UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
90/011,491	02/15/2011	6,125,447	13557.112021	8208

25226 7590 06/29/2011

MORRISON & FOERSTER LLP
755 PAGE MILL RD
PALO ALTO, CA 94304-1018

EXAMINER	
ART UNIT	PAPER NUMBER

DATE MAILED: 06/29/2011

Please find below and/or attached an Office communication concerning this application or proceeding.

UNITED STATES PATENT AND TRADEMARK OFFICE

Commissioner for Patents
United States Patent and Trademark Office
P.O. Box 1450
Alexandria, VA 22313-1450
www.uspto.gov

DO NOT USE IN PALM PRINTER

(THIRD PARTY REQUESTER'S CORRESPONDENCE ADDRESS)

KING & SPALDING LLP

1180 PEACHTREE STREET, NE

ATLANTA, GA 30309-3521

MAILED

JUN 29 2011

CENTRAL REEXAMINATION UNIT

EX PARTE REEXAMINATION COMMUNICATION TRANSMITTAL FORM

REEXAMINATION CONTROL NO. *90/011,491*.

PATENT NO. *6,125,447*.

ART UNIT *3992*.

Enclosed is a copy of the latest communication from the United States Patent and Trademark Office in the above identified *ex parte* reexamination proceeding (37 CFR 1.550(f)).

Where this copy is supplied after the reply by requester, 37 CFR 1.535, or the time for filing a reply has passed, no submission on behalf of the *ex parte* reexamination requester will be acknowledged or considered (37 CFR 1.550(g)).

Office Action in Ex Parte Reexamination	Control No. 90/011,491	Patent Under Reexamination 6,125,447	
	Examiner MARY STEELMAN	Art Unit 3992	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

a☒ Responsive to the communication(s) filed on *02/15/2011, 04/28/2011* . b☐ This action is made FINAL.
c☒ A statement under 37 CFR 1.530 has not been received from the patent owner.

A shortened statutory period for response to this action is set to expire 2 month(s) from the mailing date of this letter. Failure to respond within the period for response will result in termination of the proceeding and issuance of an *ex parte* reexamination certificate in accordance with this action. 37 CFR 1.550(d). **EXTENSIONS OF TIME ARE GOVERNED BY 37 CFR 1.550(c).** If the period for response specified above is less than thirty (30) days, a response within the statutory minimum of thirty (30) days will be considered timely.

Part I THE FOLLOWING ATTACHMENT(S) ARE PART OF THIS ACTION:

1. ☒ Notice of References Cited by Examiner, PTO-892.
2. ☒ Information Disclosure Statement, PTO/SB/08.
3. ☐ Interview Summary, PTO-474.
4. ☐ _____.

Part II SUMMARY OF ACTION

1a. ☒ Claims *1-24* are subject to reexamination.
1b. ☐ Claims _____ are not subject to reexamination.
2. ☐ Claims _____ have been canceled in the present reexamination proceeding.
3. ☐ Claims _____ are patentable and/or confirmed.
4. ☒ Claims *1-24* are rejected.
5. ☐ Claims _____ are objected to.
6. ☐ The drawings, filed on _____ are acceptable.
7. ☐ The proposed drawing correction, filed on _____ has been (7a)☐ approved (7b)☐ disapproved.
8. ☐ Acknowledgment is made of the priority claim under 35 U.S.C. § 119(a)-(d) or (f).
   a)☐ All b)☐ Some* c)☐ None of the certified copies have
   1☐ been received.
   2☐ not been received.
   3☐ been filed in Application No. _____ .
   4☐ been filed in reexamination Control No. _____.
   5☐ been received by the International Bureau in PCT application No. _____.
   * See the attached detailed Office action for a list of the certified copies not received.
9. ☐ Since the proceeding appears to be in condition for issuance of an *ex parte* reexamination certificate except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte* Quayle, 1935 C.D. 11, 453 O.G. 213.
10. ☐ Other: _____

cc: Requester (if third party requester)

DETAILED ACTION

Reexamination

Claims 1-24 of USPN 6,125,447 to Gong (file date 12/11/1997, Application Control No. 08/988,439; issue date 09/26/2000) have been requested for reexamination. The Reexamination control number is 90/011,491.

Information Disclosure Statement

IDS received 04/28/2011 has been entered into prosecution. Citations lacking dates are lined through. The Examiner notes that the court proceedings have been considered. However, the citations do not meet the requirements of 37 CFR 1.98 and have been lined through. Where patents, publications, and other such items of information are submitted by a party (patent owner or requester) in compliance with the requirements of the rules, the requisite degree of consideration to be given to such information will be normally limited by the degree to which the party filing the information citation has explained the content and relevance of the information. The initials of the examiner placed adjacent to the citations on the form PTO /SB /08A and 08B or its equivalent, without an indication to the contrary in the record, do not signify that the information has been considered by the examiner any further than to the extent noted above. See MPEP 2256.

Notice Regarding Certain Reexamination Issues

The patent owner is reminded of the continuing responsibility under 37 CFR 1.565(a), to apprise the Office of any litigation activity, or other prior or concurrent proceeding, involving this patent under reexamination throughout the course of this reexamination proceeding. The third party requester is also reminded of the ability to similarly apprise the Office of any such activity or proceeding throughout the course of this reexamination proceeding. See MPEP §§ 2207, 2282 and 2286.

Extensions of time under 37 CFR 1.136(a) will not be permitted in these proceedings because the provisions of 37 CFR 1.136 apply only to "a Patent Applicant" and not to parties in a reexamination proceeding. Additionally, 35 U.S.C. 305 requires that reexamination proceedings "will be conducted with special dispatch" (37 CFR 1.550(a)). Extension of time in ex parte reexamination proceedings are provided for in 37 CFR 1.550(c).

Litigation involving USPN 6,125,447 to Gong

Oracle America, Inc. v. Google Inc., Civil Action No.: 3:10-cv-03561.

Overview of USPN 6,125,447 to Gong.

The claims of the '447 patent are directed to an alternative computer security system, which involves using "protection domains" to organize, represent, and maintain security policies that apply to a computer system. Gong '447 at Col. 2: 52-56. Each protection domain is associated with zero or more permissions. *See id.* "Each 'permission' is an

authorization by the computer system that allows a principal (executing processes, objects, and threads) to perform a particular action or function. *Id.* at Col. 2: 27-28.

Independent claims 1, 10, and 19 are directed to methods, computer systems, or computer-readable mediums that establish an association between one or more protection domains and one or more 'classes of one or more objects.' "An association is established between the protection domains and classes of objects that may be invoked by the computer system." *Id.* at Col. 2: 57-62. When an object requests an action, a determination whether the action is permitted is based on the association between the protection domains and classes. *See id.* at Col. 2: 64-65. Permissions needed for a requested action by object OA (where object OA is an instance of class CA) will be based on the permissions (permissions associated with class CA) in protection domain PA. *id.*

Independent claims 7 and 16 are more broadly directed to a method and a computer-readable medium, respectively, that establish an association between one or more protection domains and one or more 'sources of code' (not limited to object / class embodiments).

The reasons for allowance recite, "[T]he novelty of the claims, when read as a whole, are the steps and means for establishing an association between one or more protection domains and one or more classes of one or more objects (or sources of code) and

determining whether an action requested by an object is permitted based on this association." (See Notice of Allowance dated May 24, 2000, at page 2 (Exhibit 7).)

Relevant language citations found in Gong '447:

A protection domain establishes the permissions that apply to the code. For code that belongs to object classes, an association is established between the protection domains and the classes of objects. (Abstract) The security mechanism makes use of structures referred to herein as 'protection domains' to organize, represent and maintain the security policies... (2: 53-56) A protection domain is associated with zero or more permissions. (2: 59-60) The protection domains embody sets of permissions and are constructed based on policy information. (6: 36-38) A protection domain can be viewed as a set of permissions granted to one or more principals. (8: 41-42) A protection domain in this exemplary policy is defined as the set of permissions granted to the objects associated with a particular code identifier. (9: 9-11) A protection domain object is created, using the permissions container object to populate the protection domain. (10: 37-39) Mapping of classes to protection domains is stored as static fields in the protection domain class. (11: 16-18) Protection domain class static fields store data indicating which protection domains have been created and their associated code identifiers or alternately associations between a class and protections domains associated with the class stored as static fields; static methods access static data, are invoked on behalf of entire class. (11: 21-29) Each protection domain object is associated with permission objects. The association between the objects, permission domain objects and the permission

objects is based on the domain mapper, policy object, a policy file, and constitutes the security policy with respect to the objects. (11: 59-63 - exemplary use of objects and data structures)

An association is established between the protection domains and classes of objects (i.e., instantiations of the classes) that may be invoked... (2: 60-62)

A principal is an entity in the computer system to which permissions are granted. Examples of principals include processes, objects and threads. (2: 26-28)

The code identifier may contain data describing the source of code that defines a class, a set of public cryptographic keys associated with source of code, or other information which describes the source of code, or any combination thereof." (3: 11-15) Instructions stored in a file, a database system, or attributes of a persistent objects can be used to map code identifiers to authorized permissions. (9: 13-25) The policy object contains a mapping of all code identifier/authorized permissions mapped to the code identifier. (10: 31-35) A method of the policy object returns the permissions associated with a code identifier, invoked by passing the code identifier as a parameter. (10: 40-43) The policy object returns a permissions container object containing all the permissions associated with the code identifier. (10: 43-45) An association between protection domains and code identifiers is typically recorded in data persistently stored. The data associates code identifiers with one or more permissions. (3: 21-24)

A 'source of code is an entity from which computer instructions are received. Examples of sources of code include a file or persistent object stored on a data server connected over a network, a FLASH_EPROM reader...or a set of system libraries. (3: 15-20) The code source may be a composite record containing a URL and a set of public cryptographic keys. (7: 29-37)

An association between protection domains and the sources of code is also based on public cryptographic keys associated with the sources of code. (3: 43-44) A public cryptographic key (key) is used to validate the digital signature which may be included in a file used to transport related code and data. (7: 39-41) The key name is associated with a key. The key and corresponding key name are stored together in a database. The key name can be used to find the Key in the database. (9: 30-33)

An object is a record of data combined with the procedures and functions that manipulate the record. An object is an instance of the class to which the object belongs. (7: 1-8) Each class, defined by a class definition from code stream, is associated with a class name and a code source. (7: 25-26)

Executing object program code by way of a domain mapper 248 maps a class to the protection domain (to each instruction in the policy file) using a mapping data structure. (9: 40-52; 10: 59-60)

A permission is an authorization by the computer system that allows a principal to execute a particular action or function. (8: 43-45) A data structure containing text instructions can represent permissions. (8: 51-52) Permissions can also be represented by objects, referred to as permission objects. The object can contain an action attribute, a resource attribute, and permission validation methods. (8: 64-9:5)

Resource manager object manages access to resources related to request received from object on call stack. Resource manager object invokes access controller. Access controller determines whether permission required is authorized (permissions in protection domain associated with requesting object + each permission object of each object represented in call stack) for requesting object. (12: 12-23)

Prior Art References

USPN 5,412,717 to Fischer ("**Fischer**" or '**717**) (file date 05/15/1992, issue date 05/02/1995; currently an abandoned patent), qualifies as a 35 U.S.C § 102(b) reference. USPN 5,311,591 to Fischer, a continuation of '717, was previously cited on an IDS, but never applied in a rejection of Application Control No. 08/988,439.

"The Gateway Security Model in the Java Electronic Commerce Framework" by Theodore Goldstein ("**Goldstein**") 11/29/1996, qualifies as a 35 U.S.C § 102(b) reference.

"Java APIs: Playing Monopoly with Java via the JECF", Rawn Shah ("**Shah**") 12/01/1996, qualifies as a 35 U.S.C § 102(b) reference.

Fischer discloses a method for providing computer system security, including associating protection domains with object-oriented program structures, such as classes of objects and sources of code. Fischer discloses a set of authorities (permissions) and/or restrictions stored as "program authorization information" or "PAI." (protection domains) See Fischer, at Col. 2: 16-36. The PAI is assigned to a program to be executed (associate protection domains and classes), "to thereby delineate the types of resources and functions that the program is allowed to utilize." *See id.* PAIs associated to called and invoked programs may be combined, as appropriate. *See id.* at Col. 19: 40-54. Note object oriented data structures at Fig. 3C and Col. 2: 6-9 & 7: 49 - Col. 8: 2. The PAI can be associated with a signer of a digital certificate *(see* Fischer, at Col. 6: 25-35 and Fig. 2) or a manufacturer of a program (indicates the source of code used) *(see* Fischer, at Col. 9: 3-8 and Col. 16: 12-25). Fischer checks the PAI for authorization (determining whether said action is authorized); the PAI of Fischer may include a "hierarchy of nested certifications and signatures" (permissions associated with a plurality of routines in a calling hierarchy). USPN 4,868,877 and USPN 5,005,200, which are incorporated by reference, disclose additional teachings related to digital certificates and signatures.

"The Gateway Security Model in the Java Electronic Commerce Framework" by Theodore Goldstein ("**Goldstein**") 11/29/1996, qualifies as a 35 U.S.C § 102(b) reference. Goldstein has not been previously considered.

"Java APIs: Playing Monopoly with Java via the JECF", Rawn Shah ("**Shah**") 12/01/1996, qualifies as a 35 U.S.C § 102(b) reference. Shah has not been previously considered.

The disclosures of **Goldstein in view of Shah** are not cumulative to information cited or considered during prosecution of the '447 patent.

Claim Rejections - 35 USC § 102

The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

> A person shall be entitled to a patent unless –
>
> (b) the invention was patented or described in a printed publication in this or a foreign country or in public use or on sale in this country, more than one year prior to the date of application for patent in the United States.

Rejections

Claims 1-24 are rejected under 35 U.S.C. 102(b) as being anticipated by USPN 5,412,717 to Fischer. See Request 02/15/2011, pages 17-18 and Exhibit 8 Claim Chart.

Per **claims 1, 7, 10, 16, and 19**, Fischer '717 discloses a method, computer readable medium executed by one or more processors, and system for providing security.

The Fischer disclosure references 'authorization entries' within the "program authorization information," or "PAI," to verify access authority to other code and resources. "... providing enhanced computer system security while processing computer programs... " Fischer at 1:20-25. See exemplary computer readable medium carrying a "sequence of instructions" as shown in Fischer '717, FIGs. 1 (#2, #7), 10 & 11. Fischer '717 teaches (Abstract; 4: 24-61) a system.

Fischer discloses **establishing one or more protection domains, wherein a protection domain is associated with zero or more permissions**;

Fischer's PAI ('717, 2: 34-36) reads on the claimed 'protection domain.' (2: 20-23), "The system monitor builds a data structure (establishing one or more protection domains) including a set of authorities (protection domains as PAIs defining permissions) defining that which a program is permitted to do (permissions) and/or that which the program is precluded from doing." (9: 17 – 10: 23), "The program control block 140 is loaded with program authorization information such that the PAI can be readily referenced as the associated program is executed so as to ensure that the program performs functions and accesses resources in conformance with its assigned authorizations. The program control block associated with the program to be executed is located in a storage area which cannot be modified by the program."

Fischer discloses **establishing an association between said one or more protection domains and one or more classes of one or more objects.**

(2: 6-9), "The present invention is directed to providing reliable security, even when operating with complex data structures, e.g., objects, containing their own program instructions, which are transmitted among users." (2: 26-33), "Once defined, the program authorization information [(PAI)] is thereafter associated with each program to be executed to thereby delineate the types of resources and functions that the program is allowed to utilize. The PAI associated with a particular program may be assigned by a computer system owner/user or by someone who the computer system owner/user implicitly trusts." See Fischer, FIG. 3C, #116, noting an object oriented program. An object is an instance of a class, i.e., a class template. Gong '447 recites (6: 63-7: 29) well known facts related to objects: "[e]ach object belonging to a class has the same fields ('attributes') (protection domain attributes) and the same methods." (7: 14-18, 7: 49-8: 2), "... The program authorization information is embedded in a segment 116 which specifies the authorization for the object's program or programs in a manner to be described more fully hereinafter." (15: 24-26), "Thereafter, the PAI is stored using, for example, one of the approaches set forth in FIGS. 3A through 3D so that it is associated with its program 272 ...."

Fischer discloses **determining whether an action requested by a particular object is permitted based on said association between said one or more protection domains and said one or more classes.**

(15: 56-59), "FIGS. 10 and 11 illustrate the sequence of operations of a supervisor program for controlling the processing of a program being executed in accordance with its program authorization information." (16: 66-17: 3), "Depending on the processing in block 316 [of FIG. 10], a decision is made in block 322 whether the signatures are valid, authorized and trusted. If the signatures are not determined to be valid, then the routing branches to block 324 where the execution in program X is suppressed." In the case where digital signatures are used to determine whether an action requested is permitted, Fischer discloses (17: 31-33), "If the processing in blocks 322 and 316 reveal that the signatures are valid, then the processing in block 326 is performed." As noted above, the protection domain objects are derived from (association) a protection domain class definition.

Claim 19 also recites the term "domain mapping object" for establishing an association between said one or more protection domains and one or more classes of one or more objects. Fischer discloses (2: 20-23), "The system monitor (domain mapping object) builds a data structure (establishing an association between one or more protection domains) including a set of authorities (protection domains as PAIs defining permissions) defining that which a program is permitted to do (permissions) and/or that which the program is precluded from doing." (FIG. 3C; 7: 14-18, 7: 49-8: 44), The data structure

defines (establishes associations) the type of object...embeds a segment with program authorization information, a segment with object method code, and a segment with data variables.

Per **claims 2, 11, and 20**, Fischer discloses **at least one protection domain of said one or more protection domains is associated with a code identifier.**

Fischer's PAI data structure (i.e., protection domain data structure) explicitly associates the protection domain with a "source of code" such as the signer of a digital certificate. (6:25-35 & FIG. 2), "The authorization signature includes a signature segment 40. The signature segment 40 may include a reference to the signer's certificate, i.e., an identifier for identifying the signer's certificate. In accordance with a preferred embodiment of the present invention, such a digital certificate is a digital message created by a trusted entity which contains the user's public key and the name of the user (which is accurate to the entity's satisfaction) and possibly a representation of the authority which has been granted to the user by the party who signs the digital message." See also Fischer 9: 3-8, associating the PAI with the manufacturer of the program; 11: 7-13, manufacturer may define a range of authorities associated with the program; 16: 12-25, program associated with signed 'pedigree' (public key or digital certificate) from manufacturer.

Fischer discloses **at least one class of said one or more classes is associated with said code identifier.** See Fischer, Figures 2 and 3C. The PAI data structure may contain a manufacturer's signature (i.e., code identifier) and that the PAI with the code identifier is

associated with the object/class data structure because it is expressly included as part of the object/class data structure. *Fischer,* Fig. 3C ("PROGRAM(S') SIGNED AUTHORIZATION (PAI)" included as element 116 of the disclosed object/class data structure).

Fischer discloses **associating said one or more protection domains and said one or more classes based on said code identifier.**

Authorization (e.g., protection domains) may be associated with a program based on the digital signature (i.e., code identifier) included in an object/class: (16: 15-20), "Thus, if a well known manufacturer of programs has signed the program with a public key or digital certificate, then, if desired, such a program may be assigned whatever level of authority desired depending upon how much the manufacturer is trusted and the system may permit execution of such program."

Per **claims 3, 12, and 21**, Fischer discloses **the code identifier indicates a source of code used to define each class of said one or more classes.**

(7: 51-56), "FIG. 3C shows an illustrative data structure for a secure exchangeable 'object'. The data structure may be signed by a trusted authority. The signing of such a data structure allows the object to be securely transmitted from user to user."

Per **claims 4, 13, and 22**, Fischer discloses **the code identifier indicates a key associated with each class of said one or more classes.** As an example, Figure 2 in Fischer discloses the digital certificate (i.e., code identifier) includes a public key that may be associated with the object/class. (6: 25-35), "...such a digital certificate is a digital message created by a trusted entity which contains the user's public key and the name of the user (which is accurate to the entity's satisfaction) and possibly a representation of the authority which has been granted to the user by the party who signs the digital message." Note discussion above related to an object program, where objects are instances of a class definition.

Per **claims 5, 14, and 23**, Fischer discloses **the code identifier indicates a source of code used to define each class of said one or more classes.** The digital signature (i.e., code identifier) may be associated with a manufacturer of the program (i.e., a source of code or "an entity from which computer instructions are received"): (9:3-8), "The present invention allows PAI information to be associated in any appropriate manner, so that in principle a user could define one or more levels of PAI which are then combined together with perhaps a more universal PAI, or with a PAI which was signed and supplied by the or [sic] manufacturer of this program." (11: 7-13), "FIGS. 6 through 9... a flowchart illustrating an exemplary sequence of operations of a utility program for establishing program authorization information. Such a utility program prompts a user, i.e., the end user, the user's agent, or even the manufacturer, to define a range of authorities which are associated with a program to be executed by the user's system." (16: 12-25), "If no PAI

has yet been associated with the program, then a check is made to determine whether the program has an associated signed 'pedigree' from the manufacturer (306). Thus, if a well known manufacturer of programs has signed the program with a public key or digital certificate, then, if desired, such a program may be assigned whatever level of authority desired depending upon how much the manufacturer is trusted and the system may permit execution of such program. Such a digital signature from the manufacturer can be used to verify...."

Fischer discloses **the code identifier indicates a key associated with each class of said one or more classes.** For example Figure 2 in Fischer discloses the digital certificate (i.e., code identifier) includes a public key may be associated with the object/class:

(6: 25-35 & FIG. 2), "The authorization signature includes a signature segment 40. The signature segment 40 may include a reference to the signer's certificate, i.e., an identifier for identifying the signer's certificate. In accordance with a preferred embodiment of the present invention, such a digital certificate is a digital message created by a trusted entity which contains the user's public key and the name of the user (which is accurate to the entity's satisfaction) and possibly a representation of the authority which has been granted to the user by the party who signs the digital message." See FIG. 3C. Note that Fischer teaches object code (defined classes provide a template from which an object is derived / an instance of the object).

Per **claims 6, 15, 20, and 24**, Fischer discloses **associating said one or more protection domains and said one or more classes based on data persistently stored, wherein said data associates code identifiers with a set of one or more permissions.** (6: 25-28), "The authorization signature includes a signature segment...may include a reference to the signer's certificate, i.e., an identifier for identifying the signer's certificate... (associate code identifiers with a set of one or more permissions)" (7: 49-8: 2), "FIG. 3C shows...PAI data structure (protection domains) is associated with a program... shows an illustrative data structure for a secure exchangeable 'object' (an object instance of a defined class). The data structure may be signed (code identifier) by a trusted authority. The signing of such a data structure allows the object to be securely transmitted from user to user. Although the data structure shown in FIG. 3 is set forth in a general format, it may be structured as set forth in the inventor's copending application filed on Apr. 6, 1992 and entitled 'Method and Apparatus for Creating, Supporting and Processing a Travelling Program' (U.S. Set. No. 07/863,552.), which application is hereby expressly incorporated herein by reference." "... The program authorization information (protection domain) is embedded in a segment 116 which specifies the authorization for the object's program or programs (where objects are instances of defined classes) in a manner to be described more fully hereinafter." Alternately, Fischer discloses (7: 20-35 & FIG. 3A, #102) storing PAI information (protection domain) on a separate/remote storage device or in the same memory as the program: "...Although the program authorization information, PAI 1, is depicted as being stored in a separate memory device 100 (data persistently stored), it may, if desired, be stored in the same memory media (data persistently stored) as its associated program."

The limitations of **claim 8** are a combination of limitations from claims 1-5. Mapping to Fischer '717 is noted above.

See limitations of **claim 9** addressed in the similar limitations of claims 1- 6 above.

"...said one or more sources of code..." reads on Fischer's teachings of object code (classes) provided by a trusted manufacturer. The associated 'keys' are a narrower limitation than the 'code identifier' of claim 6. The digital signature keys read on the term 'code identifier' or 'keys.' See Fischer's teachings at FIG. 3A, 3C, 7: 20-8:2.

Limitations of **claim 17** are variations of claims 1 and 2 addressed above. In claim 17, 'sources of code' is a broader term similar to 'one or more classes of one or more objects.' 'One or more keys' is a narrower variation of 'code identifier.' Fischer fairly discloses a trusted manufacturer source of code (object code) (16: 13-15), and associated public key or digital signature (16: 16-17) identifying the code. (16: 50-65), For a PAI that is signed and associated with a particular program, the signatures are verified (valid and trusted?) through a certificate hierarchy. (association between protection domain, one or more sources of code, and one or more keys associated...) See Fischer FIG. 2 & 6: 25-35.

See limitations of **claim 18** disclosed by Fischer in the similar limitations of claims 1-6 above.

Claims 1-24 are rejected under 35 U.S.C. 102(b) as being anticipated by Goldstein with supporting evidence by Shah. See Request 02/15/2011, page 18 and Exhibit 9 Claim Chart, which are incorporated by reference.

Goldstein describes a "Gateway" extension to the Java security model, for use in the Java Electronic Commerce Framework ("JECF"). *See* Goldstein, at 1. According to Goldstein, an application called a "cassette" is associated with a set of permissions (protection domain) represented by "Roles (Role objects contain permissions/authorizations)." See id. at 7-8, 10, 13-14, and Figs. 1 and 5-6. Goldstein discloses (p. 13) a Java gateway security model, which includes protection domains represented by Roles. Shah clarifies that these Roles are objects that represent and/or contain specific authorizations for a set of code (cassette), as well as a digital signature (based on a public/private key pair) corresponding to the creator of the code (cassette).

"[R]oles dictate the available resources and security levels and control [the] program's interface to the JECF code. A local (persistent) database contains these access control lists and role information. Each... cassette (program, represented in object format, object, class, package, etc.) with its specific roles (Role objects/ protection domain store permissions) must be signed by a trusted authority before use to guarantee the identity of

the originator."). The Roles may be defined on a per-package basis so that any class that is part of the package *(e.g.,* a cassette package) is necessarily associated with the protection domain.

See Goldstein, at 10 ("All classes in a package have access to package-private data members and methods .... Packages are a natural choice for creating a security principal."). In addition, Goldstein discloses establishing an association between the protection domain of a user's cassette with a class corresponding to a protected resource such as a JECF database. See id. at 11-12.

Shah provides supporting evidence for Goldstein's teachings of the JECF (Java Electronic Commerce Framework). See M.P.E.P. § 2131.01(III) (An extra reference or evidence can be used in an anticipation rejection to show an inherent characteristic of the concept taught by the primary reference).

Shah further clarifies that Roles contain information regarding the permissions (or authorization) of the cassette (invoking program) and that Roles are digitally signed (code identifier) by the originator of the cassette application: Shah (p. 2), "This JECF-based service can appear in the form of an...applet...(or it could be a specific Java application on your machine." "The JECF implementation architecture from Sun consists of the following components...Payment Cassettes...Service Cassettes..." Shah (p. 3), "When the applet is loaded, the Class Loader object is set to execute in a limited

environment " and calling programs are checked for their digital signature for uniqueness. When a JECF object is invoked, the invoking program or application is checked for its role. These roles dictate the available resources and security levels and control that program's interface to the JECF code. A local database contains these access control lists and role information. Each Payment and Service cassette with its specific roles must be signed by a trusted authority before use to guarantee the identity of the originator." Goldstein (p. 13) discloses a Role as essentially a public and private key pair.

It should be noted that Gong '447 defines the term 'principal' at 2: 27-30 to include "processes, objects and threads." A permission is an authorization by the computer system that allows a principal (an executing process, object or thread) to perform a particular action or function." Goldstein (p. 10) uses the term 'principal' in a different manner: "In this paper, a right is an abstract privilege. A principal uses a right. A principal can be a person, a corporation, a program, or a body of code." Goldstein at p. 11 recites, "Electronic commerce also needs the ability to delegate rights from one principal to another..."

In summary, claims 1-24 are rejected.

Conclusion

Amendment Proposed in Reexamination – 37 CFR 1.530(d) Patent owner is notified that any proposed amendment to the specification and/or claims in this

reexamination proceeding must comply with 37 CFR 1.530(d)-(j), must be formally presented pursuant to 37 CFR 1.52(a) and (b), and must contain any fees required by 37 CFR 1.20(c).

In order to ensure full consideration of any amendments, affidavits or declarations, or other documents as evidence of patentability, such documents must be submitted in response to this Office action. Submissions after the next Office action, which is intended to be a final action, will be governed by the requirements of 37 CFR 1.116, after final rejection and 37 CFR 41.33 after appeal, which will be strictly enforced.

Any paper filed with the USPTO, i.e., any submission made, by either the Patent Owner or the Third Party Requester must be served on every other party in the reexamination proceeding, including any other third party requester that is part of the proceeding due to merger of the reexamination proceedings. As proof of service, the party submitting the paper to the Office must attach a Certificate of Service to the paper, which sets forth the name and address of the party served and the method of service. Papers filed without the required Certificate of Service may be denied consideration. See 37 CFR 1.550(f)

Please mail any communications to:

Attn: Mail Stop "Ex Parte Reexam"
Central Reexamination Unit
Commissioner for Patents
P. O. Box 1450
Alexandria VA 22313-1450

Please FAX any communications to: (571) 273-9900

Central Reexamination Unit

Please hand-deliver any communications to: Customer Service Window

Attn: Central Reexamination Unit

Randolph Building, Lobby Level

401 Dulany Street

Alexandria, VA 22314

Registered users of EFS-Web may alternatively submit such correspondence via the electronic filing system EFS-Web, at https://sportal.uspto.gov/authenticate/authenticateuserlocalepf.html. EFS-Web offers the benefit of quick submission to the particular area of the Office that needs to act on the correspondence. Also, EFS-Web submissions are "soft scanned" (i.e., electronically uploaded) directly into the official file for the reexamination proceeding, which offers parties the opportunity to review the content of their submissions after the "soft scanning" process is complete.

Any inquiry concerning this communication or earlier communications from the examiner, or as to the status of this proceeding, should be directed to the Central Reexamination Unit at telephone number (571) 272-7705.

/Mary Steelman/

Primary Examiner

Central Reexam Unit 3992

Conferees:

EBK