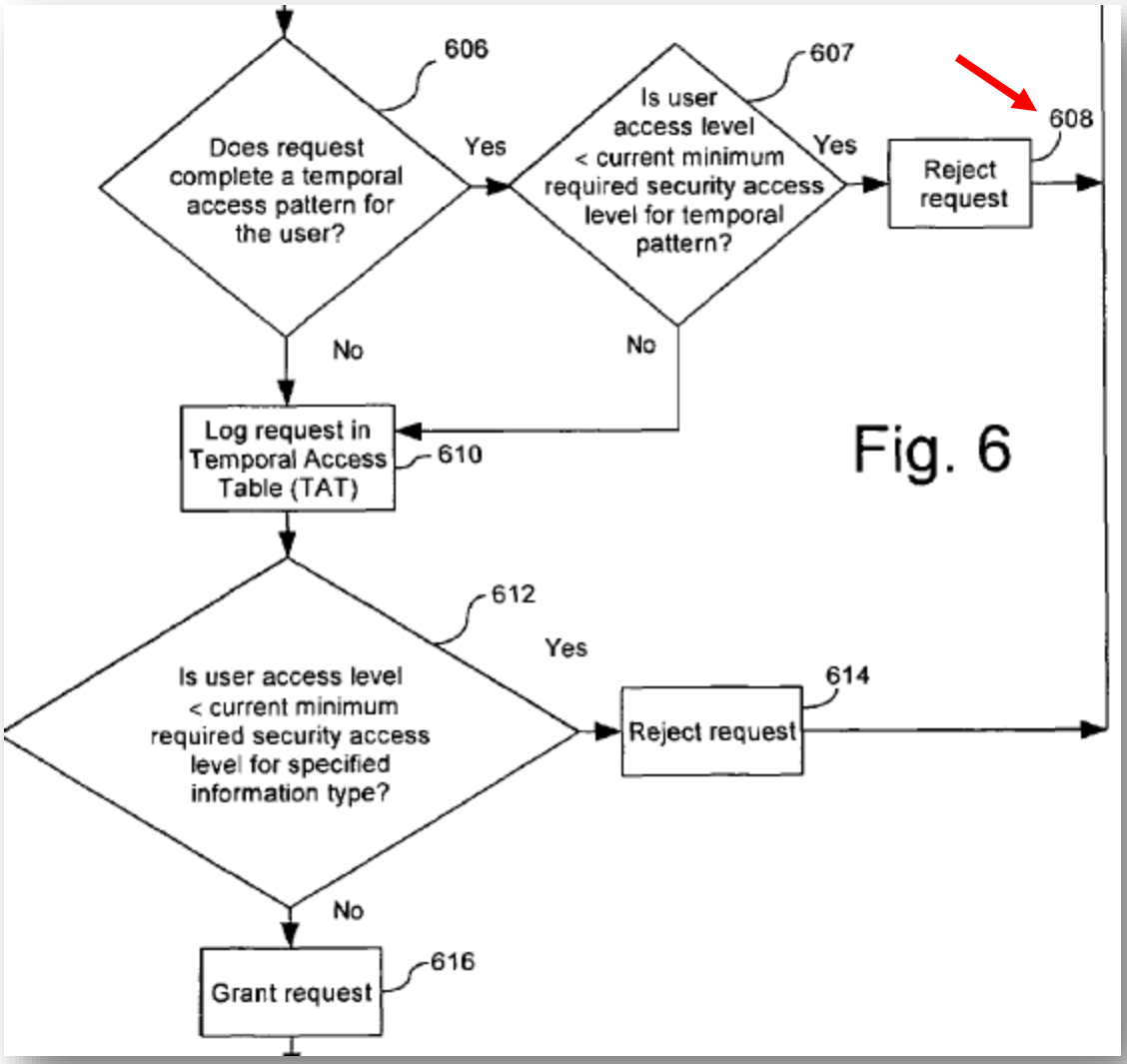
iii. [10f] <u>Function-1</u>: denies said request if said access request completes a prohibited temporal access pattern for said entity

48. **<u>Algorithm</u>**:

"The computer system determine if the access request completes a prohibited temporal access pattern for the entity. If so, the request is rejected." EX-1001 1:62-64.

"Note that the temporal ordering of primitive access operations mandates that the operations occur in the order specified for the security policy to be enforced. Thus, in FIG. 1, if 104_2 is accessed first followed by item 104_1, the security policy associated with node d, item 114_2, will not be activated because the temporal order was not satisfied." EX-1001 5:42-47, *see also* FIG. 1.

"If a [prohibited] temporal pattern is completed," then the request is rejected in step 608":



EX-1001 5:39-49, FIG. 6 (annotated).

To extent that the *a prohibited temporal access pattern* requires a comparison of *access levels*:

"If the access level of the user is not sufficient for the requested action, then the request is rejected in step 608." EX-1001 5:47-49.

iv. [10f] <u>Function-2</u>: grants said access request only if it does not complete a prohibited temporal access pattern for said entity, and said

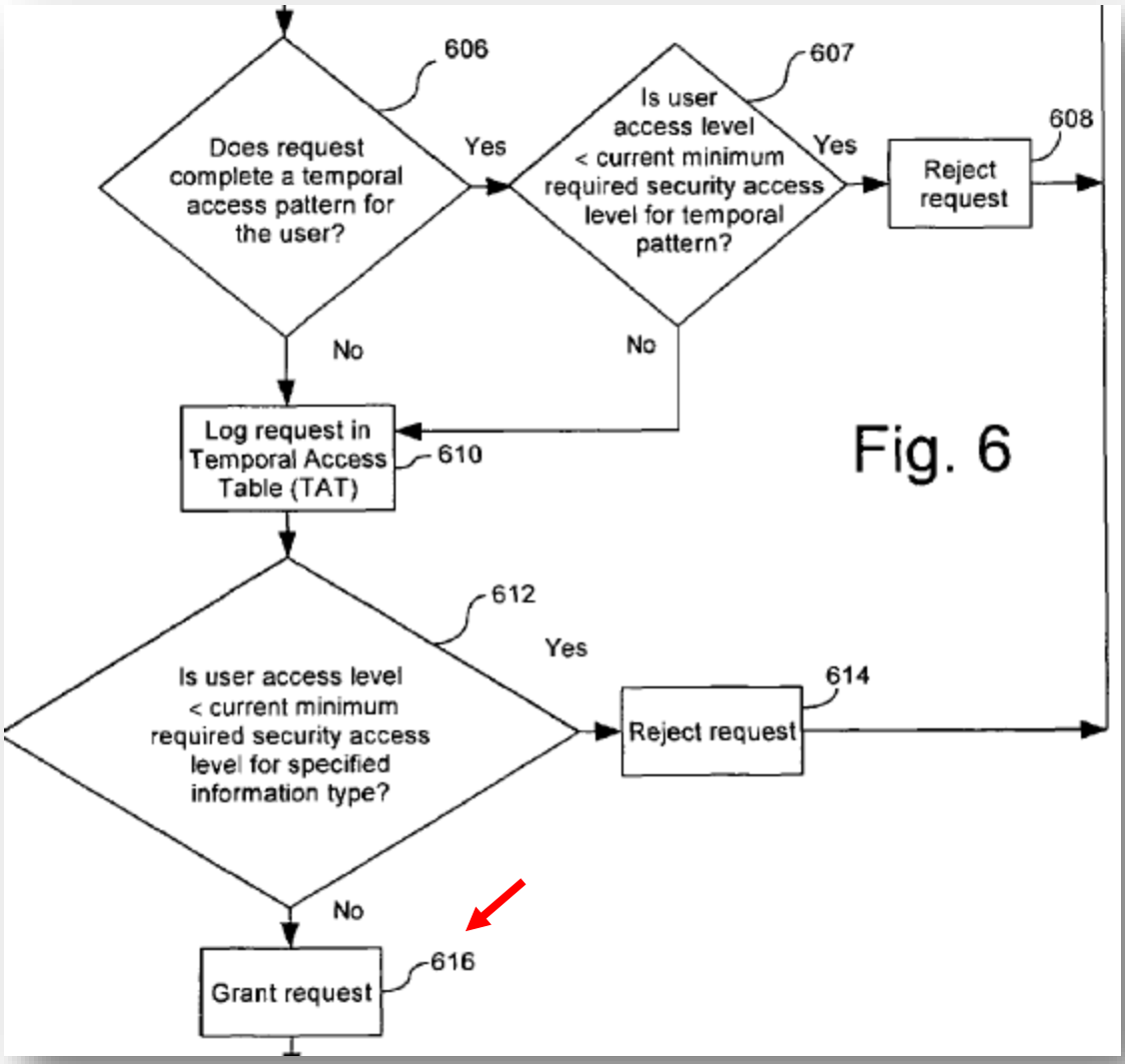
minimum access level for said first base node does not exceed said predetermined access level

49. **<u>Algorithm</u>**:

"If the request does not complete a [prohibited] temporal access pattern" "then the system continues on to step 610 and logs the request in the Temporal Access Table (TAT). As shown in FIG. 4B, the Temporal Access Table maintains a history of the primitive operations **performed** by a user." EX-1001 5:50-57.

"[T]he system can grant the access request only if the minimum access level for the first base node does not exceed to the predetermined access authorization level." EX-1001 1:67-2:3

"[I]f the user does have a sufficiently high security access level then the request is granted in step 616":



EX-1001 6:8-12, FIG. 6 (annotated).

To extent that the *a prohibited temporal access pattern* requires a comparison of *access levels*:

"If … the user's access level is sufficient for the temporal pattern completed, then the system continues on to step 610 and logs the request in the Temporal Access Table (TAT). As shown in FIG. 4B, the Temporal Access Table

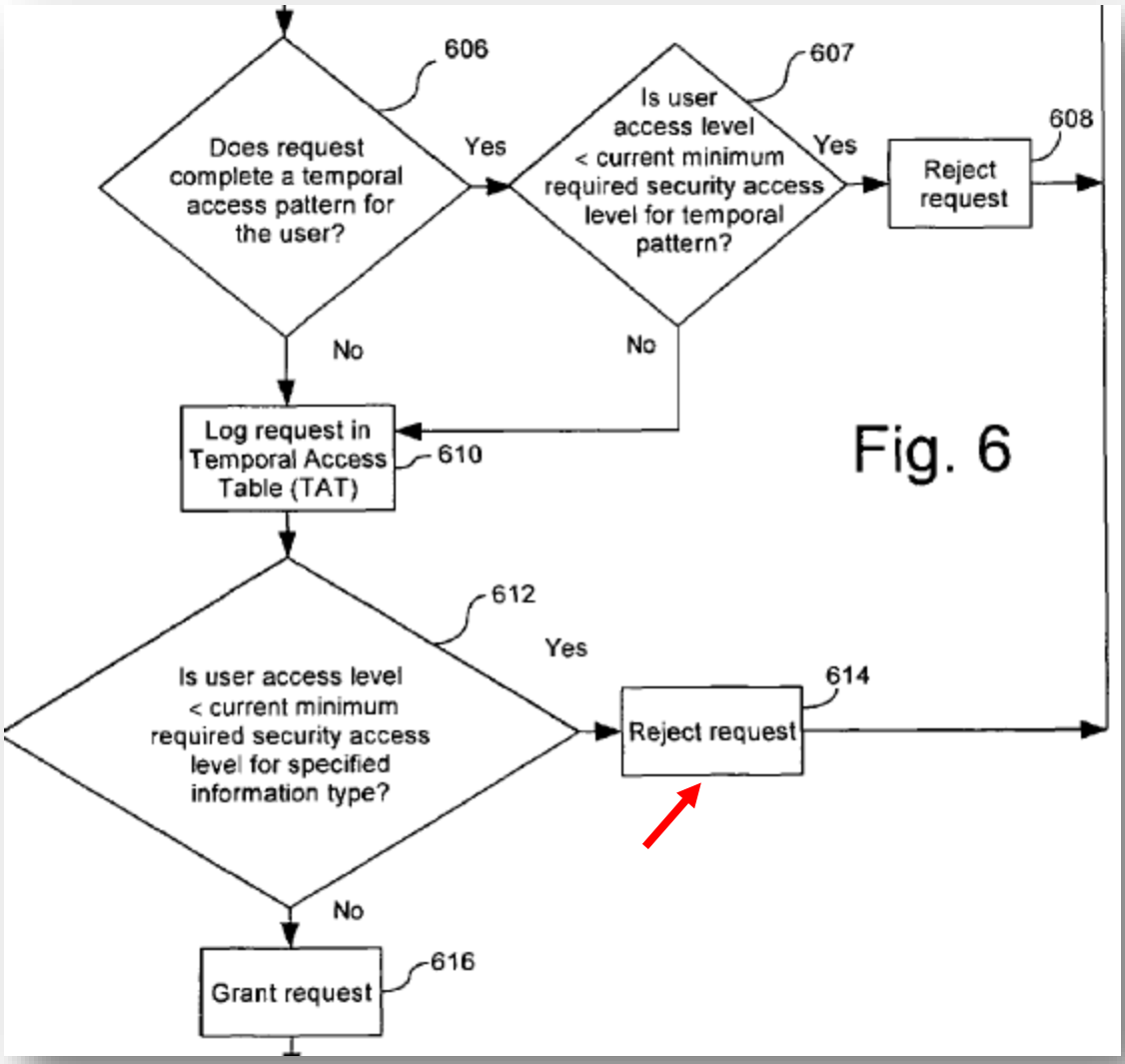
maintains a history of the primitive operations performed by a user." EX-1001 5:50-57.

v. [11] <u>Function</u>: denies said request if said minimum access level for said first base node exceeds said predetermined access level for said entity

50. **<u>Algorithm</u>**:

"The method can also include the step of denying the request if the minimum access level for the first base node exceeds the predetermined access authorization level assigned to the entity." EX-1001 2:3-6.

"If the user does not have authorization for at least the current minimum required security access level, then the request is rejected":



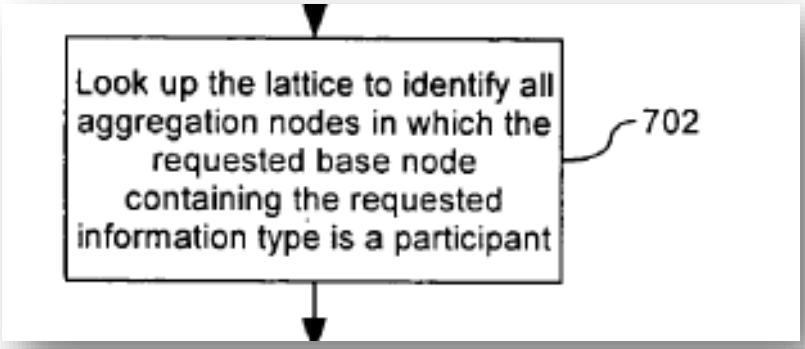
EX-1001 6:8-12, FIG. 6 (annotated).

vi. [13] <u>Function</u>: identifies within said hierarchy any higher-level nodes that are aggregations comprising said first base node

51. **<u>Algorithm</u>**:

"[T]he method can include the step of identifying within the hierarchy any higher-level nodes that are aggregations comprising the first base node." EX-1001 2:13-15.

"[T]he process can continue in step 702 of FIG. 7 by identifying all higher-level aggregation nodes 112 in which the base node containing the requested information type is a participant":



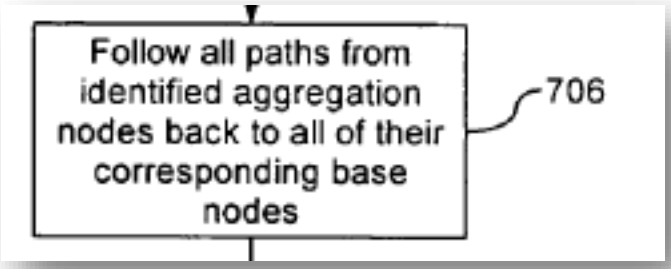
EX-1001 6:42-45, FIG. 7.

vii. [14] <u>Function</u>: identifies within said hierarchy any nodes that comprise children of any generation of said higher-level nodes that are aggregations comprising said first base node

52. **<u>Algorithm</u>**:

"The method can also include identifying within the hierarchy any nodes that directly or indirectly comprise children of any of the higher-level nodes that are aggregations comprising the first base node." EX-1001 2:16-19.

"[I]n step 706 the computer system can identify all base nodes that are also participants in that particular aggregation node. This can be accomplished, for example, by following all paths from the identified aggregation nodes back to all of their corresponding base nodes":



EX-1001 6:48-53, FIG. 7.

53. The prior art relied on in this Declaration performs these functions using the structures described in the '708 Patent or equivalents thereof. Because the prior art discloses each claim term of the Challenged Claims under any reasonable construction, including if means-plus-function applies, the Board need not adopt any claim construction in order to determine that the Challenged Claims are obvious.

D. Priority Date of the Challenged Claims

54. I have been informed that the '708 Patent claims the benefit of U.S. Patent Application Ser. No. 10/798,074, filed on March 11, 2004. Solely for the purposes of this IPR, I have been instructed to assume the priority date is March 11, 2004.

E. Person of Ordinary Skill in the Art

55. I have been informed by counsel that a POSA in the relevant art is a hypothetical person who is presumed to be aware of all pertinent prior art, thinks along conventional wisdom in the art, and is a person of ordinary creativity.

56. Unless otherwise stated, my testimony below refers to the knowledge of a POSA as of March 11, 2004, the date that I was advised by counsel to use as the priority date of the '708 Patent. I am familiar with the knowledge and capabilities of one of ordinary skill in the art as of that priority date.

57. In my opinion, as to the '708 Patent, a POSA in March of 2004 was familiar with computer-security and, in particular, access-control methods. A POSA would have had at least a Bachelor's degree in Computer Science, Computer Engineering, or a similar degree and at least two years of professional experience with computer systems, including computer-security and access-control methods. More experience can supplement less education and vice versa. The level of skill in the art is further demonstrated by the references cited herein, including the Grounds' references and the background references discussed in the State of the Art section below (§V.F). While this Declaration may occasionally use the present tense, my analysis is from the perspective of a POSA as of the time of the purported invention.

F. State of the Art

58. In the following section, I describe the state of the art for enforcing computer security as of March 2004. The prior art references, and the discussions of what was known to a POSA, provide the factual support for the general description of the state of the art at the time of the invention, provide additional

motivation to modify or combine the references. Accordingly, the Board should consider these state of the art references.

1. Well-known sensitivity-based (i.e., "access levels") access control methods

59. By March 2004, controlling computer access based on the sensitivity of the information and the authorization or "clearance" level of the requesting user was well known.

60. For example, in 1985, the Department of Defense published the *Trusted Computer Evaluation Criteria* to "provide guidance to manufacturers" for their "widely-available" "commercial products" "for sensitive applications," including "specific security feature requirements" that "encompass the capabilities typically found in" "general-purpose operating systems." EX-1013 6, 8. These features included a "security policy," or "a set of rules that" "determine whether a given subject can" "gain access to a specific object." EX-1013 9. As an example rule: "No person lacking proper personnel security clearance shall obtain access to classified information." EX-1013 9. Such access control rules required using "[a]ccess control labels" "associated with objects" and knowing "what classes of information"—*e.g.*, "security level" and "hierarchical classification"—"[users] are authorized to deal with." EX-1013 9-10, 65, 111. With such information, "a comparison of" a user's "clearance/authorization to" "the classification/sensitivity designation of the information" is performed to determine whether to grant access.

EX-1013 60. Knowing a user's assigned "security level" required identification and authentication of that user. EX-1013 10, 62.

61. These access control methods utilized a "reference monitor [that] enforces the authorized access relationships between subjects and objects." EX-1013 65. Reference monitors were known to function as "an essential element of any system that would provide multilevel secure computing" "controls." EX-1013 65. Such reference monitors were implemented by "security kernel(s)." EX-1013 65.

62. There were also other well-known systems and methods for controlling access based on such "access levels," including using the "classification/sensitivity designation" as taught by *Trusted Computer Evaluation Criteria.* For example, U.S. Pat. Appl. Pub. 2002/0147924A1 to Flyntz ("Flyntz") (Ex-1014), published 2002, discloses a multilevel computer security system:

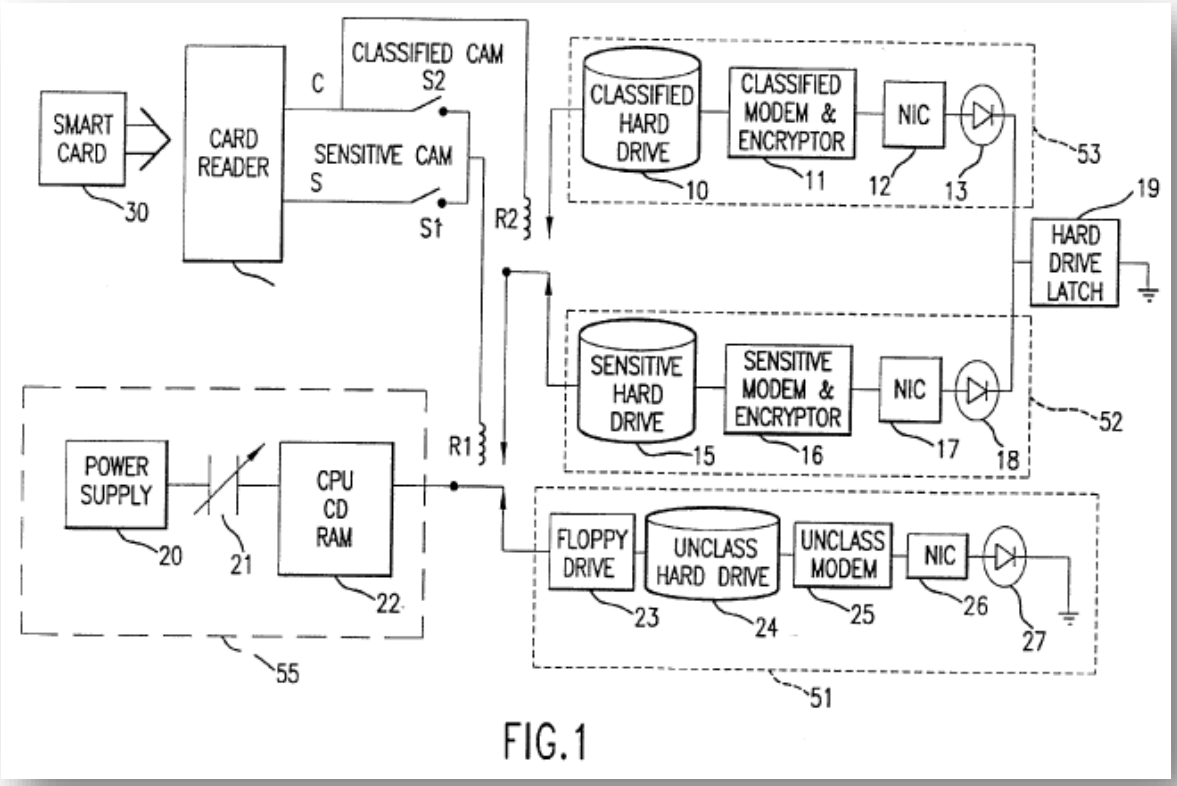


FIG.1

EX-1014 Abstract, FIG. 1. Following authentication, a user could "select one of the security levels" (*e.g.*, "sensitive," "classified," etc.) of the system and, if the user is authorized for that security level, access is permitted through an electronically-activated switch that couples the computer 55 to the information subsystem (*e.g.*, 51, 52, 53). *See, e.g.*, EX-1014 [0032-37, 41], FIG. 1.

63. Other prior art systems, as recognized by the '708 Patent, similarly provided access to information based on "security ranks". EX-1001 1:33-38. U.S. Patent No. 6,453,418 to Ooki et al. ("Ooki") (EX-1007), published 2002, permitted access to user data based on the relative security ranks among users requesting and

creating the data. EX-1007 Abstract, 2:5-9 ("an information accessing method that controls permission to access the user data according to corresponding **relation of a security rank between a user whose data is accessed and a user who would access the user data**"), 2:41-45, FIG. 3 (showing user security ranks).

64. Ooki's system was configured to set one of plural security ranks to each user, set a table for determining if the access to the user data is permitted from a security rank of a user for requesting the access to the user data and a security rank of a user whose data is accessed by the former user, and determine if the access to the user data is permitted by referencing this table. EX-1007 2:21-28. Each entry on this table was composed of a user ID 431 and a security rank 432:

FIG.3

430 USER ID TABLE

431 USER ID	432 SECURITY RANK	433 ATTRIBUTE OF ANOTHER USER
C001	A	BELONGING,POSITION AND THE LIKE OF USER
C002	A	BELONGING,POSITION AND THE LIKE OF USER
C003	B	BELONGING,POSITION AND THE LIKE OF USER
C004	C	BELONGING,POSITION AND THE LIKE OF USER
C005	C	BELONGING,POSITION AND THE LIKE OF USER
B001	C	GUEST
B002	D	GUEST

EX-1007 4:23-26, FIG. 3. Ooki disclosed that "the top rank is A and the rank goes down" "from A to Z" and "if (the security rank of the user who wants to access the data)" is less than or equal to "(the security rank of the user whose data is to be accessed), the user who wants to access the data is" "permitted to access the data, while if not, he or she is not permitted to access." EX-1007 4:38-51. Ooki's comparison of the relative security ranks ("access levels") to make access control decisions is but "an example of a **common** security management" technique well known before the '708 Patent. *Id*.

65. It was also well known in the prior art that "access levels" of objects would vary based not only on the type of information, but also due to the aggregation of information. *Cryptography and Data Security* (EX-1016), published in 1982, recognized that "[a]n **aggregate** of data might require a **higher classification** than the individual items. This is because the **aggregate** might give an **overall picture** that cannot be deduced (or easily deduced) from an individual item." EX-1016 017. This corresponds to the '708 Patent's statements regarding aggregated data and the heightened sensitivity of aggregated information, which was well known in the prior art:

> aggregation nodes will **often** have a higher minimum required security level for permitting access as compared to security levels required for access to participant information types and/or system functions that comprise base nodes 110. This is true because **aggregated data is often**

> **of a more sensitive** nature since it provides **greater context** and can identify relationships between the various individual information types. Consequently aggregated information **will inevitably** be of greater interest to unauthorized users and system administrators **will naturally** wish to impose higher level restrictions on its access.

EX-1001 6:16-27.

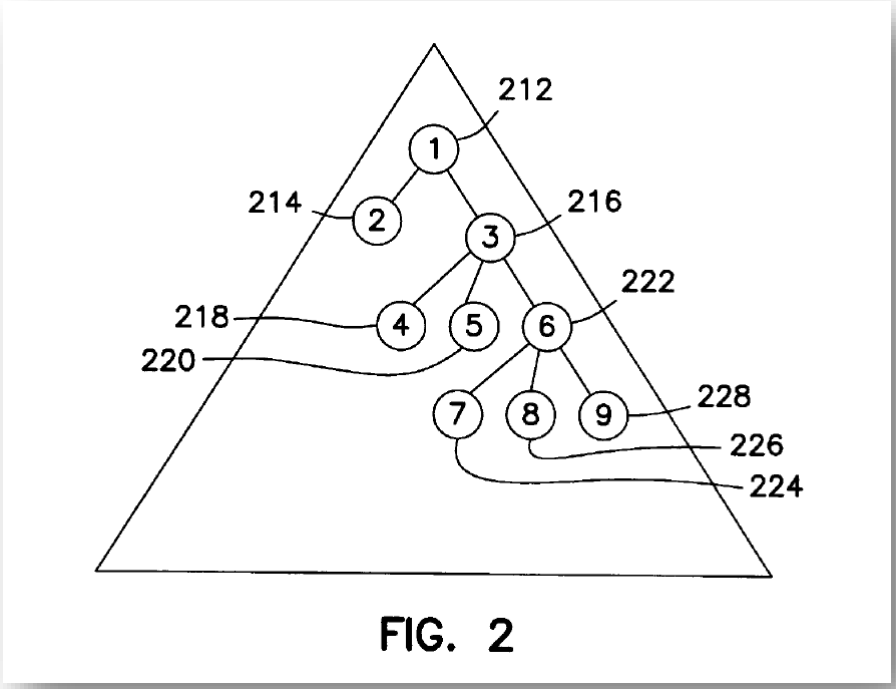
66. Thus, "*comparing a minimum access level established for said first base node to said predetermined access level*" as part of an access control decision was well known prior to the '708 Patent and is nothing more than a "simple relational comparison." EX-1001 1:24-27.

2. Well-known method of organizing data: aggregation of nodes in a tree hierarchy

67. By March 2004, logically organizing computer data in the form of a tree hierarchy having a plurality of leaf nodes and higher-level nodes and the aggregation of nodes was well known, including in methods and systems for ensuring secure access to computer systems.

68. U.S. Patent No. 6,446,077 to Straube et al. (EX-1008), issued 2002, involved aggregation of nodes in the form of a tree structured database 210 representing relationships between multiple objects on a storage domain. EX-1008 5:11-14, FIG. 2. A top level object 212 was represented by a "1" in a circle. A second object 214 was a child of the first object, as was a third object 216. The

third object 216 had three children, 218, 220 and 222 and object 222 had three child objects 224, 226, and 228. EX-1008 5:21-26, FIG. 2.



3. Updating information among nodes

69. By March 2004, propagating user-specific and node-specific information among nodes was well known. Straube (EX-1008) described one such system that involved inherited data in a tree structured database and the propagation of updated inherited information for objects across its local tree of objects. EX-1008 Abstract, 1:5-7. The propagator traversed the directory tree and modified the security descriptors of all related objects. EX-1008 2:13-15. The inherited data included information describing who can access the object, and what level of access permission they are given. EX-1008, 5:36-38. "Security

descriptors" were represented in a directory database and comprised permissions for access and modifying objects. "[T]he propagator 322 reads the queue, and then finds all the children affected by the change and modifies their inherited information by reading each affected object and its parent to determine how to change the inherited information." EX-1008 6:17-21.

4. Well-known use of patterns of behavior/temporal access patterns in computer security and access control

70. By March 2004, the use of patterns of behavior and temporal access patterns for controlling access was well known. U.S. Patent No. 6,334,121 to Primeux et al. ("Primeaux") (EX-1010) involved authenticating users using a set of security rules to track user usage patterns and flag suspicious activity on the system. EX-1010 Abstract, 1:15-20, 2:42-43 ("limiting the ability of unauthorized users to disrupt system operations by monitoring and reporting abnormal user usage patterns"). The system monitored a user's patterns of commands/requests for system access and prevented users from executing destructive commands by detecting unusual patterns in their usage of the system. EX-1010 2:58-60. The system learned a user's usage patterns and integrated the usage patterns with the security rules to recognize "normal" usage patterns for a particular user. When the user's commands did not follow the normal usage pattern, the commands used were checked against the system's security rules. EX-1010 2:63-3:1. When the system detected a usage pattern outside of the user's normal usage pattern, it

triggered the system "to react automatically." EX-1010 10:60-67, Abstract ("perform[ing] automatic responses when confronted with non-authenticated users"). Thus, contrary to statements in the '708 Patent, Primeaux did not merely flag patterns of behavior (EX-1001 1:46-51), but enforced access control policies based on the detected patterns of behavior such as by "automatically log[ging] off the intruder or effect[ing] some other more severe reaction" and "automatically performing a predetermined action" such as "excluding [a] user from further system access." EX-1010 10:41-45, 12:1-16.

71. Lane et al.'s, *Sequence Matching and Learning in Anomaly Detection for Computer Security*, ("Lane") (EX-1011), published 1997, described another computer security system known in the art that involved detecting unauthorized users based on a sequence of user actions and determining whether the sequence of actions was normal or anomalous. EX-1011 43, 44 (identifying "characteristic patterns of actions" and "us[ing] the sequence (an ordered, fixed-length set of temporally adjacent actions) as the fundamental unit of comparison"). The system learned a user's profile and based on sequences of actions (e.g., UNIX commands) by the user, the system determined whether the user's current behavior was consistent or anomalous with past behavior. EX-1011 43.

72. One other prior art system employed a method of controlling access to a computer system by determining whether an "attribute" of a use (i.e., a request to

use an application or access a file) of that system matched one of either an allowed or disallowed use. These "[P]redefined attributes" included an "Action Flow" comprising a "sequence of actions." Ben-Itzhak (EX-1012) [0001, 50-53, 60, 66, 126].

VI. IDENTIFICATION OF CHALLENGES

A. Challenged Claims

73. Claims 1-6 and 9-14 of the '708 Patent are challenged in this Declaration.

B. Statutory Grounds for Challenges

74. The Challenges are set forth in detail below and summarized as follows:

Ground	Claim(s)	Basis	Reference
1	1-5 and 9-14	§ 103	Gladstone (EX-1004) in view of Salazar (EX-1009)
2	6	§ 103	Gladstone in view of Salazar and Hada (EX-1005)

75. U.S. Patent 7,290,266 to Gladstone et al. ("Gladstone") (EX-1004), is titled "Access Control by a Real-Time Stateful Reference Monitor with a State Collection Training Mode and a Lockdown Mode for Detecting Predetermined Patterns of Events Indicative of Requests for Operating System Resources Resulting in a Decision to Allow or Deny Activity Identified in a Sequence of Events Based on a Rule Set Defining a Processing Policy." EX-1004. I have been

informed that Gladstone is prior art. Gladstone was not cited or applied by the Examiner during the '708 Patent prosecution. EX-1001 (56); EX-1002.

76. U.S. Patent Application Publication No. 2004/0122792 to Salazar ("Salazar") (EX-1009) is titled "Method, System, and Program Product for Managing Access to Data Items in a Database." EX-1009. I have been informed that Salazar is prior art. Salazar was not cited or applied by the Examiner during the '708 Patent prosecution. EX-1001 (56); EX-1002.

77. U.S. Patent Application Publication No. 2005/0076030 to Hada et al. ("Hada") (EX-1005) is titled "Method and System for Providing Path-Level Access Control for Structured Documents Stored in a Database." EX-1005. I have been informed that Hada is prior art. Hada was not cited or applied by the Examiner during the '708 Patent prosecution. EX-1001 (56); EX-1002.

VII. IDENTIFICATION OF HOW THE CHALLENGED CLAIMS ARE UNPATENTABLE

A. Ground 1: Claims 1-5 and 9-14 Are Obvious Over Gladstone (EX-1004) and Salazar (EX-1009)

1. Gladstone

a. Analogous Art

78. It is my opinion that Gladstone is analogous art to the claimed invention. For example, the field of endeavor of the '708 Patent's claimed invention is "generally [directed] to computer and information security and more particularly to security measures that enforce security based on logical"—*i.e.*,

software—"implementation methods." EX-1001 1:8-11, 19-21. The '708 Patent's stated solution uses a "temporal access pattern" in access control decisions to ensure "secure access to a computer system." EX-1001 1:55-2:6, Claims 1, 9, 10. The '708 Patent contrasts this solution with the well-known "simple relational comparison" and "point tests" of conventional "logical security safeguards [that] typically provide access based on a user/group/role identifier and an access control list for the file, database, or system function," faulting (incorrectly, §V.F.4) the art for not "enforc[ing] security based on patterns of behavior," *i.e.*, "temporal patterns of access." EX-1001 1:22-51.

79. Gladstone is in the same field of endeavor as the '708 Patent because it is directed to systems and methods for controlling security policy of a computer system. EX-1004 3:46-48, Title. These systems and methods used, *inter alia*, software "logic" within a stateful reference monitor. EX-1004 5:57-60. Like the '708 Patent, Gladstone acknowledges the well-known "common scenario" of "control[ling] access to various system resources" "for purposes of securing a system against deliberate and malicious attack" and the limitations of such access control techniques. EX-1004 1:39-42, 3:11-16. These "common" solutions control access to "system resources" that include "files" and various "network connections," "system call functions," etc. (*i.e.*, computer system functions). EX-1004 1:22-25; 10:9-3.

80. Gladstone is reasonably pertinent to the problem addressed by the '708 Patent's claimed invention because its solution, like the '708 Patent's, provides access control to such resources "based on a pattern of activity," "prohibited pattern[s]," "behavior," "anomalous behavior," "combination[s] of events," and, specifically, "a particular **sequence of events**." EX-1004 Title, Abstract, 2:3-4; 4:46-65, 6:13-24, 9:11-27, 9:54-55, 11:1-3, 11:63-65, 13:8-17, Claims 1, 8-10.

b. Overview of Gladstone

81. Gladstone is directed to a system and method for controlling a security policy of a computer system. EX-1004 3:46-48. Gladstone discloses that an "important aspect of system security is monitoring system events 100, e.g. access requests to a resource, and controlling the effect of those events, e.g. blocking or allowing the requested access to a resource." EX-1004 5:6-9. To effect this access control, events 100 are "communicated to a stateful reference monitor 104" that makes decisions to allow or block the event. EX-1004 5:9-24. The stateful reference monitor operates by "correlat[ing] the access requests for a number of different resource types-files, system configuration information, network access, system calls, local or remote object invocation, etc." EX-1004 9:43-55. Gladstone describes that this "comprehensive view" "enables the stateful

reference monitor 104 to make better access control decisions than a conventional, single resource type stateless reference monitor." EX-1004 9:50-53.

82. Correlating access requests enables Gladstone's stateful reference monitor to "detect[] anomalous behavior and prevent further access." EX-1004 9:54-55. If "a prohibited combination of events" were "X-Y-Z"—each a separate event (access request), i.e. "Event X, Event Y and Event Z"—a rule would prohibit Event Z if Event X and Event Y occurred within a prescribed timeframe of one another. EX-1004 6:13-19. Gladstone's access control decisions are not made merely on the basis of generic combinations of events (access requests). Rather, Gladstone also describes policies controlling "patterns of events" and "pattern(s) of activity" that protect against "prohibited" or "unsafe" access requests. EX-1004 9:11-14, Claims 1, 8-10. Gladstone describes that such pattern-based determinations utilize a "collection of information, that is collected in real-time, indicative of a condition of a machine or process as a result of a particular sequence of events leading to the condition." EX-1004 2:1-3:42. Gladstone explains that using the "particular sequence of events leading to the condition" is a key distinction between its stateful reference monitor and prior art "stateless reference monitors" that "control[led] access to resources" in different ways. EX-1004 2:1-3:42. Gladstone does not, however, disregard the use of stateless monitors, stating that its "stateful reference monitor 104 acts as an additional

processing step over and above any existing stateless reference monitor in the operating system. Thus, [stateful reference monitor 104] cannot grant access when the standard stateless reference monitor would have denied access." EX-1004 10:48-52.

83. Thus, Gladstone discloses the same solution as the '708 Patent—access control decisions utilizing particular sequences of events.

2. Salazar

a. Analogous Art

84. Salazar is analogous art to the claimed invention. The field of endeavor of the '708 Patent's claimed invention is "generally [directed] to computer and information security and more particularly to security measures that enforce security based on logical"—*i.e.*, software—"implementation methods." EX-1001 1:8-11, 19-21. Although the '708 Patent uses a "temporal access pattern" to distinguish the "simple relational comparison[s]" of prior art systems, it nevertheless incorporates such features into its claimed solutions, *e.g.*, by "comparing a minimum access level established for said first base node to said predetermined access level [of said entity]." EX-1001 Abstract, 1:21-51, 1:55-2:6, Claims 1, 9, 10. Further, the '708 Patent claims "logically organizing said computer system in the form of a **tree hierarchy** having a plurality of **leaf nodes** and **higher-level nodes**":

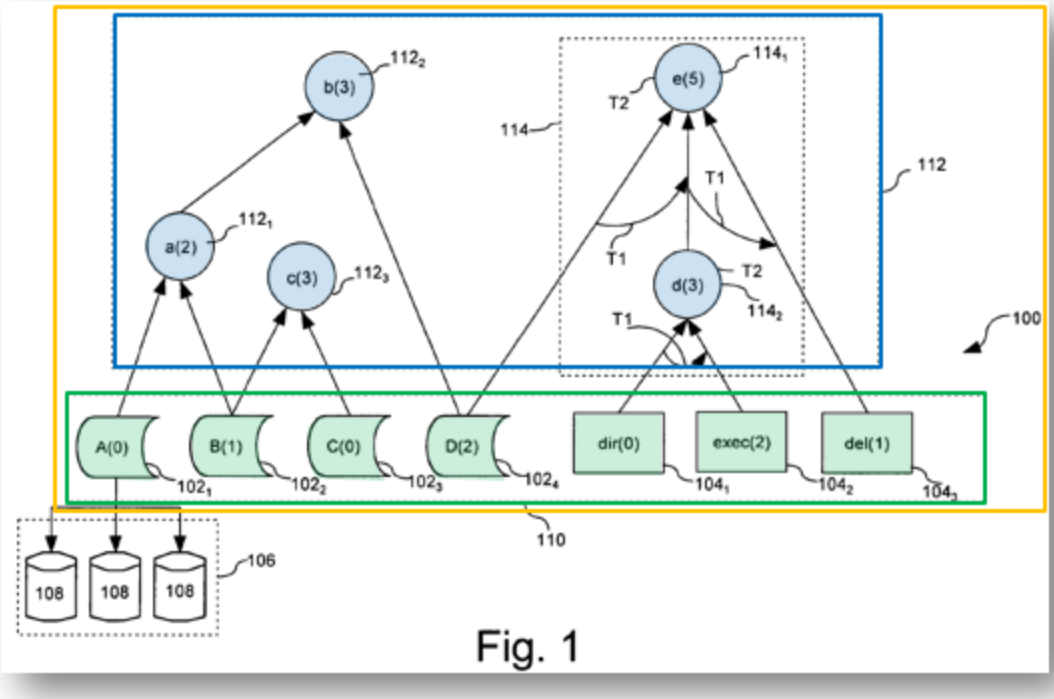
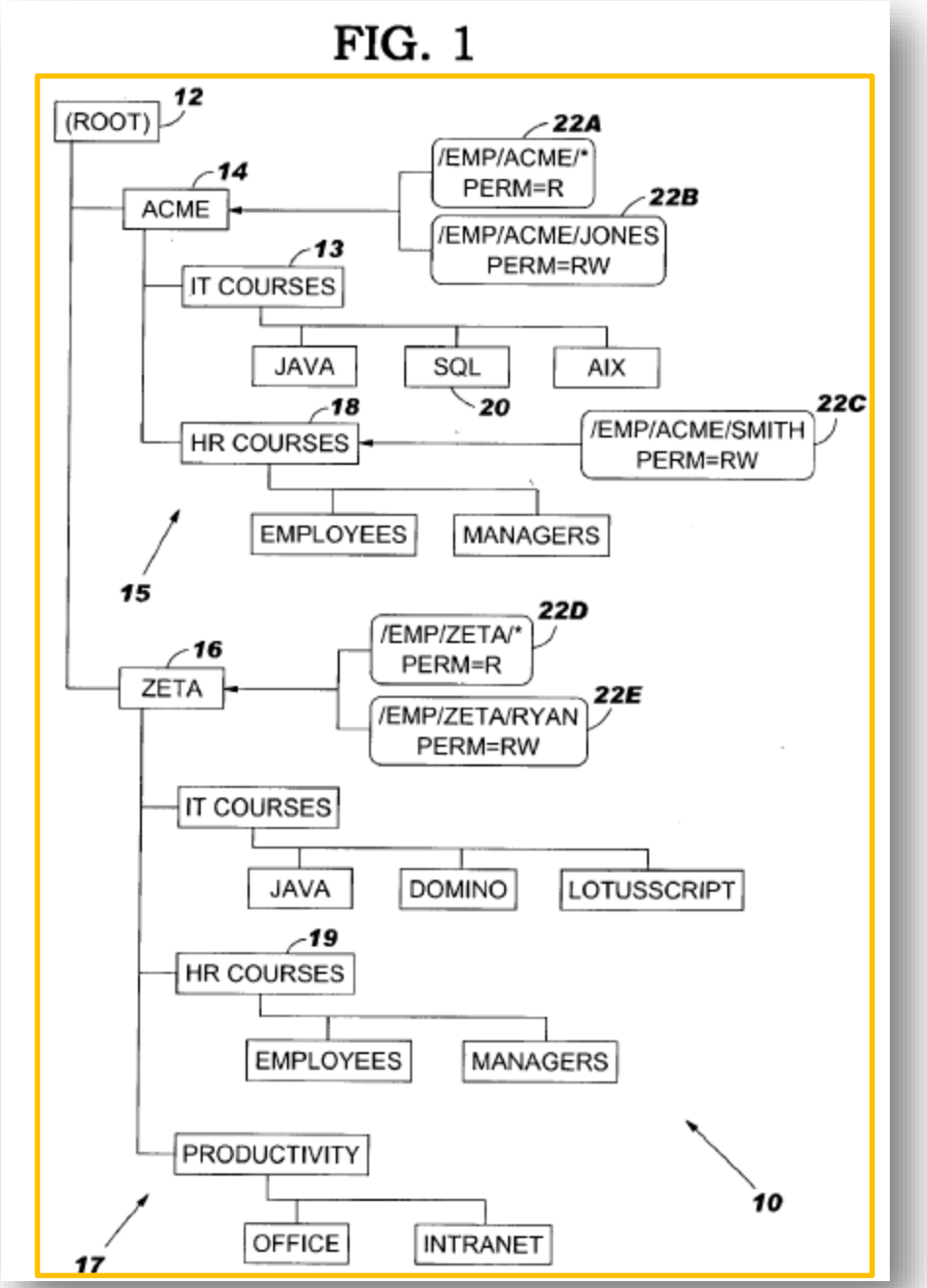


Fig. 1

EX-1001 Claims 3, 9, 10, 3:42-51, FIG. 1 (annotated).

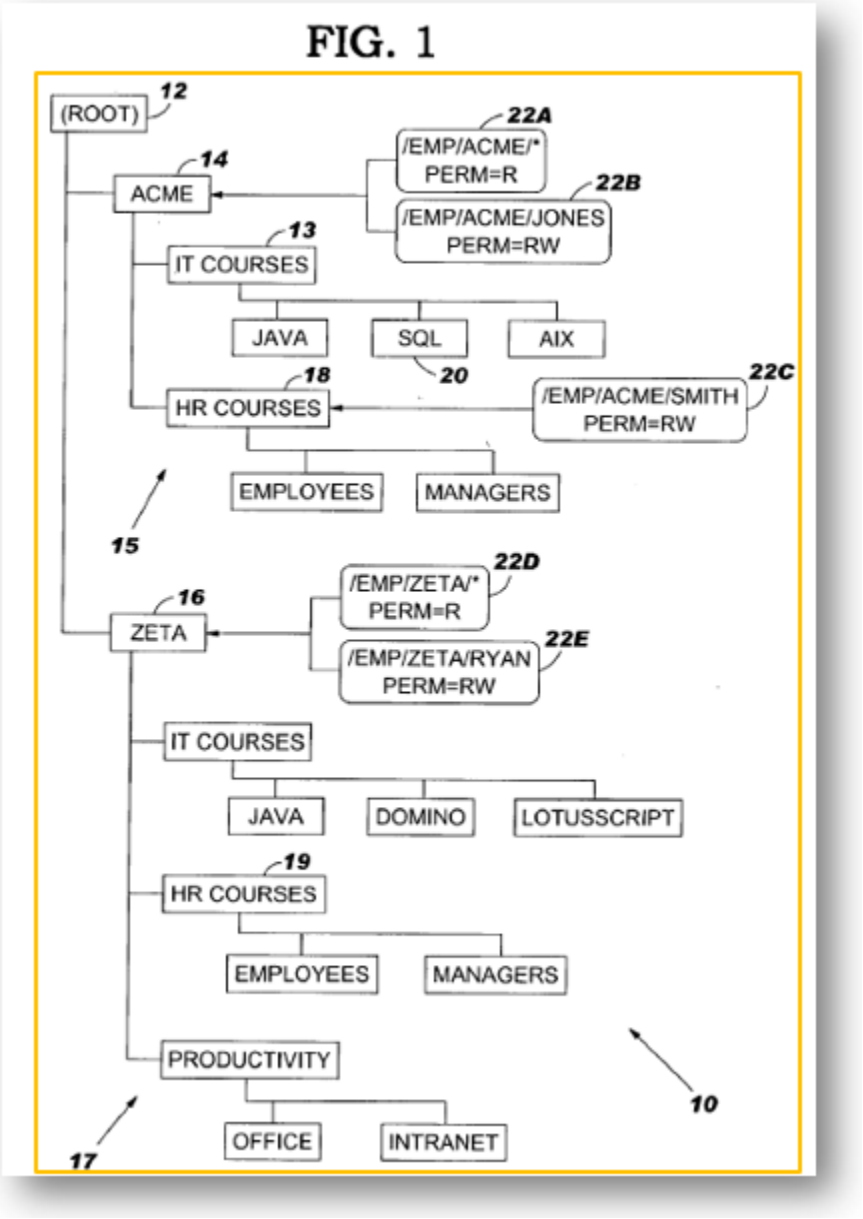
85. Salazar is in the same field of endeavor as the '708 Patent because it is directed to computer and information security by "manag[ing] access to data items." EX-1009 Abstract, [0002]. Salazar is reasonably pertinent to the problem addressed by the '708 Patent's claimed invention because, like the '708 Patent, Salazar performs the "simple relational comparison" by "matching one or more particular user properties associated with a user requesting the access to a user property in an access entry." EX-1009 Abstract, [0007]. Salazar's "data items" are also, like the '708 Patent, "arranged into a **[tree] hierarchy** of nodes":



EX-1009 [0002, 7, 20-23, 42], FIG. 1 (annotated).

b. Overview of Salazar

86. Salazar discloses systems and methods for managing access to data items in a database arranged into **a hierarchy of "nodes"** (*e.g.*, root node 12, ACME node 14, IT COURSES node 13, SQL node 20, etc.):



EX-1009 (54), Abstract, [0002, 7], FIG. 1 (annotated). Hierarchy 10 contains "parent" nodes and "child" nodes, *e.g.*, root node 12 is a parent to both ACME node

14 and Zeta node 14. EX-1009 [0020-21]. Thus, "parent" nodes "can be used to combine"—or aggregate—"child node[s]," including child nodes that form "the top node (or root node) of" another "hierarchy." *Id*. "[C]hild nodes" form "the lowest level of the hierarchy." *Id*.

87. Salazar associates "access entries," *e.g.*, access entries 22A-E, with various nodes of hierarchy 10 as shown in Figure 1. EX-1009 [0024], FIG. 1. Each access entry 22A-E associates a "user property" with what Salazar describes as an "access level." EX-1009 [0024]. The user property may be position (employee "EMP"), division ("ACME"), and user ("JONES") or combination thereof—i.e. "/EMP/ACME/JONES"—and the "levels of access" can be particular operations that are to be performed (*e.g.*, read only, read/write). EX-1009 [0023-25]. Access entry 22B associates the user property "/EMP/ACME/JONES" with a "read/write" access level (i.e., "PERM=RW"):

EX-1009 [0027], FIG. 1.

88. When a user requests data items and/or attempts to perform an operation on data items, access is allowed, or not, "based on one or more particular user properties [that are] associated with user" and, separately, included in "the

access entries." EX-1009 [0030-31]. User properties are provided to an access system that separately obtains the access entries (e.g., 22A-E) that then "determines the access level of user" "based on the one or more access entries that are **matched** with [the] user" using the property associated with the user and the access entry. EX-1009 [0032-34].

89. Salazar discloses that these "user properties" broadly cover any property or characteristic that may be associated with a requesting user. Salazar discloses that the "user property can be **any** property associated with a set of users such as a path/directory, a group designation, a user attribute, etc." EX-1009 [0024]. This includes both a "group specific property or a personal property associated with [the] user." EX-1009 [0029] (internal quotes omitted). User properties further include, for example, a user's department, country, or role. EX-1009 [0034] ("Dept=ACME and Country=USA," "isManager").

90. Therefore, Salazar discloses granting access based on a comparison of user properties to required user properties identified by access entries associated with various nodes within the tree hierarchy.

3. Motivation to combine Gladstone and Salazar

91. A POSA would have been motivated to combine Gladstone's systems and methods with Salazar's tree hierarchy of nodes and user-property-driven access control decisions to achieve more efficient implementation of access control

across a tree hierarchy of nodes and fine-tuning access control capability. Salazar states that "hierarchy of nodes 10 is **useful** in categorizing data items so that they can be **efficiently** located and retrieved." EX-1009 [0023]. Salazar discloses that its method of associating its access entries with the nodes "provides a more **flexible**, and **efficient** implementation of managing access to data items arranged into a hierarchy of nodes." EX-1009 [0007]. A POSA would have recognized that such a tree structure would improve performance and storage efficiency of the access control system and method described by Gladstone that was used for many and a variety of resources. Salazar discloses that its methods "perform access control at an application level,"—*e.g.*, a database—"based on a user identification or other **user property**" and that "[m]anaging access control at this level allows applications to **fine-tune** the access-control scheme to their needs." EX-1009 [0005]. A POSA would have recognized Salazar as describing the details of a stateless reference monitor as described by Gladstone, and would have recognized that this fine-tuning of access control would benefit Gladstone's system of implementing access control based on sequences of events at the user level.

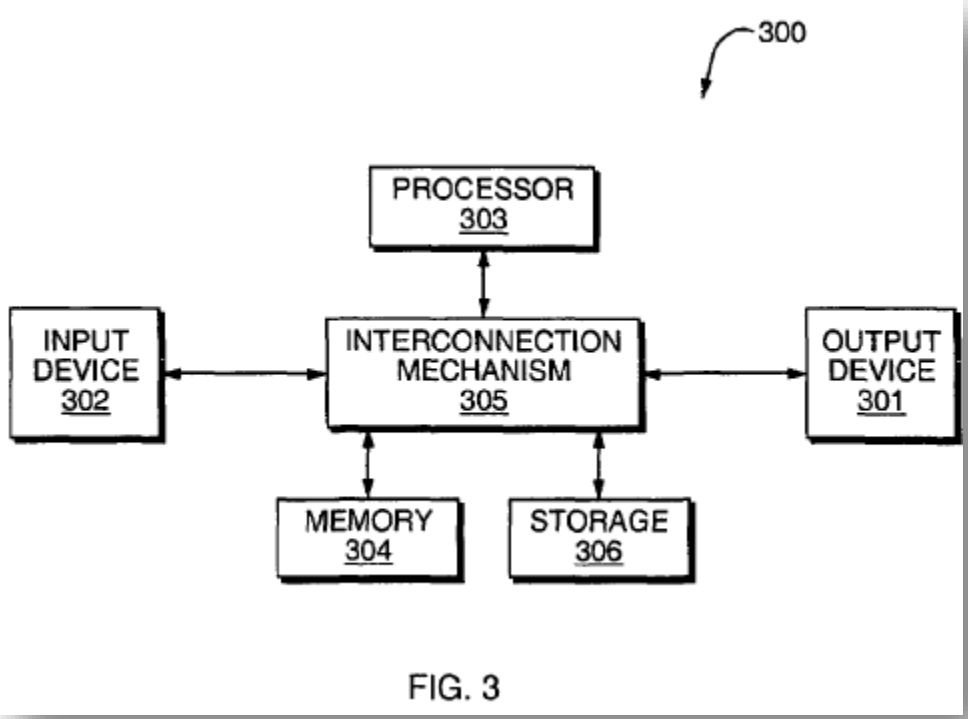
4. Detailed Application to the Challenged Claims

a. Claim 1

i. [1pre] A method for secure access to a computer system, comprising the steps of:

92. To the extent the preamble is limiting, Gladstone discloses implementing a security policy of (*method for secure access to*) a computer system 300 (*a computer system*).

93. Gladstone discloses "controlling an operating policy of a computer system, such as a security policy." EX-1004 3:46-48. Gladstone states that an "important aspect of system security is monitoring system events 100, e.g. access requests to a resource, and controlling the effect of those events, e.g. blocking or allowing the requested access to a resource." EX-1004 5:6-9. Gladstone implements these methods for "computer system 300, shown in FIG. 3":

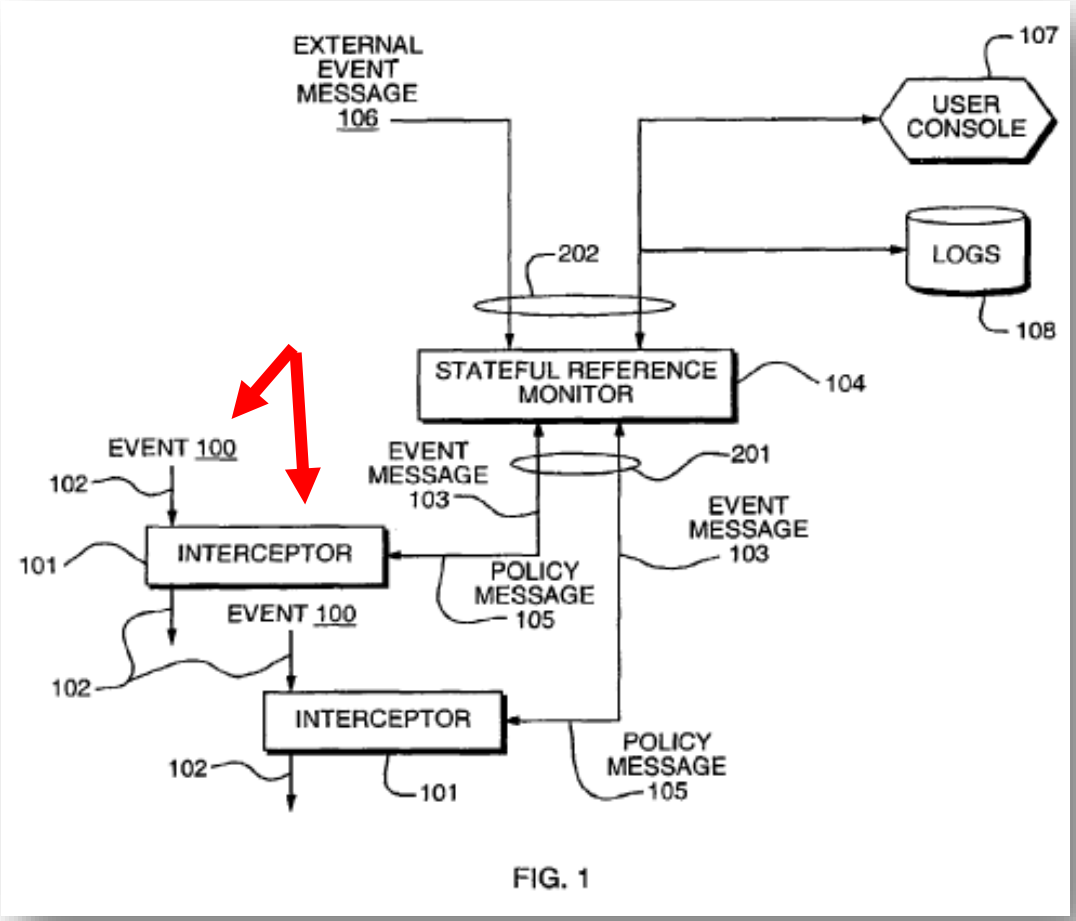


EX-1004 4:57-59 ("FIG. 3 is a functional block diagram of a hardware system on which the software system of FIGS. 1 and 2 can execute"), 14:20-15:12, FIG 3.

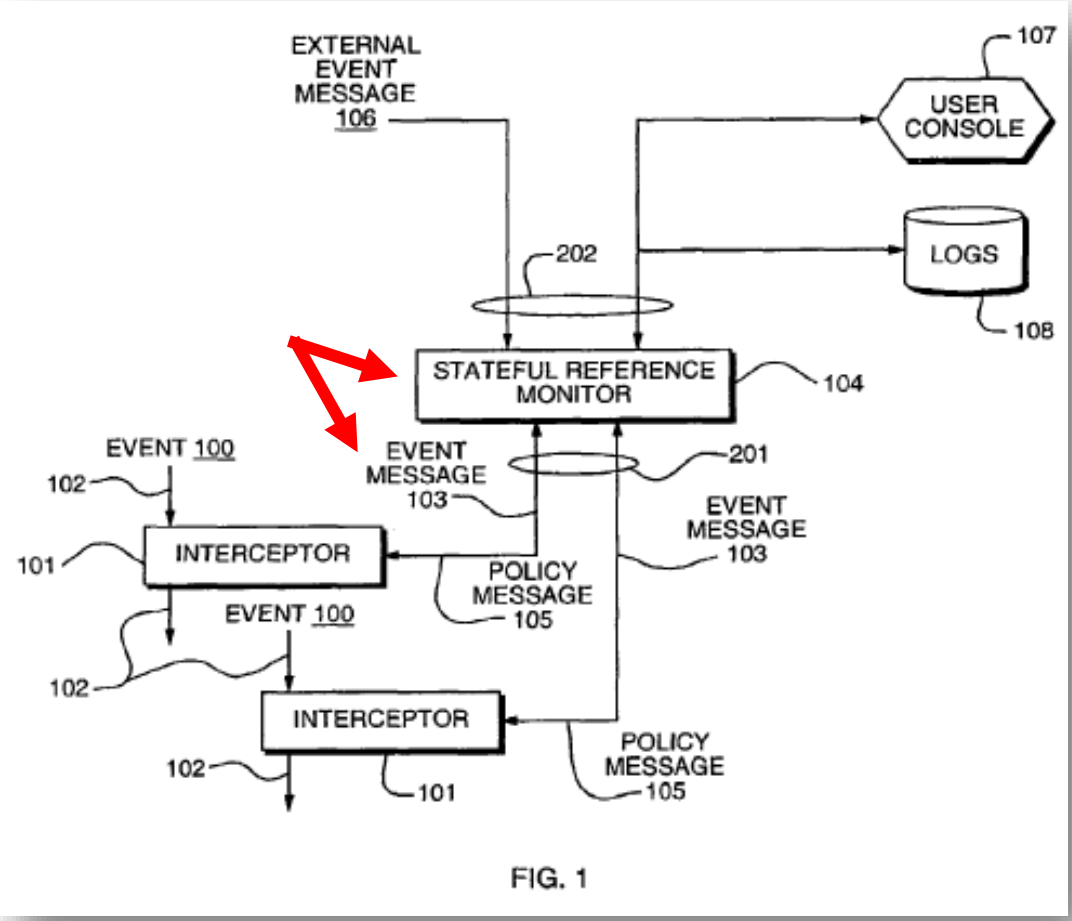
94. Thus, Gladstone discloses *a method for secure access to a computer system*. This method performs elements [1a]-[1e] as discussed below.

ii. [1a] receiving in said computer system a request from an entity with a predetermined access level for access to a first base node representing at least one of an information type and a computer system function;

95. Gladstone, in view of Salazar, renders obvious element [1a]. Gladstone discloses intercepting (*receiving*) events 100 and event messages 103 (*a request…for access*) by inceptor(s) 101 and stateless reference monitor and stateful reference monitor 104 of computer system 300 (*in said computer system*). For example, Gladstone discloses "**[i]nterceptors 101**" that "**intercept**[] various types of operating system **requests**, etc., **referred to herein as events**, for example requests for access to operating system resources such as a file or network connection":



EX-1004 6:37-41, FIG. 1 (annotated); *see also* EX-1004 5:5-6-7 ("system **events 100**, e.g. **access requests** to a resource"). After "an event is intercepted by an interceptor 101," it "generates an **event message 103**" that is "communicated to a **stateful reference monitor 104**" for stateful access control determinations:



EX-1004 5:16-24, 7:25-27, FIG. 1 (annotated). Both the intercepting of the event 100 by interceptor 101 and receiving the event message 103 by stateful reference monitor 104 are *receiving a request… for access*.

96. Gladstone also discloses that its access control methods utilize "[c]onventional" "stateless reference monitors." EX-1004 2:13-20. Gladstone discloses that "stateful reference monitor 104 acts as an **additional** processing step over and above any existing stateless reference monitor" and, "[t]hus, [the stateful reference monitor 104] cannot grant access when the standard stateless reference

monitor would have denied access." EX-1004 10:48-52. Consequently, a POSA would have understood, or at least found obvious, that the request for access associated with interception of event 100 and event message 103 would also have been *received* by Gladstone's integrated stateless reference monitor for its determination to allow or deny an access request prior to stateful reference monitor 104's evaluation of that request. EX-1004 10:48-52. Therefore, both of Gladstone's stateless and stateful reference monitors *receiv[e the] request…for access*.

97. Each of interceptor 101, stateful reference monitor 104, and stateless reference monitor are part of a computer system. As described in [1pre], Gladstone's methods, which includes the intercepting of events 100 and receiving the event messages 103, occur in computer systems 300 and, thus, Gladstone discloses *receiving in said computer system a request …for access*. *See* [1pre]; EX-1004 1:64-67 ("A reference monitor is a component of a computer system.").

98. Gladstone further discloses that events 100 and event messages 103 (*request…for access*) are requests made for resources (*node[s] representing at least one of an information type and a computer system function*). Specifically, "system events 100" are "access requests to a **resource**." EX-1004 5:6-9. Gladstone discloses "a number of different resource types—files, system configuration information,"—*i.e.*, *information type[s]*—and "network access,

system calls, local or remote object invocation, etc."—*i.e.*, *computer system function[s]*. EX-1004 9:43-55, 10:9-47 (describing various computer system functions and acknowledging that other functions and resources may need protecting); 6:16 (referencing generic "Event X, Event Y and Event Z"); 6:64-7:3 (resources include "Outlook personal address books"). Gladstone further discloses that its methods determine "an action for the interceptor 101" and stateless reference monitor "to take, such as allowing" "or not allowing" "an access request" "continue." EX-1004 5:20-24. Therefore, Gladstone's disclosure of *node[s] is* consistent with the description of the '708 Patent. EX-1001 2:67-3:2 ("The security access primitives can form base nodes in the hierarchical, directed graph."), 3:46-47 ("object types and system functions can be defined as" "leaf or base nodes 110"), 3:64-66 (describing "nodes" as "representing information access and operational functions for which the security operations are to be enforced"). Any of the above requested resources is considered *a "first" base node*.

99. Gladstone also discloses that its requests are made by a user and user process (both, individually and combined, *an entity*). For example, Gladstone states that "reference monitor[s]"—such as its stateless reference monitors and stateful reference monitor 104—"determine[] whether an access by" "a user process" "is permitted." EX-1004 1:64-67. These "[u]ser processes are" "identified to users on the basis of the user supplying certain identity tokens." EX-

1004 2:13-20. Gladstone also states that access requests, *e.g.*, "events X, Y and Z" "may be" "associated" "with the same user"—i.e., *entity*. EX-1004 6:21-24. Both the user and user process are *an entity* as described by the '708 Patent. EX-1001 1:59 ("The entity can be a user or a process").

100. Gladstone's user and user process (both, individually and combined, *an entity*) are, as modified by Salazar, each an entity with "user properties" (*with a predetermined access level*). For example, Gladstone describes its stateless reference monitors are "[c]onventional references monitors" "used to determine whether a particular **access** to a file or other resource is permitted." EX-1004 2:1-12. In most cases, the stateless reference monitor's "access decision is" "based on the **identity of the user**" "mak[ing] the request." EX-1004 2:13-20. The access decision also references "permissions flags or an access control list associated with the resource," "individual permissions [associated] with each user," "a set of tokens" for "individual users," and the "**rights** and **privileges** attributed to [a] **user**." EX-1004 2:6-33, 8:2-3. Gladstone discloses that conventional "systems give finer control by associating individual permissions with each user." EX-1004 2:30-33. Gladstone further discloses that the "conventional reference monitors" were "found in the kernels of various Operating Systems." EX-1004 2:7-12.

101. Salazar discloses that when a user requests data items and/or attempts to perform an operation on data items, access is allowed, or not, "based on one or

more particular user properties associated with user" (*a predetermined access level*) and, separately, "the access entries" (*minimum access level*, *see* [1c]) EX-1009 [0030-31]. Specifically, user properties are provided to an access system that separately obtains the access entries (*e.g.*, 22A-E) that then bases the access determination "on the one or more access entries that are **matched** with [the] user" using the property associated with the user (*a predetermined access level*) and the access entry. EX-1009 [0032-34].

102. Salazar discloses that these "user properties" broadly cover **any** property or characteristic that may be associated with a requesting user. For example, Salazar discloses that the "user property can be **any** property associated with a set of users such as a path/directory, a group designation, a user attribute, etc." EX-1009 [0024]. This includes both a "group specific property or a **personal** property associated with [the] **user**." EX-1009 [0029] (internal quotes omitted). User properties could further cover a user's department, country, or role. EX-1009 [0034] ("Dept=ACME and Country=USA," "isManager").

103. A POSA would have understood that Salazar's broad disclosure of the "user property" that "can be **any** property" included the well-known and widely implemented security or classification levels (i.e., *access levels*) used in prior art access control systems, which were known to implement access rules that "compar[ed]" a user's "clearance/authorization," "security level," and/or

"hierarchical classification"—all examples of a *predetermined access level*—"to" the "[a]ccess control labels"—the "sensitivity level (e.g., classification)" or the *minimum access level*—"associated with objects" to determine whether to grant access. EX-1013 9-10, 60, 65, 111. EX-1014 [0032-37, 41], FIG. 1 (providing access control based on "sensitive," "classified," classifications of information and permissions of a user); EX-1007 Abstract, 2:5-9, 2:21-28, 4:23-26, 4:38-51 ("security rank" assigned to users that determined whether access to information would be allowed or denied, where, *e.g.*, "the top rank is A and the rank goes down one by one from A to Z"), FIG. 3:

FIG.3

430 USER ID TABLE

431 USER ID	432 SECURITY RANK	433 ATTRIBUTE OF ANOTHER USER
C001	A	BELONGING,POSITION AND THE LIKE OF USER
C002	A	BELONGING,POSITION AND THE LIKE OF USER
C003	B	BELONGING,POSITION AND THE LIKE OF USER
C004	C	BELONGING,POSITION AND THE LIKE OF USER
C005	C	BELONGING,POSITION AND THE LIKE OF USER
B001	C	GUEST
B002	D	GUEST

EX-1016 017 ("[a]n **aggregate** of data might require a **higher classification** than the individual items"); EX-1001 1:24-27 (describing the "inevitabl[e]" and "natural[]" understanding that aggregating information changes sensitivity levels); §V.F.1.

104. A POSA would have found it obvious to combine Salazar's user properties (*a predetermined access level*) with Gladstone's user and user process (both, individually and combined, *an entity*). Both disclose access control decision systems that provide "fine control" by associating data with each user used in the access control decision. EX-1004 2:30-31; EX-1009 [0005]. Further, Salazar states that its fine control is enabled by managing access control at the "application level" (EX-1009 [0005])—the same level at which Gladstone's stateful reference monitor 104 operates (EX-1004 6:24-33). A POSA would have recognized Salazar as describing the details of an example of a stateless reference monitor as described by Gladstone, and would have recognized that this fine-tuning of access control would benefit Gladstone's system of implementing access control based on sequences of events at the user level, and thus motivated to make such modification. A POSA would have had a reasonable expectation of success because comparative access-control systems where a "common security management" requiring nothing more than "simple relational comparison," as

admitted by the '708 Patent, well within the skill of the POSA. EX-1007 4:37-44; EX-1001 1:25-28.

105. Finally, a POSA would have understood that Salazar's user properties, including multi-level security ranks and/or classifications levels, would have been associated with Gladstone's users prior to the access control decision and, therefore, are *predetermined*. For example, Salazar states that its user properties 40 are "set[]" "stored" and later "retriev[ed]" by various systems. EX-1009 [0029, 33].

iii. [1b] determining if said access request completes a prohibited temporal access pattern for said entity;

106. Gladstone discloses that stateful reference monitor 104 *determin[es]* for events 100 and event messages 103 (*said access request*) whether a sequence of events should be denied (*completes a prohibited temporal access pattern*) for the user or user process (*for said entity*).

107. For example, Gladstone discloses that "various types of patterns of events can be programmed" "for detection and action" by the stateful reference monitor 104. EX-1004 9:11-27. Gladstone's methods "track a **sequence of actions**" and can allow or deny requests. EX-1004 13:8-10. Such "sequences are used for detecting malicious behavior" and "normal behavior." EX-1004 13:13-21. This functionality enables the "stateful reference monitor [to] ensure[] that

requests are ordered, i.e. **a request either happens before or after another request**"—*i.e.*, that a *temporal access pattern* occurs or does not occur. EX-1004 12:22-28. For example, a "prohibited combination of events" of "X-Y-Z" would "prohibit Z" if the request for Event Z occurred while "state database 207 holds information X-Y"—indicating that events X and Y had already occurred. EX-1004 6:13-14. While Gladstone discloses that event Z could be blocked based on the combination of events X-Y-Z, regardless of the order in which the events occurred, Gladstone also discloses "ensur[ing] that requests are **ordered**, i.e. a request either happens before or after another request" and "track[ing] a **sequence** of actions." EX-1004 12:22-28; 13:8-10. Thus, a POSA would have understood that Gladstone discloses blocking (denying the request) combinations of actions due to the particular "sequence of actions" of those events and, therefore, would have found it obvious for the stateful reference monitor 104 to require a particular ordering of events X, Y, and Z for the security policy to be applied to them. Therefore, a POSA would have found it obvious for Gladstone's stateful monitor to deny access to event Z if a user requested access because the user had already requested access to event X and/or Y in a particular order. EX-1004 13:8-10, 12:22-28. Gladstone expressly describes that monitoring the order of actions was the very purpose of stateful monitors—such as stateful reference monitor 104—that observed the "dynamic state" of computer systems using information

"collected in real-time, indicative of a condition of a machine or process as a result of a **particular sequence of events**." EX-1004 2:1-4. These disclosures of Gladstone also correspond to the '708 Patent's description of *temporal access patterns*: "**temporal ordering** of primitive access operations mandates that the operations occur **in the order specified** for the security policy to be enforced." EX-1001 5:43-45. Therefore, Gladstone discloses *determining if said access request completes a prohibited temporal access pattern*, and applying its security policy based on the same.

108. Gladstone also discloses that the pattern of events (*prohibited temporal access pattern*) is specific to a user and user process (*for said entity*). As discussed above for [1a], Gladstone discloses user and user process (both, individually and combined, *said entity*) having, as modified by Salazar, user properties (*predetermined access level*) used in access determinations. *See* [1a]. Further, Gladstone states that the combination of "[e]vents X, Y and Z" "may be" "associated" "with the same user."[4] EX-1004 6:21-23. When the user or user

---

[4] Gladstone also discloses that access control determinations based a sequence of events (*temporal access pattern*) could be made independent of the user (EX-1004 7:62-66), an embodiment not contemplated by the '708 Patent, further illustrating

process (*said entity*) requests a particular sequence of events (e.g., events X-Y-Z) (*temporal access pattern*) that is blocked, a POSA would have understood that blocking to be directed at the user and, therefore, the blocked sequence of events is a *prohibited temporal access pattern for* the user or user process (*said entity*).

109. To the extent that Gladstone does not expressly disclose blocking a sequence of events (*prohibited temporal access pattern*) specific to a particular user or user process (*for said entity*), a POSA would have nevertheless found doing so obvious. As described above in [1a], Gladstone broadly discloses that "reference monitor[s]"— including stateful reference monitor 104—"determine whether an access by" "a user process" "is permitted." EX-1004 1:64-67; 6:21-23. These determinations "[i]n most cases" are "based on the identity of the user." EX-1004 2:13-20. Gladstone describes that conventional security "systems give finer control by associating **individual permissions with each user**." EX-1004 2:30-21. Thus, a POSA would have found it obvious to use both the "identity of the user" and the associated "individual permissions" in determining whether to block a sequence of events (*prohibited temporal access pattern*) for a particular user or user process (*for said entity*). EX-1010 2:63-3:1 (integrating user usage patterns in

---

that the prior art had far surpassed the purported solutions that the '708 Patent describes before the March 2004 priority date. §V.F.4.

security rules); EX-1011 43-44 (determining consistency of "characteristic patterns of actions," of users.). A POSA would have been motivated to include this individualized access control capability in Gladstone's stateful monitor 104 to achieve the same effect—provide access control based on the individual. Further, a POSA would have had a reasonable expectation of success because, well before March 2004, "individual[ized] permissions" were already used by access-control systems and that "operat[ing] on the basis of the user" "and the rights and privileges attributed to that user" was "conventional." EX-1004 2:30-21, 7:66-8:3; §V.F.1). Therefore, Gladstone discloses or renders obvious *determining if said access request completes a prohibited temporal access pattern for said entity.*

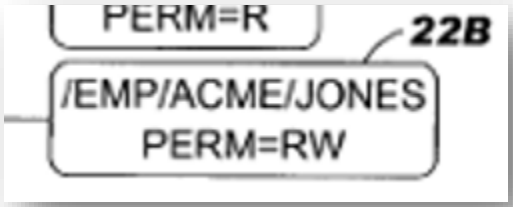
iv. [1c] comparing a minimum access level established for said first base node to said predetermined access level;

110. Gladstone, in view of Salazar, renders obvious making access control determinations by matching (*comparing)* the "user properties" of Gladstone's resources (*a minimum access level established for said first base node*) to Salazar's user properties associated with Gladstone's user and user process (*said predetermined access level*).

111. For example, Gladstone discloses that a "**reference monitor** is a component of a computer system that **determines whether an access** by one

component, for example a user process, of another component, for example, a file **is permitted**." EX-1004 1:64-67, 2:7-12.

112. As discussed above for [1a], Salazar discloses user properties, including "sensitivity level""—*a minimum access level*—to determine whether to grant access. *See* [1a]. As discussed above in [1a], a POSA would have found it obvious to combine Salazar's user properties (*a predetermined access level*) with Gladstone's user and user process (both, individually and combined, *an entity*). *See* [1a]. In addition to the association of the user properties with the user, Salazar also includes the user properties in the access entries. For example, Salazar associates "access entries," *e.g.*, access entries 22A-E, with various nodes of hierarchy 10 in Figure 1. EX-1009 [0024], FIG. 1. Each access entry 22A-E associates a "user property" with what Salazar describes as an "access level." EX-1009 [0024]. For example access entry 22B associates the user property "/EMP/ACME/JONES" with a "read/write" "access level" (i.e., "PERM=RW"):



EX-1009 [0027], FIG. 1.

113. As described for [1a], a POSA would have understood that Salazar's broad disclosure of the "user property" that "can be any property" included the

well-known and widely implemented security or classification levels (*i.e.*, access levels) used in prior art access control systems. *See* [1a].

114. A POSA would have found it obvious to use Salazar's "user properties" in place of Gladstone's one or more static permissions flags or as part of or in place of an access control list associated with the resource (as modified, *a minimum access level established for said first base node*) for the same reasons discussed above in [1a]. *See* [1a]. Both disclose access control decision systems that provide "fine control" by associating data with each user used in the access control decision. EX-1004 2:30-31; EX-1009 [0005]. Further, Salazar states that its fine control is enabled by managing access control at the "application level" (EX-1009 [0005])—the same level at which Gladstone's stateful reference monitor 104 operates (EX-1004 6:24-33). A POSA would have recognized Salazar as describing the details of an example of a stateless reference monitor as described by Gladstone, and would have recognized that this fine-tuning of access control would benefit Gladstone's system of implementing access control based on sequences of events at the user level, and thus motivated to make such modification. A POSA would have had a reasonable expectation of success because comparative access-control systems where a "common security management" requiring nothing more than "simple relational comparison," as

admitted by the '708 Patent, was well within the skill of the POSA. EX-1007 4:37-44; EX-1001 1:25-28.

115. Gladstone, in view of Salazar, discloses the matching (*comparing*) of the above discussed user properties associated with the resources (*a minimum access level established for said first base node*) and the user properties associated with the user (*said predetermined access level*). For example, Gladstone discloses that "In most cases the access decision is made **based on** the identity of the user" "and one or more static permissions flags or an access control list associated with the resource" and matching in the context of access control lists. EX-1004 2:13-20, 2:30-33, 2:36-39. Likewise Salazar discloses that the access control decisions are "based on the one or more access entries that are **matched** with [the] user" using the property associated with the user and the access entry. EX-1009 [0032-34].

116. Therefore, Gladstone, in view of Salazar, renders obvious comparing a minimum access level established for said first base node to said predetermined access level.

v. [1d] granting said access request only if it does not complete a prohibited temporal access pattern for said entity, and said minimum access level for said first base node does not exceed said predetermined access level; and

vi. [1e] denying said request if said access request completes a prohibited temporal access pattern for said entity.

117. Gladstone, in view of Salazar, renders these elements obvious. As discussed in [1b], Gladstone discloses that "stateful reference monitor 104 makes a decision" "to block or allow" a sequence of events (*prohibited temporal access pattern*) for the user or user process (*for said entity*). *See* [1b]. For example, Gladstone discloses that if "a prohibited combination of events is" "X-Y-Z," stateful reference monitor 104 would "prohibit Z" if it had recorded in "state database 207" "[events] X-Y." EX-1004 6:13-24; *see* [1b]. Conversely, a POSA would have understood that Gladstone's monitor 104 would have allowed event Z if it did not complete a *prohibited temporal access pattern*, *e.g.*, if the combination of events X-Y-Z were allowed for the user, if either of events X and Y had not been completed, or if the events occurred in some other order. Therefore, Gladstone discloses or renders obvious *granting said access request only if it does not complete a prohibited temporal access pattern for said entity* ([1d]) and, likewise, *denying said request if said access request completes a prohibited temporal access pattern for said entity* ([1e]).

118. Gladstone, in view of Salazar, renders obvious *granting said access request only if …* Salazar's user properties associated with the Gladstone's resource (*said minimum access level established for said first base node*) does not

exceed the user property assigned to the user and/or user process (*said predetermined access level*). *See also* [1a] (*a predetermined access level*), [1c] (Gladstone and Salazar's user properties assigned to a resource rendering obvious *a minimum access level established for said first base node* and *comparing a minimum access level established for said first base node to said predetermined access level*). As discussed above, a POSA would have understood that Salazar's "user property" included the well-known and widely implemented security or classification levels (*i.e.*, access levels) used in prior art access control systems. *See* [1a, 1c]; EX-1013 9-10 ("what classes of information [users] are authorized to deal with" for "for handling sensitive (e.g., classified) information"); EX-1014 [0032-37, 41], FIG. 1 (providing access control based on "sensitive," "classified," classifications of information and permissions of a user); EX-1007 Abstract, 2:5-9, 2:21-28, 4:23-26, 4:38-51, FIG. 3 (disclosing a "security rank" assigned to users that determined whether access to information would be allowed or denied, where, *e.g.*, "the top rank is A and the rank goes down one by one from A to Z."); §V.F.1. A POSA would have understood that "matching" the user properties for (*see, e.g.*, EX-1004 1:38-2:44; EX-1009 [0026-27, 34-35]) would mean *said minimum access level for said first base node* ***does not exceed*** *said predetermined access level* as "***does not exceed***" is used by the '708 Patent. EX-1001 3:36-41 ("The number represents a minimum security access level required for any process or

user to access that particular function. Thus, for example, the "exec system function labeled exec(2) requires that a process or user possess a security level of at least two in order to access that function."). Further, a POSA would have understood that Gladstone in view of Salazar would allow access when the user property assigned to the user (*predetermined access level*) exceeded the user property assigned to the resource (*minimum access level for said first base node*). EX-1007 4:38-51 (disclosing a system where "the top rank is A and the rank goes down one by one from A to Z" and "if (the security rank of the user who wants to access the data)" is less than or equal to "(the security rank of the user whose data is to be accessed), the user who wants to access the data is" "permitted to access the data, while if not, he or she is not permitted to access."). Further, Gladstone expressly states that stateful reference monitor 104 "cannot grant access when the standard stateless reference monitor" "denie[s] access." EX-1004 10:48-52. Thus, Gladstone in view of Salazar renders obvious *granting said access request only if … said minimum access level for said first base node does not exceed said predetermined access level*.

119. Therefore, Gladstone in view of Salazar renders obvious granting said access request only if it does not complete a prohibited temporal access pattern for said entity, and said minimum access level for said first base node does not exceed

said predetermined access level [1d] and denying said request if said access request completes a prohibited temporal access pattern for said entity [1e].

b. Claim 2

i. 2. The method according to claim 1, further comprising the step of denying said request if said minimum access level for said first base node exceeds said predetermined access level for said entity.

120. Gladstone, in view of Salazar, renders obvious the method of claim 1. *See* [1pre]-[1e]. Further, Gladstone, in view of Salazar, renders obvious claim 2 for the reasons stated above with respect to [1a, c-d]. *See* [1a, c-d].

c. Claim 3

i. [3pre] The method according to claim 1, further comprising the steps of:

121. Gladstone, in view of Salazar renders obvious the method according to claim 1. *See* [1pre]-[1e].

ii. [3a] logically organizing said computer system in the form of a tree hierarchy having a plurality of leaf nodes and higher-level nodes;

122. Gladstone discloses, or renders obvious alone or in combination with Salazar, this limitation.

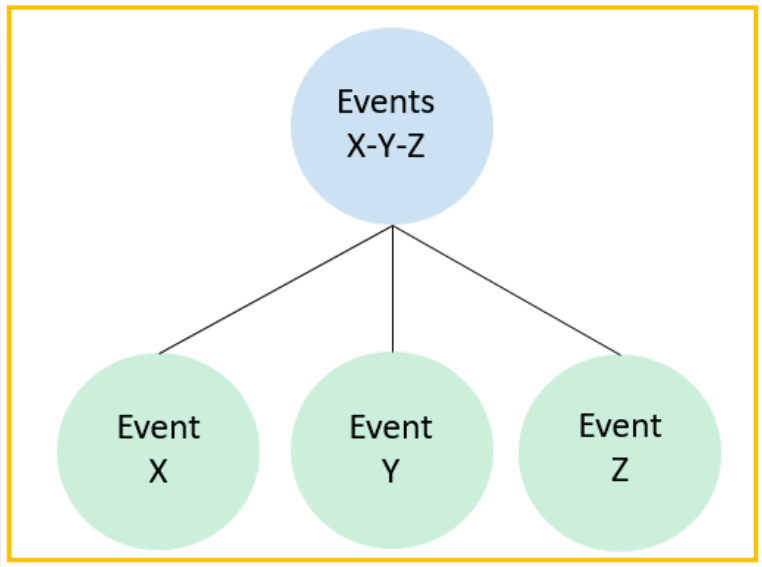
123. For example, Gladstone discloses a stateless monitoring system determining whether to allow or deny requests to resources—such as Events "X," "Y" and "Z" as discussed above in claim 1. *See*, *generally* [1pre]-[1e]. APOSA

would have understood that the association of Salazar's user properties with Gladstone's resources (as described above in [1c]), such as Event X, Event Y, and Event Z are examples of ***nodes*** consistent with the description of the '708 Patent. EX-1001 3:64-66 (describing "nodes" as "**representing information access and operational functions** for which the security operations are to be enforced"); [1c]. A POSA would have further understood each of **Events X, Y, and Z** to each be a ***leaf* node** as described by the '708 Patent because each (1) *represent[s] at least one of an information type and a computer system function* for which an access-control determination is made and (2) are aggregated together into higher-level nodes (*see below*). Therefore, Gladstone's resources, such as those represented by each of Events "X," "Y" and "Z," associated with Salazar's user properties are *a plurality of leaf nodes*:

124. Gladstone further describes access control decisions based on the combined occurrence of each of Events "X," "Y" and "Z" within a certain timeframe. *See, generally* [1pre]-[1e]. A POSA would have understood that such a combination of a plurality of leaf/base nodes is a higher-level node as described

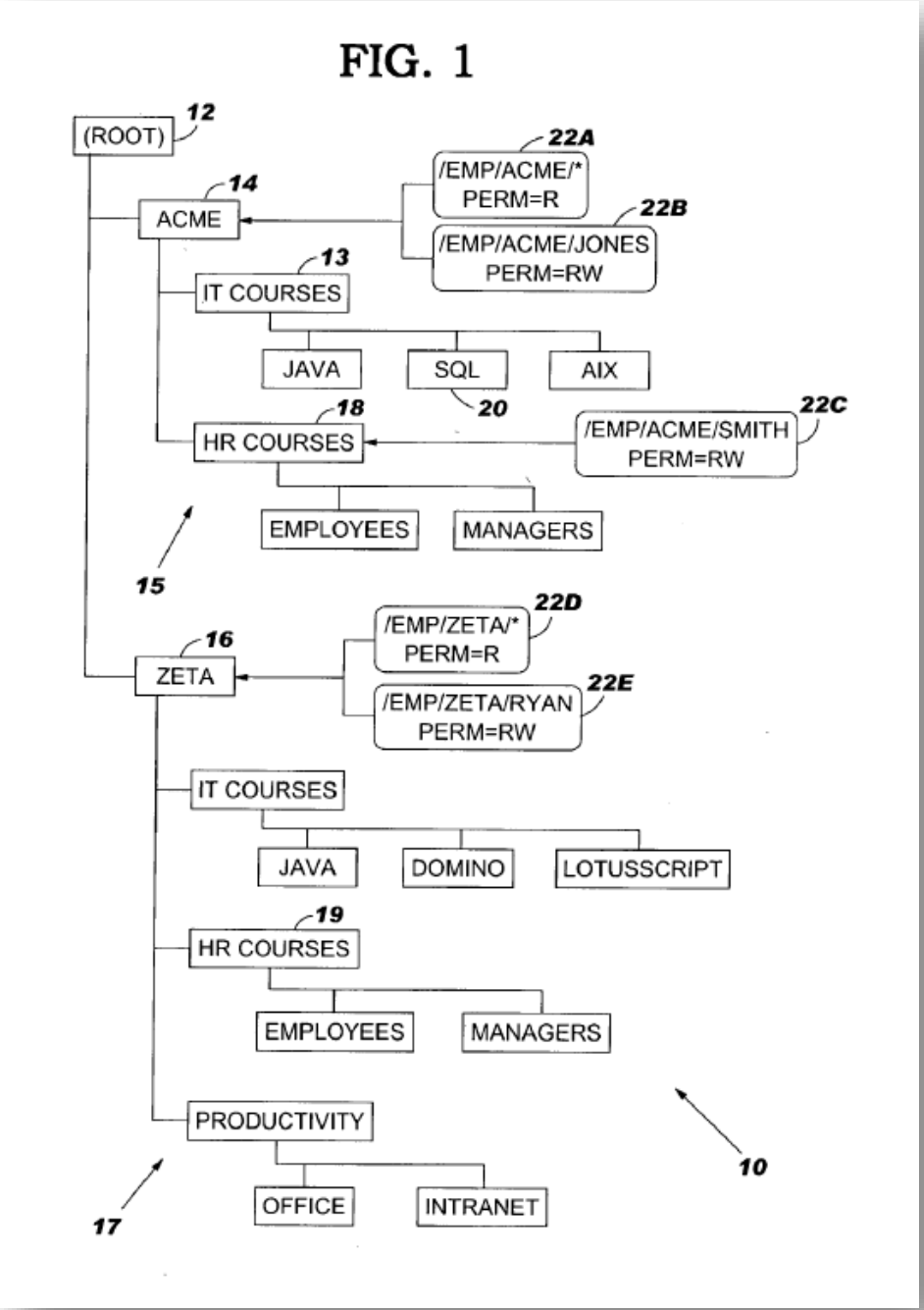
by the 708 Patent. EX-1001 3:2-6 ("Higher level nodes within the graph represent information aggregation sets and/or temporal patterns of access."). A POSA would have also found it obvious to include other higher-level nodes where needed as a mere duplication of this express example of Gladstone. Therefore, Gladstone discloses *a plurality of … higher-level nodes*:

125. A POSA would have understood that Gladstone discloses, or at least found it obvious, that Gladstone's *plurality of leaf nodes and higher-level nodes* were *logically organizing said computer system in the form of a tree hierarchy*. As described above in claim 1, Gladstone's methods are applied to a *computer system*. *See generally* [1pre]-[1e]. Further, a POSA would have understood Gladstone's description of Events X, Y and Z and their combination as X-Y-Z as a *logical[] organiz[ation] … in the form of a tree hierarchy* these *leaf nodes and higher-level nodes*:



EX-1004 5:57-60 (disclosing Gladstone's systems and methods using "logic"). Therefore, Gladstone discloses, or renders obvious, *logically organizing said computer system in the form of a tree hierarchy having a plurality of leaf nodes and higher-level nodes*.

126. To the extent that Patent Owner argues Gladstone does not disclose *logically organizing said computer system in the form of a tree hierarchy*, a POSA would have found this obvious in view of Salazar. For example, Salazar discloses a typical hierarchy of nodes included a "root node" that is the initial (or top) node of the hierarchy. The root node had one or more "child nodes" that were located at the level immediately below the root node, and the root node was called the "parent node of the child nodes":



EX-1009 [0020], FIG. 1. Salazar also states that "hierarchy of nodes 10 is **useful** in categorizing data items so that they can be **efficiently** located and retrieved." EX-1009 [0023]. Salazar further discloses that its method of associating its access

entries with the nodes "provides a more **flexible**, and **efficient** implementation of managing access to data items arranged into a hierarchy of nodes." EX-1009 [0007]. A POSA would have recognized this efficiency as beneficial to the access control system and method described by Gladstone and, using this teaching, found it obvious to arrange Events X, Y and Z and their combination as X-Y-Z as a ***logical****[] organiz[ation] … in the form of a tree hierarchy* these *leaf nodes and higher-level nodes*. §V.F.2

iii. [3b] defining a plurality of said base nodes as comprising respectively a plurality of leaf nodes of said tree hierarchy; and

127. Gladstone discloses, or renders obvious alone or in combination with Salazar, this limitation for the same reasons as stated above for [3a], and because the '708 Patent uses the terms "leaf" and "base" (nodes) interchangeably. *See* [3a]; EX-1001 3:46-47 ("the various object types and system functions can be defined as a plurality of leaf or base nodes 110 in the hierarchical tree 100.").

iv. [3c] defining said higher-level nodes as aggregations of said base nodes.

128. Gladstone discloses, or renders obvious this limitation for the same reasons stated above for [3a]. *See* [3a].

d. Claim 4

i. 4. The method according to claim 3 further comprising the step of identifying within said hierarchy any higher-level nodes that are aggregations comprising said first base node.

129. Gladstone discloses, or renders obvious alone or in combination with Salazar, the method of claim 3 (*see* [3pre]-[3c]) *further comprising the step of identifying within said hierarchy* the aggregated node of Event X-Y-Z (*any higher-level nodes that are aggregations comprising*) Event X, Y, and Z (*said first base node*).

130. As explained above for claim 3, the combination of Events X-Y- Z is a combination—or *aggregation*—and therefore a *higher-level node*, of Events X, Y and Z, any of which may be the *first base node*, and, these higher-level and base nodes are arranged in a *hierarchy*. *See* [3pre]-[3c]. Gladstone also discloses that stateful monitor 104 is aware of a prohibited combination of X-Y-Z, records the occurrence of any of the events X, Y and Z, and blocks the last event of the combination, e.g., Z, when the other two have been performed. EX-1004 6:13-19. A POSA would have understood that, in performing the above, stateful reference monitor 104 is identifying the *aggregated* node associated with prohibited combination of events X-Y-Z when, e.g., the first base node associated with, *e.g.*, event Z is blocked. Therefore, Gladstone discloses, or at least renders obvious *the method according to claim 3 further comprising the step of identifying within said hierarchy any higher-level nodes that are aggregations comprising said first base node*.

e. Claim 5

i. 5. The method according to claim 4, further comprising the step of identifying within said hierarchy any nodes that comprise children of any generation of said higher-level nodes that are aggregations comprising said first base node.

131. Gladstone discloses, or renders obvious alone or in combination with Salazar, the method of claim 4 (*see* claim 4) further comprising the step of identifying within said hierarchy any nodes that comprise children of any generation of said higher-level nodes that are aggregations comprising said first base node. As explained above, Gladstone discloses that stateful monitor 104 is aware of a prohibited combination of X-Y-Z, records the occurrence of any of the Events X, Y and Z, and blocks the last event of the combination, e.g., Event Z, when the other two have been performed. EX-1004 6:13-19. A POSA would have understood that blocking Event Z and the recognition of the occurrence of Events X and Y is identifying within said hierarchy Events X, Y, and Z (any nodes that comprise children of any generation) of prohibited combination of X-Y-Z (said higher-level nodes that are aggregations comprising said first base node.).

f. Claim 9

i. [9pre] A method for restricting access to a computer system having a plurality of logical base nodes representing at least one of an information type and a computer system function, and a plurality of higher-level nodes arranged together with said base nodes in the

form of a tree hierarchy, comprising the steps of:

132. Gladstone discloses *a method for restricting access to a computer system* for the reasons stated above. *See* [1pre]. For example, Gladstone discloses "monitoring system events 100, e.g. access requests to a resource, and controlling the effect of those events, *e.g.*, blocking or allowing the requested access to a resource." EX-1004 5:6-9; *see also* [1pre]. Gladstone restricts access to computer system 300 as discussed above. *See* [1pre].

133. Gladstone discloses computer system 300 having resources for which access control decisions are made, *e.g.*, allowed or not allowed, which is consistent with the '708 Patent's description of nodes (*having a plurality of ... base nodes representing at least one of an information type and a computer system function*) as discussed above. *See* [1a], [3a] (*plurality*). Gladstone also discloses that its systems and methods use "logic." EX-1004 5:57-60. Therefore, a POSA would have understood that Gladstone discloses *a plurality of logical base nodes representing at least one of an information type and a computer system function*.

134. Gladstone discloses, or renders obvious alone or in combination with Salazar, that computer system 300 has *a plurality of higher-level nodes* (*e.g.*, prohibited combination of Events X-Y-Z) *arranged together with said base nodes* (*e.g.*, individual Events X, Y, and Z) *in the form of a tree hierarchy* as describe above. *See* [3a]

ii. [9a] receiving in said computer system a request from an entity with a predetermined access level for access to a first base node;

135. Gladstone, in view of Salazar, renders obvious this limitation for the reasons discussed above in [1a]. *See* [1a].

iii. [9b] determining if said access request completes a prohibited temporal access pattern for said entity;

136. Gladstone discloses this limitation for the reasons discussed above in [1b]. *See* [1b]

iv. [9c] comparing a minimum access level established for said first base node to said predetermined access level;

137. Gladstone, in view of Salazar, renders obvious this limitation for the reasons discussed above in [1c]. *See* [1c];

v. [9d] granting said access request only if it does not complete a prohibited temporal access pattern for said entity, and said minimum access level for said first base node does not exceed said predetermined access level; and

vi. [9e] denying said request if said access request completes a prohibited temporal access pattern for said entity.

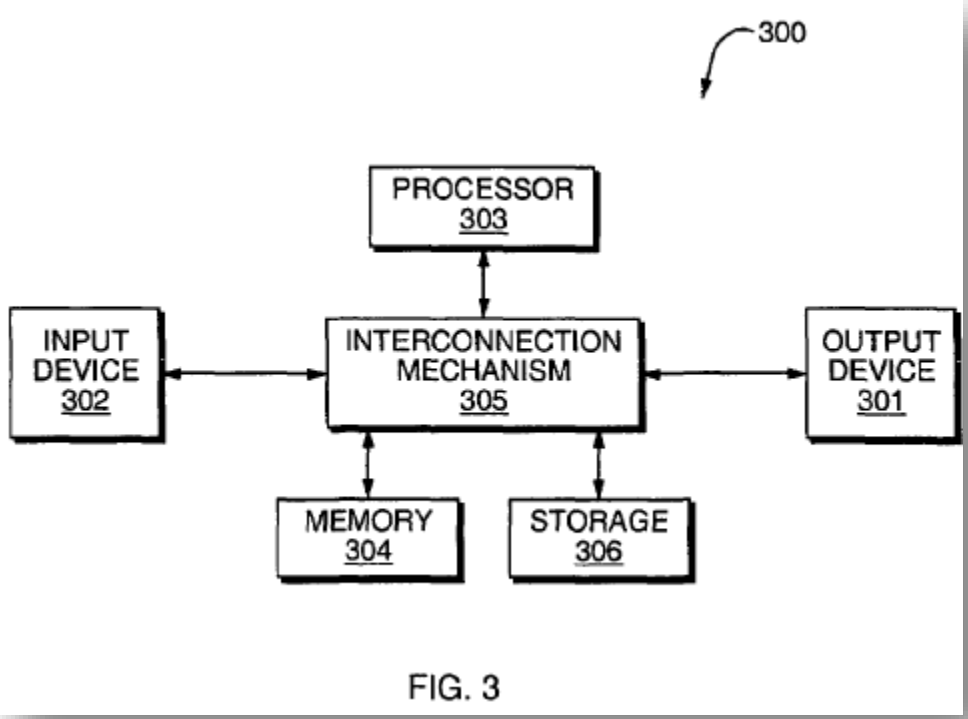
138. Gladstone, in view of Salazar, renders obvious these limitations for the reasons discussed above in [1d-e]. *See* [1d-e].

g. Claim 10

i. [10pre] A secure computer system comprising:

139. To the extent the preamble is limiting, Gladstone discloses implementing a security policy on computer system 300 (*a secure computer system*).

140. Gladstone discloses "controlling an operating policy of a computer system, such as a security policy." EX-1004 3:46-48. Gladstone states that an "important aspect of system security is monitoring system events 100, e.g. access requests to a resource, and controlling the effect of those events, e.g. blocking or allowing the requested access to a resource." EX-1004 5:6-9. Gladstone implements these methods for "computer system 300, shown in FIG. 3":



EX-1004 4:57-59 ("FIG. 3 is a functional block diagram of a hardware system on which the software system of FIGS. 1 and 2 can execute"), 14:20-15:12, FIG 3.

141. Thus, Gladstone discloses *a secure computer system*. Gladstone's *secure computer system* comprises the elements [10a]-[10f] as discussed below.

ii. [10a] a plurality of logical base nodes representing at least one of an information type and a computer system function;

142. Gladstone discloses making access control decisions for various types of resources (a plurality of logical base nodes representing at least one of an information type and a computer system function).

143. Gladstone discloses "a number of different resource types—files, system configuration information,"—*i.e.*, *information type[s]*—and "network access, system calls, local or remote object invocation, etc."—*i.e.*, *computer system function[s]*. EX-1004 9:43-55, 10:9-47 (describing various computer system functions and acknowledging that other functions and resources may need protecting); 6:16 (referencing generic resource requests as "Event X, Event Y and Event Z"); 6:64-7:3 (resources include "Outlook personal address books."). Each of these examples, *e.g.*, "Event X, Event Y and Event Z," is a base node:

144. Gladstone's methods determine "an action" for stateless reference monitor "to take, such as allowing" "or not allowing" "an access request." EX-1004 5:20-24. Therefore, Gladstone discloses access control decisions for various types of resources (*a plurality of logical base nodes representing at least one of an information type and a computer system function*) consistent with the description of the '708 Patent. EX-1001 2:67-3:2 ("The security access primitives can form base nodes in the hierarchical, directed graph."), 3:46-47 ("object types and system functions can be defined as" "leaf or base nodes 110"), 3:64-66 (describing "nodes" as "representing information access and operational functions for which the security operations are to be enforced"). Further, Gladstone discloses that its systems and methods used "logic" and, therefore, a POSA would understand that the resources to which access is controlled are "*logical*." EX-1004 5:57-60.

iii. [10b] a plurality of higher-level nodes arranged together with said base nodes in the form of a tree hierarchy;

145. Gladstone discloses, or renders obvious alone or in combination with Salazar, this limitation.

146. As described above in [10a], Gladstone discloses *base nodes* "Event X, Event Y and Event Z":

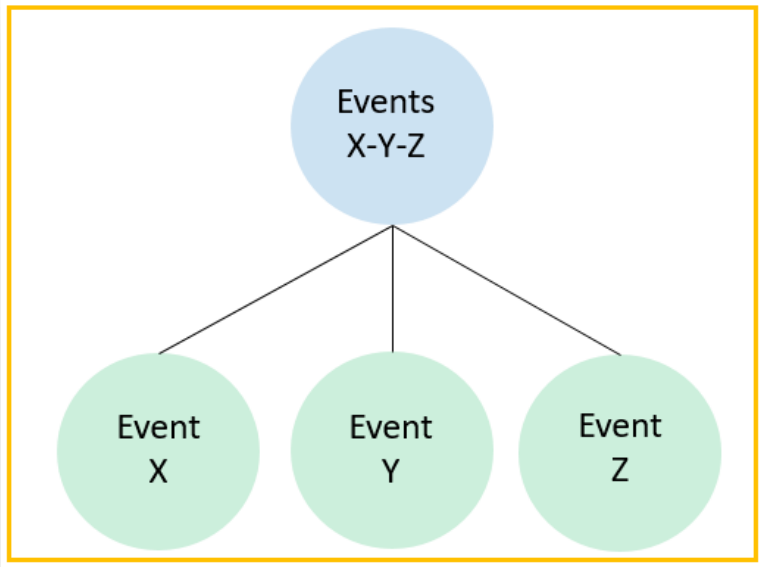
See [10a].

147. Gladstone further describes access control decisions based on the combined occurrence of each of Events "X," "Y" and "Z" within a certain timeframe. For example, a "prohibited combination of events" of "X-Y-Z" would "prohibit Z" if the request for Event Z occurred while "state database 207 holds information X-Y"—indicating that events X and Y had already occurred. EX-1004 6:13-14. A POSA would have understood that such a combination of base nodes:

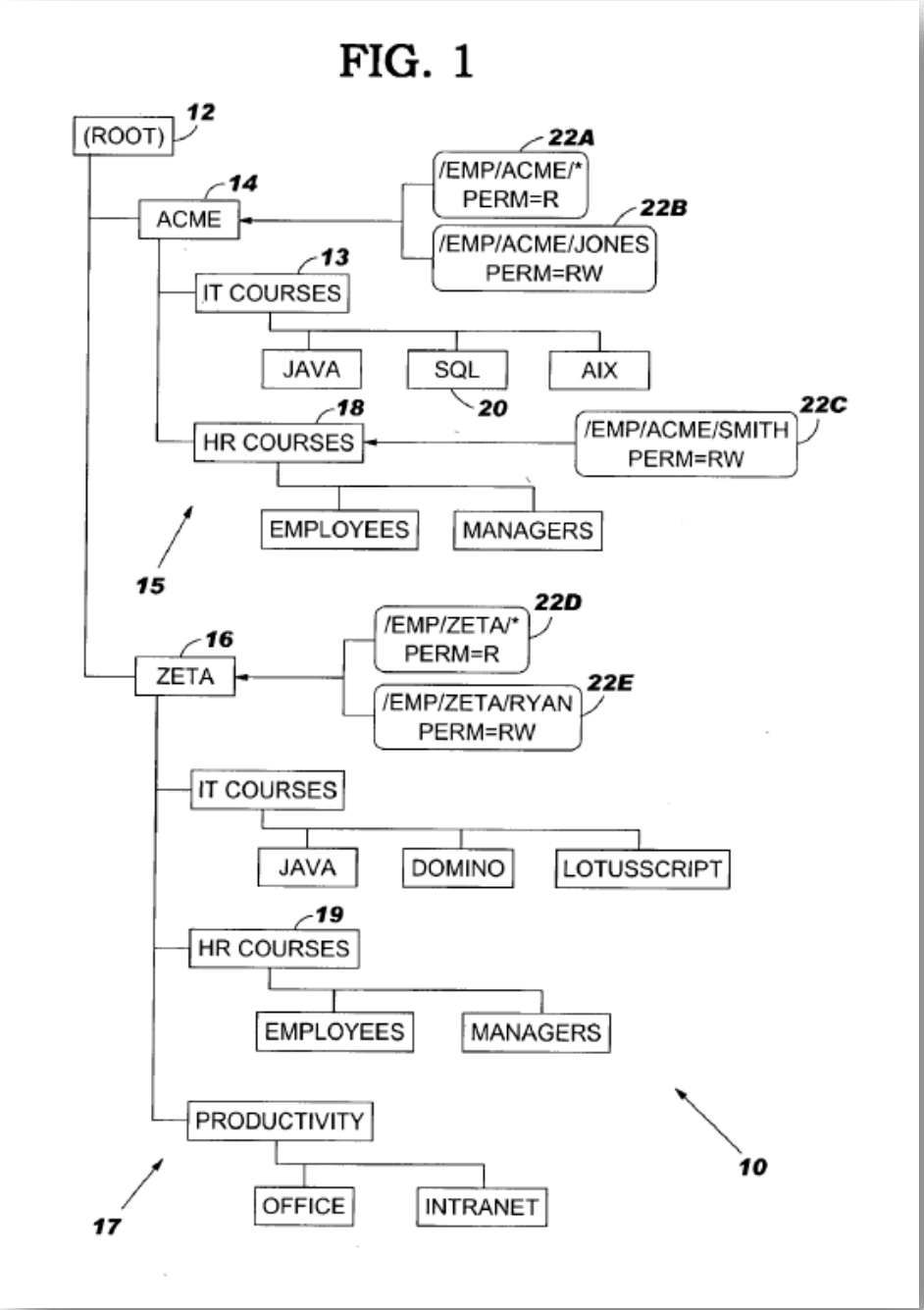
is a *higher-level node* as described by the 708 Patent. EX-1001 3:2-6 ("Higher level nodes within the graph represent information aggregation sets and/or temporal patterns of access."). A POSA would have also found it obvious to

include other higher-level nodes where needed as a mere duplication Gladstone's express example. Therefore, Gladstone discloses *a plurality of higher-level nodes*.

148. A POSA would have understood that Gladstone discloses, or at least found it obvious, that Gladstone's *plurality of higher-level nodes* and *said base nodes* were *arranged together… in the form of a tree hierarchy*. For example, Gladstone description of the separate requests of Events X, Y and Z and the prohibited combination of X-Y-Z to described a deliberate, knowing relationship between the individual events and the combined event. EX-1004 6:13-19. A POSA would have understood, or at least found obvious, to organize such information in the typical fashion—including in the computer security space—of using a *tree hierarchy*. §V.F.2. Thus, a POSA would have understood *plurality of higher-level nodes* and *said base nodes* were *arranged together… in the form of a tree hierarchy*:



149. To the extent that Patent Owner argues Gladstone does not disclose or render obvious a *plurality of higher-level nodes* and *said base nodes arranged together… in the form of a tree hierarchy*, a POSA would have found this obvious in view of Salazar. For example, Salazar discloses a typical hierarchy of nodes included a "root node" that is the initial (or top) node of the hierarchy. The root node had one or more "child nodes" that were located at the level immediately below the root node, and the root node was called the "parent node of the child nodes":



EX-1009 [0020], FIG. 1. Salazar also states that "hierarchy of nodes 10 is **useful** in categorizing data items so that they can be **efficiently** located and retrieved." EX-1009 [0023]. Salazar further discloses that its method of associating its access

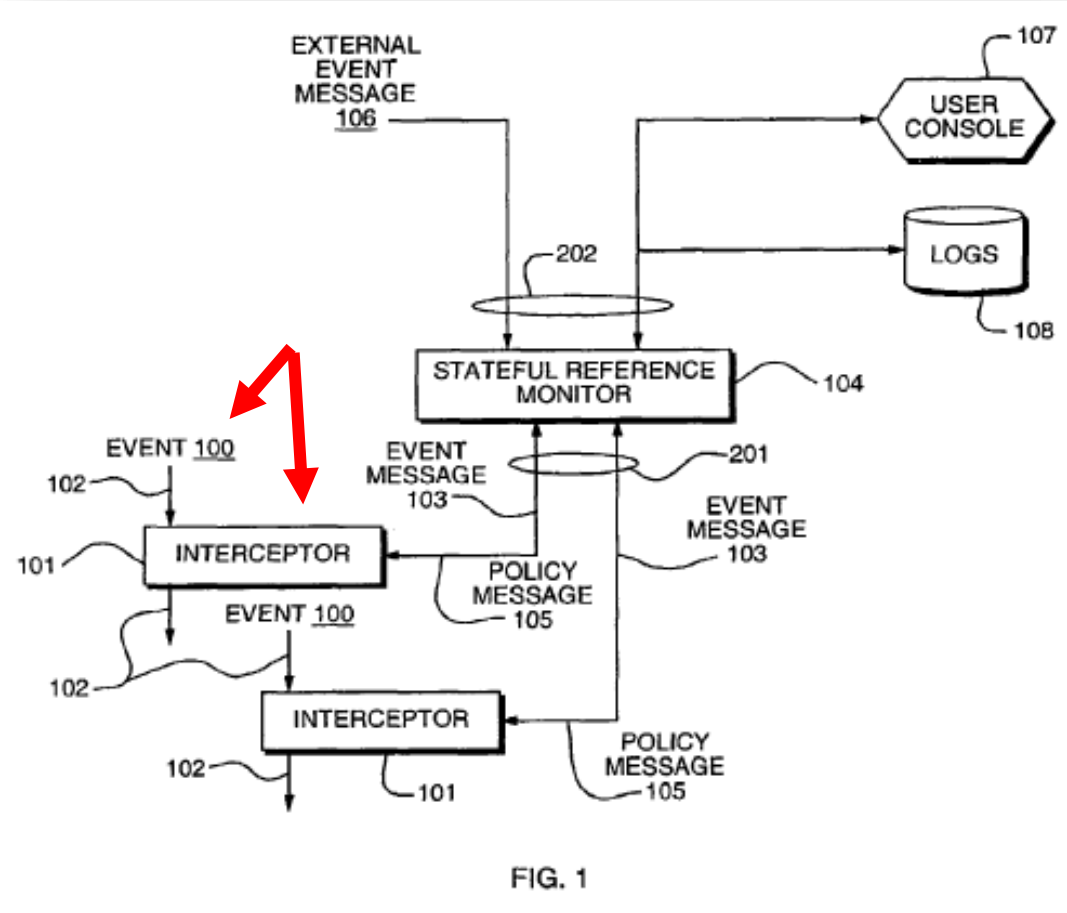
entries with the nodes "provides a more **flexible**, and **efficient** implementation of managing access to data items arranged into a hierarchy of nodes." EX-1009 [0007]. A POSA would have recognized this efficiency as beneficial to the access control system and method described by Gladstone and, using this teaching, found it obvious to *arrange* Events X, Y and Z (*base nodes*) and their combination as X-Y-Z (*higher-level node*) *in the form of a tree hierarchy*.

iv. [10c] a computer system interface capable of receiving a request from an entity with a predetermined access level for access to a first base node;

150. Gladstone discloses, or renders obvious alone or with Salazar, this limitation.

151. Gladstone discloses a *plurality of logical base nodes* as described above in [10a]. *See* [10a]. Any one of these base nodes can be the *first base node*.

152. Gladstone discloses intercepting (*receiving*) events 100 and event messages 103 (*a request… for access to a first base node*) by inceptor(s) 101 and stateless reference monitor and stateful reference monitor 104 of computer system 300 (*in said computer system*). For example, Gladstone discloses "**[i]nterceptors 101**" that "**intercept**[] various types of operating system **requests**, etc., **referred to herein as events**, for example requests for access to operating system resources such as a file or network connection":



EX-1004 6:37-41, FIG. 1 (annotated); *see also* EX-1004 5:5-6-7 ("system **events 100**, e.g. **access requests** to a resource"). After "an event is intercepted by an interceptor 101," it "generates an **event message 103**" that is "communicated to a **stateful reference monitor 104**" for stateful access control determinations:

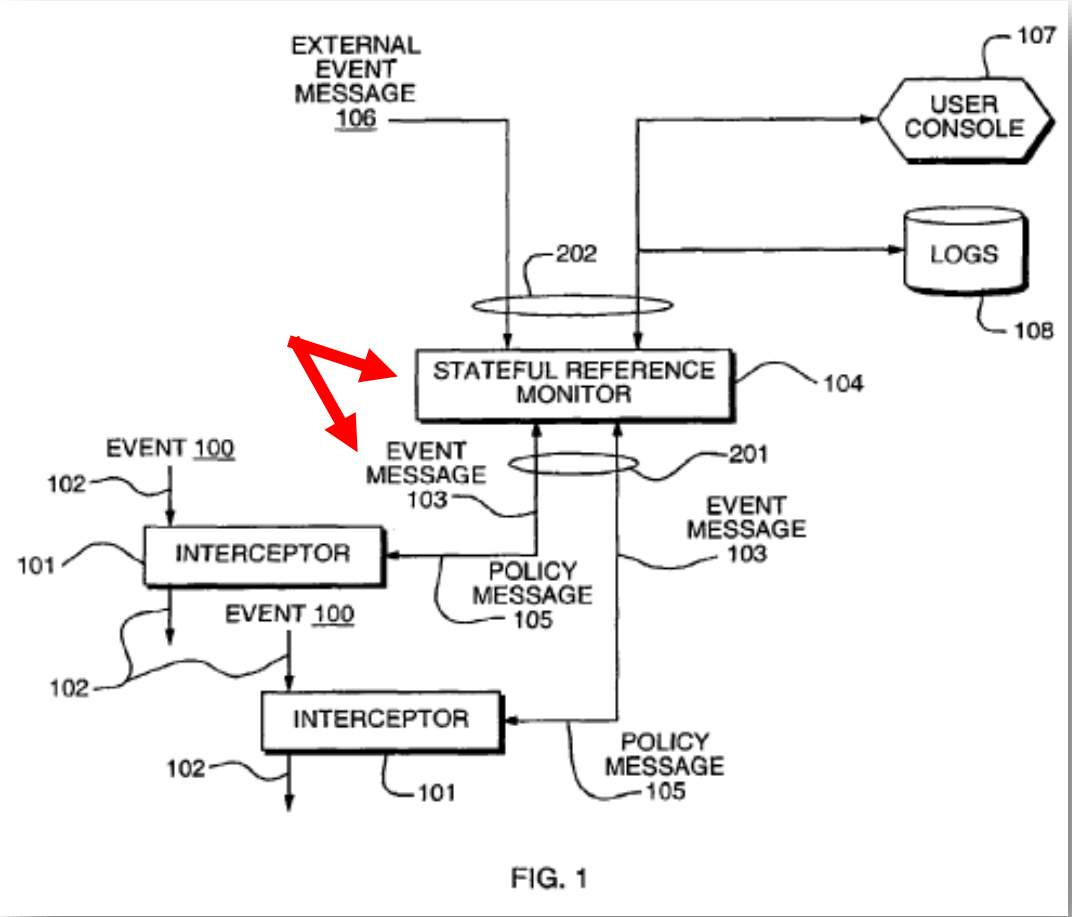


FIG. 1

EX-1004 5:16-24, 7:25-27, FIG. 1 (annotated). Both the intercepting of the event 100 by interceptor 101 and receiving the event message 103 by stateful reference monitor 104 are *receiving a request… for access*.

153. Gladstone also discloses that its access control methods utilize "[c]onventional" "stateless reference monitors." EX-1004 2:13-20. Gladstone discloses that "stateful reference monitor 104 acts as an **additional** processing step over and above any existing stateless reference monitor" and, "[t]hus, [the stateful reference monitor 104] cannot grant access when the standard stateless reference

monitor would have denied access." EX-1004 10:48-52. Consequently, a POSA would have understood, or at least found obvious, that the request for access associated with interception of event 100 and event message 103 would also have been *received* by Gladstone's integrated stateless reference monitor for its determination to allow or deny an access request prior to stateful reference monitor 104's evaluation of that request. EX-1004 10:48-52. Therefore, both of Gladstone's stateless and stateful reference monitors *receiv[e the] request… for access*.

154. As described above, interceptor 101, stateless reference monitor, and stateful reference monitor 104 each receive the request, are part computer system 300 and are therefore each *a computer system interface* as claimed. Gladstone further discloses that application programming interface (API) 203 of monitor as another interface that receives the requests. EX-1004 5:44-48.

155. Gladstone also discloses its requests are made by a user and user process (both, individually and combined, *an entity*). For example, Gladstone states that "reference monitor[s]"—such as its stateless reference monitors and stateful reference monitor 104—"determine[] whether an access by" "a user process" "is permitted." EX-1004 1:64-67. These "[u]ser processes are" "identified to users on the basis of the user supplying certain identity tokens." EX-1004 2:13-20. Gladstone also states that access requests, *e.g.*, "events X, Y and Z"

"may be" "associated" "with the same user"—i.e., *entity*. EX-1004 6:21-24. Both the user and user process are *an entity* as described by the '708 Patent. EX-1001 1:59 ("The entity can be a user or a process").

156. Gladstone's user and user process (both, individually and combined, *an entity*) are, as modified by Salazar, each an entity with "user properties" (*with a predetermined access level*). For example, Gladstone describes its stateless reference monitors are "[c]onventional references monitors" "used to determine whether a particular **access** to a file or other resource is permitted." EX-1004 2:1-12. In most cases, the stateless reference monitor's "access decision is" "based on the **identity of the user**" "mak[ing] the request." EX-1004 2:13-20. The access decision also references "permissions flags or an access control list associated with the resource," "individual permissions [associated] with each user," "a set of tokens" for "individual users," and the "**rights** and **privileges** attributed to [a] **user**." EX-1004 2:6-33, 8:2-3. Gladstone discloses that conventional "systems give finer control by associating individual permissions with each user." EX-1004 2:30-33. Gladstone further discloses that the "conventional reference monitors" were "found in the kernels of various Operating Systems." EX-1004 2:7-12.

157. Salazar discloses that when a user requests data items and/or attempts to perform an operation on data items, access is allowed, or not, "based on one or more particular user properties associated with user" (*a predetermined access*

level) and, separately, "the access entries." EX-1009 [0030-31]. Specifically, user properties are provided to an access system that separately obtains the access entries (*e.g.*, 22A-E) that then bases the access determination "on the one or more access entries that are **matched** with [the] user" using the property associated with the user (*a predetermined access level*) and the access entry. EX-1009 [0032-34].

158. Salazar discloses that these "user properties" broadly cover **any** property or characteristic that may be associated with a requesting user. For example, Salazar discloses that the "user property can be **any** property associated with a set of users such as a path/directory, a group designation, a user attribute, etc." EX-1009 [0024]. This includes both a "group specific property or a **personal** property associated with [the] **user**." EX-1009 [0029] (internal quotes omitted). User properties could further cover a user's department, country, or role. EX-1009 [0034] ("Dept=ACME and Country=USA," "isManager").

159. A POSA would have understood that Salazar's broad disclosure of the "user property" that "can be **any** property" included the well-known and widely implemented security or classification levels (i.e., *access levels*) used in prior art access control systems, which were known to implement access rules that "compar[ed]" a user's "clearance/authorization," "security level," and/or "hierarchical classification"—all examples of a *predetermined access level*—"to" the "[a]ccess control labels"—the "sensitivity level (e.g., classification)" or the

minimum access level—"associated with objects" to determine whether to grant access. EX-1013 9-10, 60, 65, 111. EX-1014 [0032-37, 41], FIG. 1 (providing access control based on "sensitive," "classified," classifications of information and permissions of a user); EX-1007 Abstract, 2:5-9, 2:21-28, 4:23-26, 4:38-51 ("security rank" assigned to users that determined whether access to information would be allowed or denied, where, *e.g.*, "the top rank is A and the rank goes down one by one from A to Z"), FIG. 3:

FIG.3

430 USER ID TABLE

431 USER ID	432 SECURITY RANK	433 ATTRIBUTE OF ANOTHER USER
C001	A	BELONGING,POSITION AND THE LIKE OF USER
C002	A	BELONGING,POSITION AND THE LIKE OF USER
C003	B	BELONGING,POSITION AND THE LIKE OF USER
C004	C	BELONGING,POSITION AND THE LIKE OF USER
C005	C	BELONGING,POSITION AND THE LIKE OF USER
B001	C	GUEST
B002	D	GUEST

EX-1016 017 ("[a]n **aggregate** of data might require a **higher classification** than the individual items", EX-1001 1:24-27 (describing the "inevitabl[e]" and

"natural[]" understanding that aggregating information changes sensitivity levels); *see generally* §V.F.1.

160. A POSA would have found it obvious to combine Salazar's user properties (*a predetermined access level*) with Gladstone's user and user process (both, individually and combined, *an entity*). Both disclose access control decision systems that provide "fine control" by associating data with each user used in the access control decision. EX-1004 2:30-31; EX-1009 [0005]. Further, Salazar states that its fine control is enabled by managing access control at the "application level" (EX-1009 [0005])—the same level at which Gladstone's stateful reference monitor 104 operates (EX-1004 6:24-33). A POSA would have recognized Salazar as describing the details of an example of a stateless reference monitor as described by Gladstone, and would have recognized that this fine-tuning of access control would benefit Gladstone's system of implementing access control based on sequences of events at the user level, and thus motivated to make such modification. A POSA would have had a reasonable expectation of success because comparative access-control systems where a "common security management" requiring nothing more than "simple relational comparison," as admitted by the '708 Patent, well within the skill of the POSA. EX-1007 4:37-44; EX-1001 1:25-28.

161. Finally, a POSA would have understood that Salazar's user properties, including multi-level security ranks and/or classifications levels, would have been associated with Gladstone's users prior to the access control decision and, therefore, are *predetermined*. For example, Salazar states that its user properties 40 are "set[]" "stored" and later "retriev[ed]" by various systems. EX-1009 [0029, 33].

v. [10d] a temporal access table;

162. Gladstone discloses a state database 207 (*a temporal access table*). For example, Gladstone discloses that stateful reference monitor 104 contains a "state database 207 [that] is simply a repository of dynamic state information":

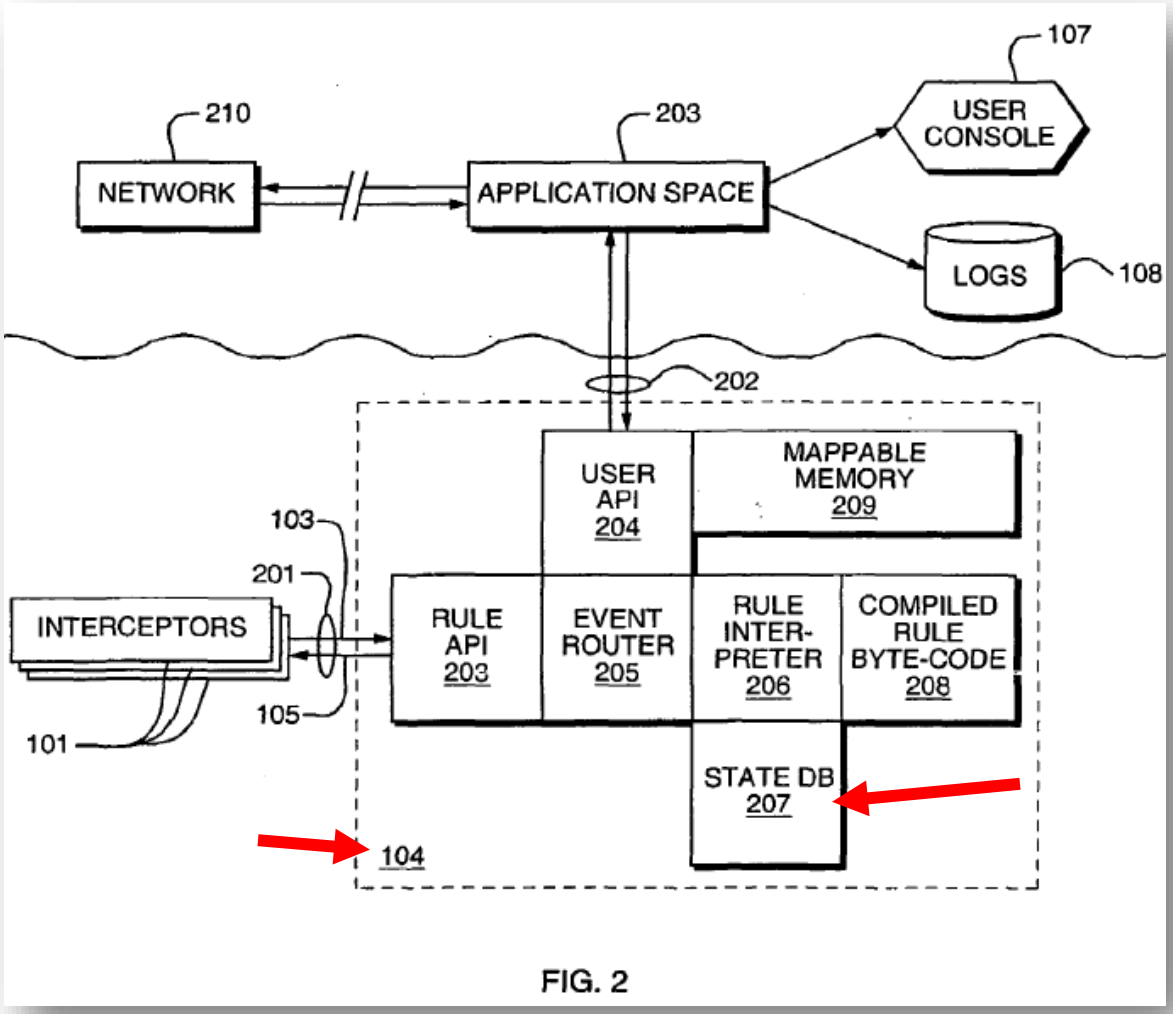


FIG. 2

EX-1004 5:58-67, FIG. 2 (annotated). "State information" is, "but is not limited to, information that a certain event occurred within a certain time of the present time." EX-1004 6:13-15. Gladstone provides an example of the recordation of these events (*i.e.*, requests for access to a resource) and the times at which they occur: "Suppose a prohibited combination of events is denoted X-Y-Z. That is, Event X, Event Y and Event Z must occur within a certain timeframe to produce a bad result. Then, if the state database 207 holds information X-Y, a rule in byte-code

208 will prohibit Z. But, another rule in byte-code 208 will cause event X and/or event Y to be purged from state database 207 after a suitable time." EX-1004 6:15-21; 8:5-13 ("information stored, may include some or all of, but not necessarily limited to, **time of the event, type of event, application generating the event, user owning the application and the like.**").

163. A POSA would have understood that Gladstone's description and use of state database 207 to be the same as '708 Patent's description and use of a temporal access table. First, a POSA would have understood a database, such as Gladstone's state database 207, to be a table. Second, Gladstone's state database 207 is used in the same way as the '708 Patent describes the Temporal Access Table shown in Figure 4B:

Temporal Access Table (TAT)

Usr	A	B	C	D	dir	exec	del
1(1)					102	112	
2(1)							
3(2)					111	103	
4(1)							
5(3)							

EX-1001 FIG. 4B. Like Gladstone's state database 207, the '708 Patent's Temporal Access Table maintains a history of the time at which operations

performed. EX-1001 5:55-57. For example, as shown in Figure 4B (above), user 1 performs a "dir" operation at time "102" and then requested an "exec" operation at time "112." EX-1001 5:62-66. A POSA would have understood that because the '708 Patent's Temporal Access Table is used in the same way as Gladstone's state database 207, Gladstone discloses *a temporal access table*.

vi. [10e] processing means programmed for comparing said access request to said temporal access table to determine if said access request completes a prohibited temporal access pattern for said entity, and for comparing a minimum access level established for said first base node to said predetermined access level; and

164. Gladstone, in view of Salazar, renders obvious this limitation.

165. Gladstone discloses *a processing means*. For example, Gladstone discloses that its "invention relate[s] generally to software that controls an operating policy of a computer system"—such as general purpose computer system 300 having processor 303—to, *e.g.*, control "access to various system resources, such as files and network connections" by performing the functions below, as modified by Salazar where stated. EX-1004 1:22-25, 3:44-4:9 ("software product"), 14:20-57, FIG. 3. To the extent this limitation is governed by means-plus-function, Gladstone's general purpose computer system 300 adapted by this software, as modified by Salazar where stated, to carry out algorithms performing the respective functions discussed below is the structure indicated in the '708

Patent or equivalent thereof (§V.C) and, therefore, Gladstone, modified by Salazar where stated, discloses *a processing means* under any reasonable construction.

166. As discussed above in [10c-d], Gladstone discloses events 100 and event messages 103 (both, individually and combined, *said access request*) and state database 207 (*temporal access table*) in the same manner as the '708 Patent. *See* [10c-d]. As also discussed, Gladstone utilizes its state database 207 in the same manner as the temporal access table in the '708 Patent (*see* [10d]) and, therefore, likewise discloses *comparing said access request to said temporal access table*.

167. To the extent that *comparing* requires the '708 Patent's temporal order table or equivalent structure (*see* §V.C.i), Gladstone discloses, or renders obvious, this same or equivalent structure. For example, Gladstone discloses that it is "a rule in byte-code 208" that "prohibit[s] Z" "if the state database **207** holds information X-Y" EX-1004 6:18-19. Further, "various types of patterns of events can be programmed into the rules byte-code 208," including rules generated in a "state collection mode, wherein normal combinations and patterns of activity are stored in state database 207." EX-1004 9:11-27. A POSA would have understood, or at least found obvious, that for the "various types of patterns of events" analyzed by Gladstone, the rules would be organized in the same form and format as the means by which those patterns were collected. Specifically, such rules would be contained in a corresponding table of the "various types of patterns of events"

because such rules were generated from just such a table—state database 207. *Id.*, *see* [10d]. Further, a POSA would have understood that a table is one of the well-known options for organizing relational data, and would have found it obvious to organize byte-code 208 as a table as a simple design choice.

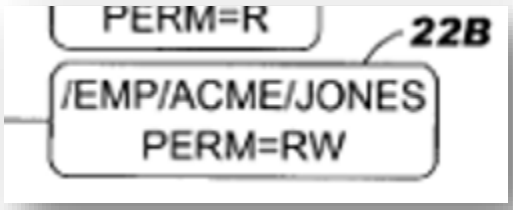
168. Gladstone discloses, or renders obvious alone or with Salazar, stateful reference monitor 104 using this comparison when *determin[ing] if*, for events 100 and event messages 103 (both, individually and combined, *said access request*), a sequence of actions should be denied (*completes a prohibited temporal access pattern*) for the user or user process (*for said entity*).

169. For example, Gladstone discloses that "various types of patterns of events can be programmed" "for detection and action" by the stateful reference monitor 104. EX-1004 9:11-27. Gladstone's methods "track a **sequence of actions**" and can allow or deny requests. EX-1004 13:8-10. Such "sequences are used for detecting malicious behavior" and "normal behavior." EX-1004 13:13-21. This functionality enables the "stateful reference monitor [to] ensure[] that requests are ordered, i.e. **a request either happens before or after another request**"—*i.e.*, that a *temporal access pattern* occurs or does not occur. EX-1004 12:22-28. For example, a "prohibited combination of events" of "X-Y-Z" would "prohibit Z" if the request for Event Z occurred while "state database 207 holds information X-Y"—indicating that events X and Y had already occurred. EX-

1004 6:13-14. While Gladstone discloses that event Z could be blocked based on the combination of events X-Y-Z, regardless of the order in which the events occurred, Gladstone also discloses "ensur[ing] that requests are **ordered**, i.e. a request either happens before or after another request" and "track[ing] a **sequence** of actions." EX-1004 12:22-28; 13:8-10. Thus, a POSA would have understood that Gladstone discloses blocking (denying the request) combinations of actions due to the particular "sequence of actions" of those events and, therefore, would have found it obvious for the stateful reference monitor 104 to require a particular ordering of events X, Y, and Z for the security policy to be applied to them. Therefore, a POSA would have found it obvious for Gladstone's stateful monitor to deny access to event Z if a user requested access because the user had already requested access to event X and/or Y in a particular order. EX-1004 13:8-10, 12:22-28. Gladstone expressly describes that monitoring the order of actions was the very purpose of stateful monitors—such as stateful reference monitor 104—that observed the "dynamic state" of computer systems using information "collected in real-time, indicative of a condition of a machine or process as a result of a **particular sequence of events**." EX-1004 2:1-4. These disclosures of Gladstone also correspond to the '708 Patent's description of *temporal access patterns*: "**temporal ordering** of primitive access operations mandates that the

operations occur **in the order specified** for the security policy to be enforced." EX-1001 5:43-45.

170. To extent that the *a prohibited temporal access pattern* requires the algorithm and structure comparing *access levels* (§V.C.i), this is obvious over Gladstone in view of Salazar. As discussed above for [10c], Salazar discloses user properties, including "sensitivity level"— *a minimum access level*—to determine whether to grant access. *See* [10c]. A POSA would have found it obvious to combine Salazar's user properties with Gladstone's resources for the same reason as described with respect to the user or user processes. *See* [10c]. For example, Salazar includes the user properties in the access entries. Salazar associates "access entries," *e.g.*, access entries 22A-E, with various nodes of hierarchy 10 in Figure 1. EX-1009 [0024], FIG. 1. Each access entry 22A-E associates a "user property" with what Salazar describes as an "access level." EX-1009 [0024]. For example access entry 22B associates the user property "/EMP/ACME/JONES" with a "read/write" "access level" (i.e., "PERM=RW"):



EX-1009 [0027], FIG. 1.

171. As described above in [10c], a POSA would have understood that Salazar's broad disclosure of the "user property" that "can be any property" included the well-known and widely implemented security or classification levels (*i.e.*, access levels) used in prior art access control systems. *See* [10c].

172. A POSA would have found it obvious to use Salazar's "user properties" in place of Gladstone's one or more static permissions flags or as part of or in place of an access control list associated with the resource (as modified, *a minimum access level established for said first base node*) for the same reasons discussed above in [10c]. *See* [10c]. Both disclose access control decision systems that provide "fine control" by associating data with each user used in the access control decision. EX-1004 2:30-31; EX-1009 [0005]. Further, Salazar states that its fine control is enabled by managing access control at the "application level" (EX-1009 [0005])—the same level at which Gladstone's stateful reference monitor 104 operates (EX-1004 6:24-33). A POSA would have recognized Salazar as describing the details of an example of a stateless reference monitor as described by Gladstone, and would have recognized that this fine-tuning of access control would benefit Gladstone's system of implementing access control based on sequences of events at the user level, and thus motivated to make such modification. A POSA would have had a reasonable expectation of success because comparative access-control systems where a "common security

management" requiring nothing more than "simple relational comparison," as admitted by the '708 Patent, was well within the skill of the POSA. EX-1007 4:37-44; EX-1001 1:25-28.

173. Further, a POSA would have understood, or at least found obvious, that Salazar's "user properties" for resources and access control decisions utilizing the same also applied to aggregated, combined resource requests, *e.g.*, the prohibited combination of events X-Y-Z. This combination is an aggregation of events, which a POSA would have understood, or at least found obvious, to be subject to higher sensitivity or classification levels than the individual resources due to the aggregation or, nevertheless, subject to the sensitivity level of each individual resources request. EX-1016 017 ("[a]n **aggregate** of data might require a **higher classification** than the individual items"); EX-1001 1:24-27 (describing the "inevitabl[e]" and "natural[]" understanding that aggregating information changes sensitivity levels). A POSA would have been motivated to implement the "user properties" with an aggregation of events, *e.g.*, prohibited combined event X-Y-Z, for the prohibited for the same reasons provided above for the individual resources, and would have reasonably expected this combination to be successful as requiring nothing more than routine skill and a simple relational comparison.

174. Therefore, Gladstone discloses, or renders obvious alone or with Salazar, *determining if said access request completes a prohibited temporal access pattern*, and applying its security policy based on the same.

175. Gladstone also discloses that the blocked sequence of actions (*prohibited temporal access pattern*) is specific to a user and user process (*for said entity*). As discussed above for [10c], Gladstone discloses user and user process (both, individually and combined, *said entity*) having, as modified by Salazar's user properties (*predetermined access level*) used in access determinations. *See* [10c]. Further, Gladstone states that the combination of "[e]vents X, Y and Z" "may be" "associated" "with the same user."[5] EX-1004 6:21-23. When the user or user process (*said entity*) requests a particular sequence of events (e.g., events X-Y-Z) (*temporal access pattern*) that is blocked, a POSA would have understood that blocking to be directed at the user and, therefore, the blocked sequence of

---

[5] Gladstone also discloses that access control determinations based a sequence of events (*temporal access pattern*) could be made independent of the user (EX-1004 7:62-66), an embodiment not contemplated by the '708 Patent, further illustrating that the prior art had far surpassed the purported solutions that the '708 Patent describes before the March 2004 priority date. §V.F.4.

events is a *prohibited temporal access pattern for* the user or user process (*said entity*).

176. To the extent that Gladstone does not expressly disclose blocking a sequence of events (*prohibited temporal access pattern*) specific to a particular user or user process (*for said entity*), a POSA would have nevertheless found doing so obvious. Gladstone broadly discloses that "reference monitor[s]"— including stateful reference monitor 104—"determine whether an access by" "a user process" "is permitted." EX-1004 1:64-67; 6:21-23. These determinations "[i]n most cases" are "based on the identity of the user." EX-1004 2:13-20. Gladstone describes that conventional security "systems give finer control by associating **individual permissions with each user**." EX-1004 2:30-21. Thus, a POSA would have found it obvious to use both the "identity of the user" and the associated "individual permissions" in determining whether to block a sequence of events (*prohibited temporal access pattern*) for a particular user or user process (*for said entity*). EX-1010 2:63-3:1 (integrating user usage patterns in security rules); EX-1011 43-44 (determining consistency of "characteristic patterns of actions," of users.). A POSA would have been motivated to include this individualized access control capability in Gladstone's stateful monitor 104 to achieve the same effect—provide access control based on the individual. Further, a POSA would have had a reasonable expectation of success because, well before

March 2004, "individual[ized] permissions" were already used by access control systems and that "operat[ing] on the basis of the user" "and the rights and privileges attributed to that user" was "conventional." EX-1004 2:30-21, 7:66-8:3; §V.F.1. Therefore, Gladstone discloses or renders obvious *determin[ing] if said access request completes a prohibited temporal access pattern for said entity.*

177. Gladstone, in view of Salazar, renders obvious making access control determinations by matching (*comparing)* the "user properties" of Gladstone's resources (*a minimum access level established for said first base node*) to Salazar's user properties associated with Gladstone's user and user process (*said predetermined access level*).

178. For example, Gladstone discloses that a "**reference monitor** is a component of a computer system that **determines** whether an access" "**is permitted**." EX-1004 1:64-67, 2:7-12. For example, Gladstone discloses that "[i]n most cases the access decision is made **based on** the identity of the user" "and one or more static permissions flags or an access control list associated with the resource" and matching in the context of access control lists. EX-1004 2:13-20, 2:30-33, 2:36-39. Likewise Salazar discloses that the access control decisions are "based on the one or more access entries that are **matched** with [the] user" using the property associated with the user and the access entry. EX-1009 [0032-34].

179. Gladstone, in view of Salazar, renders obvious implementing Salazar's "user properties" with Gladstone's resources (*a minimum access level established for said first base node*) for the reasons stated above.

180. Therefore, Gladstone, in view of Salazar, renders obvious comparing a minimum access level established for said first base node to said predetermined access level.

vii. [10f] wherein said processing means denies said request if said access request completes a prohibited temporal access pattern for said entity and grants said access request only if it does not complete a prohibited temporal access pattern for said entity, and said minimum access level for said first base node does not exceed said predetermined access level.

181. Gladstone, in view of Salazar, renders these elements obvious.

182. As discussed above, Gladstone, modified by Salazar where stated below, discloses or renders obvious processor 303, and a general purpose computer 300 adapted with an algorithm to perform the function discussed below, and, therefore, discloses *a processing means* having the same or equivalent structure indicated in the '708 Patent (§V.C). *See* [10e].

183. As discussed above, Gladstone discloses that "stateful reference monitor 104 makes a decision" "to block or allow" a sequence of events (*prohibited temporal access pattern*) for the user or user process (*for said entity*). *See* [10e]. For example, Gladstone discloses that if "a prohibited combination of

events is" "X-Y-Z," stateful reference monitor 104 would "prohibit Z" if it had recorded in "state database 207" "[events] X-Y." EX-1004 6:13-24; *see* [10e]. Conversely, a POSA would have understood that Gladstone's monitor 104 would have allowed event Z if it did not complete a *prohibited temporal access pattern*, *e.g.*, if the combination of events X-Y-Z were allowed for the user, if either of events X and Y had not been completed, if the events occurred in some other order, or if the user had a sufficient clearance for the combination of events. Therefore, Gladstone discloses or renders obvious *said processing means den[ying] said request if said access request completes a prohibited temporal access pattern for said entity* and *grant[ing] said access request only if it does not complete a prohibited temporal access pattern for said entity*.

184. Gladstone, in view of Salazar, renders obvious *granting said access request only if …* Salazar's user properties associated with the Gladstone's resource (*said minimum access level established for said first base node*) also do not exceed the user property assigned to the user and/or user process (*said predetermined access level*). *See also* [10c, 10e]. As discussed above, a POSA would have understood that Salazar's "user property" included the well-known and widely implemented security or classification levels (*i.e.*, *access level[s*]) used in prior art access control systems. *See* [10c, 10e]; EX-1013 9-10 ("what classes of information [users] are authorized to deal with" for "for handling sensitive (e.g.,

classified) information"); EX-1014 [0032-37, 41], FIG. 1 (providing access control based on "sensitive," "classified," classifications of information and permissions of a user); EX-1007 Abstract, 2:5-9, 2:21-28, 4:23-26, 4:38-51, FIG. 3 (disclosing a "security rank" assigned to users that determined whether access to information would be allowed or denied, where, *e.g.*, "the top rank is A and the rank goes down one by one from A to Z."); §V.F.1. A POSA would have understood that requiring a "match" of user properties for (*see, e.g.*, EX-1004 1:38-2:44; EX-1009 [0026-27, 34-35]) would mean that *said minimum access level for said first base node* ***does not exceed*** *said predetermined access level* as "***does not exceed***" is used by the '708 Patent. EX-1001 3:36-41 ("The number represents a minimum security access level required for any process or user to access that particular function. Thus, for example, the "exec system function labeled exec(2) requires that a process or user possess a security level of at least two in order to access that function."). Further, a POSA would have understood that Gladstone in view of Salazar would allow access when the user property assigned to the user (*predetermined access level*) exceeded the user property assigned to the resource (*minimum access level for said first base node*). EX-1007 4:38-51 (disclosing a system where "the top rank is A and the rank goes down one by one from A to Z" and "if (the security rank of the user who wants to access the data)" is less than or equal to "(the security rank of the user whose data is to be accessed), the user who

wants to access the data is" "permitted to access the data, while if not, he or she is not permitted to access."). Further, Gladstone expressly states that stateful reference monitor 104 "cannot grant access when the standard stateless reference monitor" "denie[s] access." EX-1004 10:48-52. Thus, Gladstone in view of Salazar renders obvious *granting said access request only if ... said minimum access level for said first base node does not exceed said predetermined access level*.

185. Therefore, Gladstone in view of Salazar renders obvious wherein said processing means denies said request if said access request completes a prohibited temporal access pattern for said entity and grants said access request only if it does not complete a prohibited temporal access pattern for said entity, and said minimum access level for said first base node does not exceed said predetermined access level.

h. Claim 11

i. 11. The secure computer system according to claim 10, wherein said processing means denies said request if said minimum access level for said first base node exceeds said predetermined access level for said entity.

186. Gladstone in view of Salazar renders obvious *the secure computer system according to claim 10*. *See* [10pre]-[10f].

187. As discussed above, Gladstone, alone or in view of Salazar, discloses or renders obvious processor 303, and a general purpose computer 300 adapted with an algorithm to perform the function discussed below, and, therefore, disclosing *a processing means* having the same or equivalent structure indicated in the '708 Patent (§V.C). *See* [10e].

188. Gladstone also discloses denying said request if said minimum access level for said first base node exceeds said predetermined access level for said entity as discussed above. *See* Claim [10f].

i. Claim 12

i. 12. The secure computer system according to claim 10 wherein said higher-level nodes are aggregations of said base nodes.

189. Gladstone in view of Salazar renders obvious *the secure computer system according to claim 10*. *See* [10pre]-[10f].

190. Gladstone discloses, or renders obvious, *wherein said* prohibited combination of Events X-Y-Z (*higher-level node[]*) *[is an] aggregation[] of said* Event X, Event Y, and Event Z (*base nodes*) as discussed above. *See* [10b]. As discussed above, a POSA would find it obvious to extend the teaching of Gladstone to multiple prohibited combinations of events as mere repetition of Gladstone's example.

j. Claim 13

i. 13. The secure computer system according to claim 12 wherein said processing means identifies within said hierarchy any higher-level nodes that are aggregations comprising said first base node.

191. Gladstone in view of Salazar renders obvious *the secure computer system according to claim 12*. *See* Claim 12.

192. As discussed above, Gladstone, alone or in view of Salazar, discloses or renders obvious processor 303, and a general purpose computer 300 adapted with an algorithm to perform the function discussed below, and, therefore, disclosing *a processing means* having the same or equivalent structure indicated in the '708 Patent (§V.C). *See* [10e].

193. Gladstone discloses identif[ying] within said hierarchy any higher-level nodes that are aggregations comprising said first base node.

194. As explained above, the combination of Events X-Y- Z is a combination—or *aggregation*—and therefore a higher-level node, of Events X, Y and Z, any of which may be the *first base node*;, and, these higher-level and base nodes are arranged in a *hierarchy*. *See* [10b]. Gladstone also discloses that stateful monitor 104 is aware of a prohibited combination of X-Y-Z, records the occurrence of any of the events X, Y and Z, and blocks the last event of the combination, e.g., event Z, when the other two have been performed. EX-1004 6:13-19. A POSA would have understood that, in performing the above, stateful

reference monitor 104 is identifying the *aggregated* node associated with prohibited combination of events X-Y-Z when, *e.g.*, the first base node associated with, *e.g.*, event Z is blocked. Therefore, Gladstone discloses, or at least renders obvious *identif[ying] within said hierarchy any higher-level nodes that are aggregations comprising said first base node*.

k. Claim 14

i. 14. The secure computer system according to claim 13 wherein said computer processing means identifies within said hierarchy any nodes that comprise children of any generation of said higher-level nodes that are aggregations comprising said first base node.

195. Gladstone in view of Salazar renders obvious *the secure computer system according to claim 13*. *See* Claim 13.

196. As discussed above, Gladstone, alone or in view of Salazar, discloses or renders obvious processor 303, and a general purpose computer 300 adapted with an algorithm to perform the function discussed below, and, therefore, disclosing *a processing means* having the same or equivalent structure indicated in the '708 Patent (§V.C). *See* [10e].

197. Gladstone discloses, or renders obvious alone or in combination with Salazar *identif[ying] within said hierarchy any nodes that comprise children of any generation of said higher-level nodes that are aggregations comprising said first base node* as discussed above. As explained above, Gladstone discloses that

stateful monitor 104 is aware of a prohibited combination of X-Y-Z, records the occurrence of any of the Events X, Y and Z, and blocks the last event of the combination, e.g., Event Z, when the other two have been performed. EX-1004 6:13-19. A POSA would have understood that blocking Event Z and the recognition of the occurrence of Events X and Y is identif[ying] within said hierarchy Events X, Y, and Z (any nodes that comprise children of any generation) of prohibited combination of X-Y-Z (said higher-level nodes that are aggregations comprising said first base node.). *See* [10e].

B. Ground 2: Claim 6 Is Obvious Over Gladstone in View of Salazar and Hada (EX-1005)

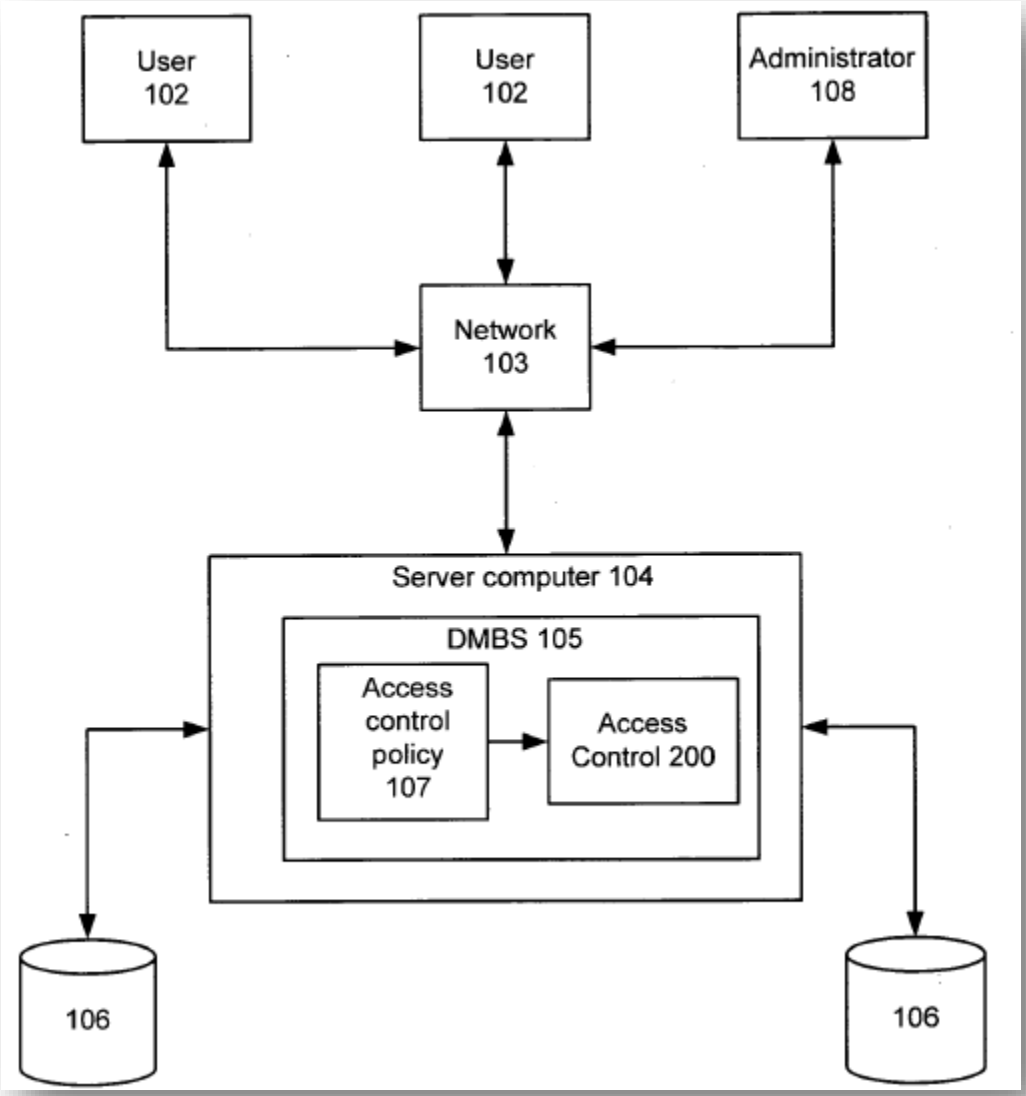
1. Hada

a. Analogous Art

198. Hada is analogous art to the claimed invention. For example, the field of endeavor of the '708 Patent's claimed invention is "generally [directed] to computer and information security and more particularly to security measures that enforce security based on logical implementation methods." EX-1001 1:8-11, Abstract, 2:65 ("controlling access"). Hada is in the same field of endeavor as the '708 Patent because it "relates generally to computer implemented database systems and, more particularly, to a method and system for providing path-level access control." EX-1005 [0001].

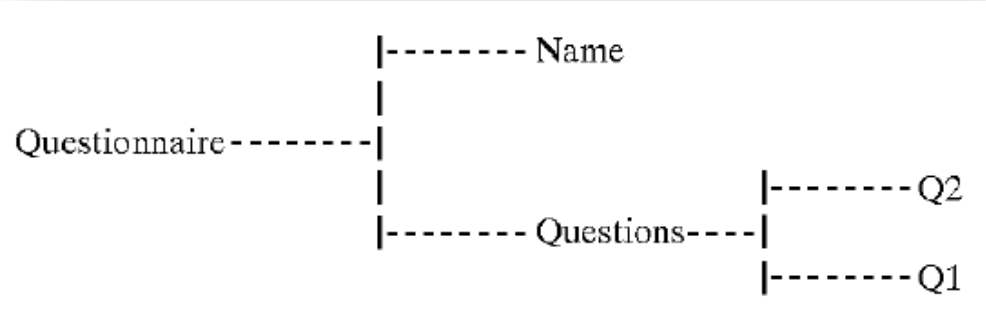
b. Overview of Hada

199. Hada is directed to "[a]n improved method and system for providing path-level access control to a structured document in a collection stored in a database." EX-1005 Abstract, [0011]. Hada implements its teachings in a generic "computer environment," such as that shown in Figure 1:



EX-1005 FIG. 1. Hada describes that the structured document "is parsed into a plurality of nodes that form a node tree." EX-1005 [0021]. "Each node" of the

structured document "is then associated with a path that describes the node's hierarchical relationship to its parent node(s)," in the document:



EX-1005 [0021], [0033]. As an example, the "path to element node Q1 is" "/Questionnaire/Questions/Q1." EX-1005 [0033-34].

200. Hada also discloses "a database management system (DBMS) 105" that includes "access control policies 107" and "an Access Control mechanism 200." EX-1005 [0026]. "The access control policy 107" "comprises a plurality of access control rules." EX-1005 [0029]. "Each access control rule" "defines a subject"—a user's name or a group of users—"to which the rule applies, an action"—*e.g.*, read, update, create or delete action, etc.—"and a path" that "identifies the node"—the portion of the structured document—"to which the rule applies." EX-1005 [0029].

201. Hada discloses "Access Control mechanism 200 that provides path-level access control to structured documents." EX-1005 [0026]. Hada describes that a normalized form of an access control rule governing access to the document

is "propagates upward toward" "ancestor nodes" (that may be aggregations, e.g. "Questions" aggregating Q1 and Q2 as shown above) or down toward "descendant nodes," meaning that the action (e.g., read, update, etc.) and effect (grant, deny) for the user governed by a that rule is propagated to related nodes according to a set of rules, *e.g.*:

TABLE 1

Action	Effect	Mode	Semantics
Read	Grant	Down	Rule specified to any node propagates down.
		Up	Every GRANT for read-access propagates up to all ancestor elements (but not including the attributes and text nodes of these ancestor elements).
	Deny	Down	Rule specified to any node propagates down.
		Up	No propagation
Update	Grant	Down	Every GRANT for update-access propagates down to all the attributes and text nodes.

EX-1005 [0043]. Hada describes that Access Control mechanism 200 uses the access control policy 107 and paths to the nodes to create a "value expression"—"an executable statement regarding who, if anyone, has access to the node associated with the path." EX-1005 [0028], [0036]. Hada discloses that value expressions are optimized and consolidated to remove identical value. EX-1005 [0046]. Hada states that "the DBMS 105 checks the value expression corresponding to the requested node/path, as opposed to evaluating the access

control policy" leading to "more efficient and fast" access control decisions. EX-1005 [0068], [0075-76].

2. Motivation to Combine Gladstone, Salazar, and Hada

202. A POSA would have been motivated to combine Gladstone's access control method, as modified with Salazar's user properties, with Hada's path-level access control decisions. For example, Hada states that database security is a "crucial concern" (EX-1005 [0004]), Gladstone is broadly directed to protecting computer resources (EX-1004 Abstract), and Salazar (EX-1009 Abstract) discloses systems and methods for managing access to data items in a database, which a POSA would have understood to include databases and structured documents (information) and the computer functions associated therewith. Further, a POSA would have been motivated to implement Hada's path-level methods in modified-Gladstone to provide "more efficient and fast" access control decisions. EX-1005 [0068], [0075-76].

3. Detailed Application to the Challenged Claims

a. Claim 6

i. 6. The method according to claim 5, further comprising the step of updating a minimum required entity access level for any base nodes that comprise children of any generation of said higher-level nodes that are aggregations comprising said first base node.

203. Gladstone discloses, or renders obvious alone or in combination with Salazar, *the method according to claim 5* as shown above in Ground 1. *See* Claim 5.

204. Gladstone, as modified by Salazar and in view of Hada, renders obvious this step. Gladstone, as modified by Salazar, discloses user properties "associated with [Gladstone's] resource" (*a minimum access level established for any base node*). EX-1004 2:13-20, 2:30-33; *see also* [3a]. Gladstone also discloses, or renders obvious alone or in combination with Salazar, higher-level nodes that are aggregations comprising said first base node, *e.g.*, the prohibited combination X-Y-Z is an aggregation of the individual Events X, Y, and Z. *See, generally, claim 3*. A POSA would have understood that this access control involves evaluating Salazar's user property (a *minimum required entity access level*) to access the aggregated, higher-level node, as well as for each of the individual Events X, Y and Z (base nodes). *See* [1a, 1c]. A POSA would have also understood that it was well known in the art to incorporate higher-access levels for aggregated information (EX-1016 017 ("An **aggregate** of data might require a **higher classification** than the individual items."); §V.F.1) as admitted by the '708 Patent (EX-1001 6:20-26 ("[A]ggregated data is often of a more sensitive nature since it provides greater context and can identify relationships between the various individual information types. Consequently aggregated information will inevitably

be of greater interest to unauthorized users and system administrators will naturally wish to impose higher level restrictions on its access.”)).

205. Hada discloses “propagating” (*updating*) the action (*e.g.*, read, update, etc.) and effect (grant, deny) of an access control rule—such as that applied by Gladstone using Salazar’s user properties “associated with the resource” when controlling access to Gladstone’s aggregated node X-Y-Z—down toward “descendant[s].” EX-1005 [0043]. A POSA would have found it obvious to incorporate Hada’s teaching into Gladstone, as modified by Salazar, such that the access control rules governing node X-Y-Z would be propagated down to individual Events X, Y, Z (*base nodes that comprise children of any generation of* X-Y-Z (*said higher-level node*). A POSA would have been motivated to incorporate this propagation of Hada into Gladstone, as modified by Salazar, in order to implement Hada’s “more efficient and fast” access control decisions for structured documents. EX-1005 [0068], [0075-76]. Further, a POSA would have found it obvious to perform this same function multiple times (for *any base nodes that comprise children of any generation of said higher-level nodes*).

CONCLUSION AND DECLARATION

My findings and opinions set forth in this Declaration are based on my research and analysis to date. I may continue my examinations in view of additional documents or other factual evidence over the course of this proceeding that may necessitate supplementing and/or refining my opinions. I reserve the right to add to, alter, or delete my opinions and my Declaration upon discovery of additional information. I reserve the right to make such changes as necessary.

In signing this Declaration, I understand that the Declaration will be filed as evidence in a contested case before the Patent Trial and Appeal Board of the United States Patent and Trademark Office. I acknowledge that I may be subject to cross-examination in this case and that cross-examination will take place within the United States. If cross-examination is required of me, I will appear for cross-examination within the United States during the time allotted for cross-examination.

I declare that all statements made herein of my knowledge are true, and that all statements made on information and belief are believed to be true, and 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.

[SIGNATURE ON FOLLOWING PAGE]

Date: 3/8/2023 By: _____

Paul C. Clark

APPENDIX A – MATERIALS RELIED UPON

Exhibit #	***Description***
1001	U.S. Patent 7,302,708 ("the '708 Patent")
1002	Prosecution history of the '708 Patent ("File History")
1004	U.S. Patent No. 7,290,266 to Gladstone et. al. ("Gladstone")
1005	U.S. Patent Application Publication No. 2005/0076030 to Hada et. al. ("Hada")
1006	(Reserved)
1007	U.S. Patent No. 6,453,418 to Ooki et al. ("Ooki")
1008	U.S. Patent No. 6,446,077 to Straube et al. ("Straube")
1009	U.S. Patent Application Publication No. 2004/0122792 to Salazar ("Salazar")
1010	U.S. Patent No. 6,334,121 to Primeux et al. ("Primeaux")
1011	Lane et al., *Sequence Matching and Learning in Anomaly Detection for Computer Security*, AAAI Technical Report WS-97-07 (1997)
1012	U.S. Patent Application Publication No. 2005/0038881 to Ben-Itzhak ("Ben-Itzhak")
1013	Department of Defense Standard, DoD 5200.28-STD, *Department of Defense Trusted Computer System Evaluation Criteria* (1983)
1014	U.S. Patent Application Publication No. 2002/0147924A1 to Flyntz ("Flyntz")
1015	(Reserved)
1016	Dorothy et al., *Cryptography and Data Security* (Purdue University 1982)

APPENDIX B – *CURRICULUM VITAE*

PAUL C. CLARK
4705 Broad Brook Drive, Bethesda, Maryland, 20814
703.628.9500
paul@securemethods.com

SKILLS SUMMARY

A senior executive with extensive IT design, development and deployment experience, Dr. Clark has twenty five years of direct technical and management experience in the computer and systems engineering environment. He is the President of Paul C. Clark LLC. His specialties include complex commercial development and deployment of scalable secure network and data processing systems. He has served as keynote speaker, expert witness for high profile commercial clients and before Congress. He has served on federal advisory committees and as an adjunct professor at The George Washington University.

EMPLOYMENT

President and CTO July 1999 – Present
SecureMethods Inc. / Paul C. Clark LLC
Bethesda, MD

- Serves as Managing Director
- Manages operations and sales staff.
- Manages commercial product development staff.
- Provides product design, development and deployment guidance.
- Provides sales engineering support and collected customer feedback.
- Defines and communicates strategic technical vision.
- Successfully directed the development and deployment of commercial products on multiple Windows and Unix Platforms

Chief Scientist Jan. 1995 – July 1999
DynCorp Network Solutions
Fairfax, VA

- Managed technical staff and deliverables for multiple large projects.
- Provided project design, development and troubleshooting guidance.
- Provided customer technical interface and problem resolution.
- Designed and deployed next generation architecture for high volume network database and storage systems.
- Created a suite of secure products marketed and sold to DoD and the Federal Government.
- Provided corporate-wide technical consultation and support.

Senior Security Engineer Sept. 1990 – Jan. 1995
Trusted Information Systems
Glenwood, Maryland

- Participated in the design and implementation of the reference implementation of Privacy Enhanced Mail (PEM) with public and secret key encryption to provide security services for electronic transmissions.
- Designed NIST's Smartcard API (SCAPI). Implemented the SCAPI for the NIST 250 and utilized it to perform cryptographic operations for PEM.
- Implemented X.500 Certification and Distinguished Naming support for PEM. Functions supported included: CA and user registration, revocation, and high speed database for certificate storage and retrieval.
- Systems design and programming, including the TCB, for the Trusted Xenix and Trusted Mach MLS operating systems. Tasks included MAC labeling and audit strategies, as well as application development.
- Designed and implemented multilevel electronic mail and HTTP proxy for Trusted Mach. Task included cryptographic support for the TCB as well as application specific validation and semantic checking for up/downgrade.
- Inventor of the Boot Integrity Token System (BITS), which provides hardware, enforced authentication within the boot sequence and guarantees operating system integrity using smart tokens. U.S. Patent Nos. 5,448,045; 5,892,902; 8,695,066 and 10,129,214

Technical Lead Nov 1989 – Sept 1990
GTE Government Systems
Rockville, MD

- Managed the SAFE91 testbed effort with responsibilities including: budgets, schedules, customer negotiations, briefings, training, and tasking of technical staff. The testbed was created to simulate the client environment for the purpose of evaluating software packages and platforms for use by the client.
- Developed network load simulations for OS/2 LAN Manager to determine performance characteristics during periods of heavy traffic. This task included the design and development of a multi-threaded connection server under OS/2 version 1.0
- Designed and implemented an X Windows interface for the Minstrel System. This included the individual development of 20,000 lines of code in less than two months.
- Developed and taught DEC Windows and X Windows classes for GTE technical personnel. Responsible for instruction and problem solving throughout the subsequent development cycle.

Systems Engineer May 1985 – Nov. 1989
Ultrasystems Defense and Space

- Redesigned the Morse Mission Trainer while coordinating and tasking a team of programmers. During this time, received the President's quarterly award for outstanding performance. Task included systems and network and kernel level programming on SCO Xenix.
- Designed and implemented a database file server, benchmarked at over 100 retrievals per second on a million entry database. System implemented B* trees in C and ran on a Sun 3/260 workstation.
- Designed and implemented a satellite mission scheduler for multiple vehicles with multiple resources. Task included defining areas of interest (AOIs), flight paths, and optimal time allocation of resources.
- Designed and implemented the Ultraplot graphics tool to produce line and scatter plots, as well as bar graphs and histograms from large datafiles. This utility was implemented utilizing the DI3000 (device independent) graphics package.

HARDWARE Mainframe, Workstation, PC, including: IBM, DEC, Sun, HP, SGI, Intel

SOFTWARE UNIX (Linux, System V, BSD, Solaris, AIX, IRIX, Xenix), MS DOS/Windows, VMS, OS/2, X Windows, DEC Windows, Presentation Manger, TCP/IP, X25, Xenix Net, IBM LAN, DEC Net, Ethernet, Token Ring

EDUCATION

DSc. in Computer Science
Concentration in Security, Graphics, Intellectual Property Law
The George Washington University, 1994

M.S. in Electrical Engineering and Computer Science
University of Southern California, 1988

B.S. in Mathematics
University of California Irvine, 1986

Graduate level study in all major areas of computer science

REPRESENTATIVE PUBLICATIONS

"BITS – A Smartcard Protected Operating System," with Lance Hoffman, Communications of the ACM, November 1994.

"Service Layering Promotes Secure Data Exchange in Diverse Environments," Computer News, October 23, 1995

"Threats Posed to Cryptographic Applications by Random Numbers," presented to the RSA Data Security Conference, January 1996.

"A Reference Model for Electronic Commerce," with Daniel J. Blum and John Jauregui, Messaging Magazine, December 1996, Volume 2, Number 7.

"Secure Compartmented Data Access over an Untrusted Network Using a COTS-based Architecture," with Marion C. Meissner, and Karen O. Vance, Presented to the Annual Computer Security Applications Conference (ACSAC'00), New Orleans, December, 2000. Later published in "Statistical Methods in Computer Security," Marcel Dekker, ISBN 0-8247-5939-7, edited by William W. S. Chen, 2005

ADDITIONAL INFORMATION

(1) Dr. Clark was a member of the Federal Advisory Committee for Key Management Infrastructure (KMI); he was Chairman of the Interoperability Working Group for Cryptographic Key Recovery.

(2) Dr. Clark served as a Cooperative Research and Development Agreements (CRADA) partner, which is a joint effort between the National Institute of Standards and Technology (NIST) and several companies formed to begin development of the elements of a Public Key Infrastructure (PKI). A core element of this effort is the development of a Minimum Interoperability Specification for PKI components MISPC.

(3) Dr. Clark serves as an adjunct professor in the Electrical Engineering and Computer Science Department at The George Washington University. He teaches doctoral level cryptography and computer security courses.

(4) Speaker at The Federal Information Security Conference; presented "Embedded Security Deployment," Colorado Springs, CO, March 31, 2006

(5) Keynote Speaker for the Washington DC Bar Association; presented "Security for the Networked Computing Environment," August 8, 2005

(6) Appeared before Congressional committee to provide expert testimony; presented "Advanced Technology for Border Control," July 23, 1998.

(7) Keynote speaker at Mass Storage Conference; presented "Secure Data Access Over Public Networks," New Orleans, LA, October, 1996

(8) Keynote speaker at health care convention, presented "Security for Health Care Records," Nashville, TN, May, 1997.

(9) Keynote speaker at the USDA Plant and Genome Conference; presented "A Secure Architecture for Data and Systems," Nimes, France, October, 1997

(10) Speaker at IEEE Technical Meeting; presented "A Comprehensive Security Architecture," Virginia, June, 1996.

LEGAL CASES

Testifying expert for Google et al – ContentGuard v Google – Verdict for Google

Testifying expert for TransPerfect v MotionPoint – Verdict for Transperfect

Testifying expert for Cisco – VirNetX v Cisco – Verdict for Cisco

Testifying expert for CME – RealTime v CME – Summary judgment for CME

Testifying expert for IBM et al – Tecsec v IBM – Summary judgment for IBM

Testifying expert for Netflix et al – Parallel Networks v Netflix – Case dismissed

Testifying expert for Oracle - EpicRealm v Oracle – Summary judgment for Oracle

Testifying expert for Lucent - Microsoft v Lucent –Verdict for Lucent

Testifying expert for Oracle - Mangosoft v Oracle – Summary judgment for Oracle

Testifying expert for RSA Data Security – Digital Privacy v RSA – Summary judgment for RSA at the Markman