5-28-02

A

PATENTS

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re Application of: Ron Redlich, et al.

Serial No.:

Filed: Herewith

For: Data Security System and Method With Parsing and Dispersion Techniques

TRANSMITTAL AND CERTIFICATE OF EXPRESS MAILING

Box PATENT APPLICATION
Commissioner for Patents
Washington, D.C. 20231

Sir:

I HEREBY CERTIFY that the below listed documents are being deposited with the United States Postal Service "Express Mail Post Office to Address," Mailing Label No. **EL 777613587 US**, under 37 C.F.R. 1.10 on May 23, 2002 and are addressed to Box Patent Application, Commissioner for Patents, Washington, D.C. 20231.

X Patent application (93 pgs.; 27 clms.)

X Seventeen (17) sheets drawings - informal

X Declaration and Power of Attorney for Patent Application (2 pgs.)

X Assignment to Digital Doors, Inc. w/assignment cover sheet (5 pages)

X IDS and Form PTO-1449 (5 pgs.) with Twenty three (23) references

X Credit Card Payment Form in the amount of $515.00

X Fee calculation letter (1 pg.) (duplicate)

X Return receipt postcard.

X Please charge any underpayments or credit any overpayments to Deposit Account No. 03-1231. A duplicate copy of this transmittal is enclosed.

Respectfully submitted,

By _____
Ellen Kotler

By _____ William Trueba
Robert C. Kain, Jr. Reg. No. 48002
Reg. No. 30,648
FLEIT, KAIN, GIBBONS, GUTMAN & BONGINI, P.L.
750 S.E. Third Avenue, Suite 100
Ft. Lauderdale, Florida 33316-1153
(954) 768-9002

\\Tiger\data share\RCK\CLIENTS\Redlich\6851-2 CIP-3 EXPMAIL PAT wpd

IPR2023-00969
International Business Machines Corporation v. Digital Doors, Inc.
IBM's Exhibit 1002
Ex. 1002, Page 1

FLEIT, KAIN, GIBBONS, GUTMAN & BONGINI P.L.
ATTORNEYS AT LAW

Miami • Fort Lauderdale • Boca Raton

750 Southeast Third Avenue
Suite 100
Fort Lauderdale, Florida 33316-1153
Telephone: (954) 768-9002
Toll Free: (800) 846-0900
Facsimile: (954) 768-0158
www.FleitKain.com
Robert C. Kain, Jr. rkain@fleitkain.com
William R. Trueba, Jr. wtrueba@fleitkain.com

May 23, 2002

Commissioner for Patents
Washington, D.C. 20231

Transmitted herewith for filing are papers related to the patent application of:

Inventor(s): Ron Redlich, et al.
For: Data Security System and Method With Parsing and Dispersion Techniques
Our Ref: 6851-2-CIP-3

The filing fee for the enclosed patent application has been calculated as shown below:

	No. Filed	No. Extra	Small Entity Rate	Small Entity Fee	Other than Small Entity Rate	Other than Small Entity Fee
Basic Fee				$370		$740
Total Claims	27 - 20 =	7	x 9	=$ 63	x 18	=$
Ind. Claims	4 - 3 =	1	x 42	=$ 42	x 84	=$
____Multiple claims presented			x140	=$	x 280	=$
			Total	$475	Total	$

A Credit Card Payment Form in the amount of $515.00 is enclosed to cover the filing fee (plus applicable Assignment recording fee). Please charge any additional filing fees required under 37 CFR 1.16 to Deposit Account No. 03-1231. A duplicate copy of this letter is enclosed.

Small Entity under 37 C.F.R. §1.27. YES __X__ NO _____

Respectfully submitted,

Fleit, Kain, Gibbons, Gutman & Bongini, P.L.

By _____ WILLIAM TRUEBA
FOR Robert C. Kain, Jr. REG. NO. 48,002
Reg. No. 30,648

\\Tiger\data share\RCK\CLIENTS\Redlich\6851-2cip-3 fee calc ltr pto credit card wpa.wpd

DATA SECURITY SYSTEM AND METHOD WITH PARSING AND DISPERSION TECHNIQUES

[001] This is a continuation-in-part of patent application no. 10/008,209 and 10/008,218, both filed on December 6, 2001, which are continuations-in-part of patent application no. 09/916,397 filed July 27, 2001 which is a regular patent application is based upon provisional patent application no. 60/260,398, filed June 5, 2001; application no. 60/287,813, filed on May, 2, 2001; application no. 60/267,944, filed February 12, 2001; application no. 60/247,242, filed November 13, 2000 and application no. 60/247,232, filed November 13, 2000.

[002] The present invention relates to a data security system and method and, more specifically, parsing (typically through a filter), extracting and dispersion of the parsed or segregated data with the controlled reconstruction of data utilizing security protocols to provide a system, method and programming instructions stored on computer readable medium for securing data. Scrubbing credit card data or financial data from text, a data object or data stream is also discussed herein.

Background of the Invention

[003] The extensive use of computers and the continued expansion of telecommunications networks, particularly the Internet, enable businesses, governments and individuals to create documents (whether text, images, data streams or a combination thereof, sometimes identified as "data objects") and distribute those documents widely to others. Although the production, distribution and publication of documents is generally beneficial to society, there is a need to limit the distribution and publication of security sensitive words, characters or icons. Concerns regarding the privacy of certain data (for example, an individual's social security number, credit history,

Ex. 1002, Page 3

medical history, business trade secrets and financial data) is an important issue in society. In another words, individuals and businesses have a greater concern regarding maintaining the secrecy of certain information in view of the increasing ease of distribution of documents through computer networks and the Internet.

[004] U.S. Patent No. 6,055,544 to DeRose et al. discloses the generation of chunks of a long document for an electronic book system. DeRose '544 discloses solutions available to book publishers to publish books in electronic format on the worldwide web. One of the problems is that the books are published as small document fragments rather than publishing an entire book which, due to the formatting, protocol and command structure on the Internet, downloads an entire book to the user. The problem involved with publishing small documents is that there is no relationship to other portions of the book. See col. 3, lines 51-55 and col. 4, lines 3-5. One methodology to solve the problem involves inserting hypertext links in the book. This places a large burden on the book publisher. Col. 4, lines 19-21. Accordingly, it is an object of DeRose '544 to provide a mechanism for accessing only a portion of a large, electronically published document and automatically determining what portion of the document to download to the user based upon user selections that is, previous portions and subsequent portions of the document are downloaded with the selected portion, without maintaining separate data files for each portion of the document. Col. 4, lines 34-39. In other words, if a person wanted to access chapter 4 of a text, the system in DeRose '544 would display chapter 4, chapter 3 (the preceding chapter) and chapter 5 (the subsequent chapter). This publishing of portions of the document utilizes a subset of marked up elements established as being significant and a second subset of elements being less significant. For example, "Title elements" define a table of contents. A first representation of the document structure defined by all

Ex. 1002, Page 4

of the marked up elements may be used in combination with a second representation of the document structure defined only by the significant elements to control selection of portions of the documents such that previous and subsequent portions may be selected and rendered in a consistent and intuitive manner." Col. 4, lines 38-55. A computer system stores a first representation of the hierarchy of all elements in the electronic document. As example, this may be each chapter in its entirety. The computer also stores a second representation of the hierarchy of only significant elements in the electronic document. As an example, this may be a listing of each chapter without the text associated with the chapter. In response to request for a portion of the document, the computer system selects the portion defined by the significant element in the second representation. For example, if the user requested chapter 4, the entirety of chapter 4 would be downloaded from the web server to the client computer. In addition to rendering or publishing the selected chapter, the computer system looks to the relationship of the elements in the first representation of the hierarchy (the list of all chapters) and downloads from the web server the adjacent chapters. In this example, this would involve downloading chapters 3 and chapter 5. In a further embodiment, the computer system selects only a leaf element of the second representation as a significant element during the download. See the Summary of the Invention, col. 4, line 40 through col. 6, line 14.

[005] U.S. Patent No. 5,832,212 to Cragun et al. discloses a censoring browser method for viewing downloaded and downloading Internet documents. The abstract describes the system as including a user profile including user selected censoring parameters. Data packet contents are received from the Internet and the packets are compared with the user selected censoring parameters. Responsive to the comparison, the received data packet contents are processed and selectively displayed. The user selected censoring parameters include censored words and word fragments, and

user selected categories. Compared word and word fragments can be removed and selectively replaced with predefined characters or acceptable substitute words. Tallies of weights for user selected categories are accumulated and compared with used selected threshold values. A predefined message can be displayed responsive to an accumulated tally exceeding a user selected threshold value without displaying the received data packet contents.

[006] U.S. Patent No. 6,094,483 to Fridrich discloses an encryption methodology hiding data and messages in images. In one application of the system in Fridrich '483, a method is disclosed of embedding a secret digital square image with 256 gray levels within an image carrier. The secret image is first encrypted using a chaotic Baker map. The resulting image is a random collection of pixels with randomly distributed gray levels without any spatial correlations. The carrier image is twice the size (height and width or 2nX2m) the secret image with 256 gray levels. The carrier image is modified according to a mathematical formula.

[007] U.S. Patent No. 5,485,474 to Rabin discloses a scheme for information dispersal and reconstruction. Information to be transmitted or stored is represented as N elements of a field or a computational structure. These N characters of information are grouped into a set of n pieces, each containing m characters. col. 1, lines 37-46. The system is used for fault tolerance storage in a partitioned or distributed memory system. Information is disbursed into n pieces so that any m pieces suffice for reconstruction. The pieces are stored in different parts of the memory storage medium. A fairly complex mathematical algorithm is utilized to provide reconstruction of the information utilizing no fewer than m pieces.

[008] U.S. Patent No. 6,192,472 B1 to Garay et al. discloses a method and apparatus for the secure distributed storage and retrieval of information. Garay '472 identifies the problem as how

Ex. 1002, Page 6

to store information in view of random hardware or telecommunications failures. Col. 1, lines 17-20. The initial solution is to replicate the stored data in multiple locations. Col. 1, lines 28-31. Another solution is to disburse the information utilizing in Information Disbursal Algorithm (IDA). The basic approach taking in IDA is to distribute the information F being stored among n active processors in such a way that the retrieval of F is possible even in the presence of up to t failed (inactive) processors. Col. 1, lines 40-44. Another issue is the utilization of cryptographic tools. With the use of tools called distributed fingerprints (hashes), the stored data is distributed using the fingerprints and coding functions to determine errors. In this way, the correct processors are able to reconstruct the fingerprint using the code's decoding function, check whether the pieces of the file F were correctly returned, and finally reconstruct F from the correct pieces using the IDA algorithm. Col. 2, lines 50-59. Garay '472 also discloses the use of Secure Storage and Retrieval of Information (SSRI) with the added requirement of confidentiality of information. Col. 3, line 56. With this added requirement, any collision of up to t processors (except ones including the rightful owner of the information) should not be able to learn anything about the information. Confidentiality of information is easily achieved by encryption. Col. 3, lines 56-61. The issue involves encryption key management, that is, the safe deposit of cryptographic keys. Garay '472 discloses confidentiality protocol utilizing distributed key management features. This mechanism allows the user to keep his or her decryption key shared among several n servers in such a way that when the user wants to decrypt a given encrypted text, the user would have to interact with a single server (the gateway) to obtain the matching plaintext while none of the servers (including the gateway) gets any information about the plaintext. Col. 4, lines 5-14.

Ex. 1002, Page 7

[009] U.S. Patent No. 5,996,011 to Humes discloses a system and a method for filtering data received over the Internet by a client computer. The system restricts access to objectionable or target data received by a client computer over an Internet by a web server by filtering objectionable data from the data received. The Humes '011 system filters the data "on the fly." Further, the Humes '011 system can be applied to process any type of target data from the data received and displayed to the user. Col. 2, lines 32-44. If the web page requested by the user contains only a minimum amount of objectionable or target data, the user receives only a portion of the filtered web page for viewing. Hume '011 also provides that if the web page contains a large amount of objectionable material, the system blocks the entire display of the web page on the user's computer monitor. Col. 2, lines 56-62. Hume '011 provides three levels of filtering. At the first level, if the domain name contains objectionable words or material, the initial download from the domain is blocked. At the second level, the text in the download is filtered and objectionable words are replaced with a predetermined icon, for example, "----". Col. 3, lines 32-35. The filter uses a dictionary. Col. 3, lines 45-48. The filtered out words are counted. If the final score of "filtered out" material exceeds a predetermined threshold, the entire page is blocked from the user's view. Col. 4, lines 2-4.

[010] U.S. Patent No. 5,905,980 to Masuichi, et al., discloses a document processing apparatus for processing various types of documents, a word extracting apparatus for extracting a word from a text item including plural words, a word extracting method used in the document processing apparatus, and a storage medium for storing a word extracting program. Extracted words are associated with other words via an algorithm. The extracted words and associated words are used as a search index for the document.

Ex. 1002, Page 8

[011] U.S. Patent No. 5,996,011 to Humes discloses a computer based system and method for filtering data received by a computer system, and in particular, for filtering text data from World Wide Web pages received by a computer connected to the Internet, for purposes of restricting access to objectionable web sites.

[012] U.S. Patent No. 6,148,342 to Ho discloses a system for managing sensitive data. The system prevents a system administrator from accessing sensitive data by storing data and identifier information on different computer systems. Each query from a user's terminal is encrypted using two codes, the first code readable only by an identifier database and a second code readable only by a data access database. The data is routed from the user's source terminal to the identifier database at the first computer. The first computer/identifier database first verifies the user's ID and the security clearance for the requested information and substitutes a second internal ID to the data packet/query. The modified query is then presented to the data access database (the second computer) and, subject to a second security clearance, the response to the data query is sent back to the user's source terminal.

[013] A publication entitled "Element-Wise XML Encryption" by H. Maruyama T. Imamura, published by IBM Research, Tokyo Research Laboratory, April 20, 2000 discloses a protocol or process wherein certain parts of an XML document are encrypted and the balance of the plaintext is not encrypted. The protocol is useful in three party transactions, for example, when a buyer sends an order in an XML document to a merchant which contains the buyer's credit card information. The credit card information is sent to a credit company and the merchant does not need to know the credit number as long as he obtains clearance or authorization from the credit card company. Another instance is an access control policy which requires a certain part of an XML

Ex. 1002, Page 9

document to be readable only by a privileged user (for example, a manager could access the salary field in an employee records but others could only access name, phone and office fields). The Imamura article discusses encryption protocol, the delivery of keys and the utilization of compression. The article does not discuss separate storage of the critical data apart from the plaintext of the XML document.

[014] The Ingrain i100 Content Security Appliance product brochure, available in June, 2001, discloses a system coupled to multiple web servers (computers) to accelerate secured transactions between multiple client computers (over the Internet) and prevents Secure Sockets Layer SSL performance bottlenecks by performing high-performance SSL handshakes and encrypting all data sent to back end servers using long-lived SSL session.

[015] An article entitled "Survivable Information Storage Systems" by J. Wylie M. Bigrigg, J. Strunk, G. Ganger, H. Kiliccote, and P. Khosla, published August, 2000 in COMPUTER, pp. 61-67, discloses a PASIS architecture which combines decentralized storage system technologies, data redundancy and encoding and dynamic self-maintenance to create survivable information storage. The Bigrigg article states that to achieve survivability, storage systems must be decentralized and must spread information among independent storage nodes. The decentralized storage systems partition information among nodes using data distribution and redundancy schemes commonly associated with disc array system such as RAID (redundancy array of independent discs) insuring scalable performance for tolerance. P. 61. Thresholding schemes - also known as secret sharing schemes or information disbursal protocols - offer an alternative to these approaches which provide both information confidentiality and availability. These schemes and codes, replicate, and divide information to multiple pieces or shares that can be stored at different storage nodes. The system

can only reconstruct the information when enough shares are available. P. 62. The PASIS architecture combines decentralized storage systems, data redundancy and encoding and dynamic self-maintenance to achieve survivable information storage. The PASIS system uses threshold schemes to spread information across a decentralized collection of storage nodes. Client-side agents communicate with the collection of storage node to read and write information, hiding decentralization from the client system. P. 62. The device maintains unscrubable audit logs --that is, they cannot be erased by client-side intruders -- security personal can use the logs to partially identify the propagation of intruder-tainted information around the system. P. 63. The article states that, as with any distributed storage system, PASIS requires a mechanism that translates object names -- for example file names -- to storage locations. A directory service maps the names of information objects stored in a PASIS system to the names of the shares that comprised the information object. A share's name has two parts: the name of the storage node on which the share is located and the local name of the share on the storage node. A PASIS file system can embed the information needed for this translation in directory entries. P.63. To service a read request, the PASIS call client (a) looks up in the directory service the names of the n shares that comprise the object; (b) sends read requests to at least m of the n storage nodes; (c) collects the responses and continues to collect the responses until the client has collected m distinct shares; and (d) performs the appropriate threshold operation on the received shares to reconstruct the original information. P. 63. The p-m-n general threshold scheme breaks information into n shares so that (a) every shareholder has one of the n shares; (b) any m of the shareholders can reconstruct the information; and (c) a group of fewer than p shareholders gains no information. P. 64. Secret-sharing schemes are m-m-n threshold schemes that trade off information confidentiality and information availability:

the higher the confidentiality guaranty, the more shares are required to reconstruct the original information object. Secret sharing schemes can be thought of as a combination of splitting and replication techniques. P. 64. The article discusses the technique of decimation which divides information objects into n pieces and stores each piece separately. Decimation decreases information availability because all shares must be available. It offers no information theoretic confidentiality because each share expresses 1/n of the original information. P. 64. Short secret sharing encrypts the original information with a random key, stores the encryption key using secret sharing, and stores the encrypted information using information disbursal. P. 64. An extension to the threshold schemes is cheater detection. In a threshold scheme that provides cheater detection, shares are constructed in such a fashion that a client reconstructing the original information object can tell, with high probability, whether any shares have been modified. This technique allows strong information integrity guarantees. Cheater detection can also be implemented using cryptographic techniques such as adding digest to information before storing it. P. 65. For the highest architecture to be effective as possible, it must make the full flexibility of threshold schemes available to clients. The article believes this option requires automated selection of appropriate threshold schemes on a per object basis. This selection would combine object characteristics and observations about the current system environment. For example, a client would use short secret sharing protocol to store an object larger than a particular size and conventional secret sharing protocol to store smaller objects. The size that determines which threshold scheme to use could be a function of object type, current system performance, or both. P. 67.

[016] The MAIL sweeper and MIME sweeper programs by ReSoft International uses a keyword search engine to review e-mails for certain words or phrases. IF the e-mail does not clear the filter,

the addressee data must clear a data base check to protect the privacy and/or confidentiality of the e-mail data. See re-soft.com/product/mimesweep. The Aladdin eSafe Appliance restricts outgoing e-mails from sending classifier or prohibited content. See aks.com/news/2001/esafe.

Objects of the Invention

[017] It is an object of the present invention to provide a data security system, an information processing system and a method for securely storing data and rebuilding that data in the presence of an adequate security clearance.

[018] It is another object of the present invention to provide a method for securing data on a single personal computer (PC), on a plurality of computers linked together through a local area network (LAN) or a wide area network (WAN) or the Internet.

[019] It is a further object of the present invention to provide a method for securing data utilizing a client-server computer system. The client-server computer system may be implemented over the Internet. The security system may be provided to the public, to government or to private entities as an Application Service Provider or ASP over the Internet.

[020] It is a further object of the present invention to provide a method for securing data which is highly flexible and programmable by a user.

[021] It is an additional object of the present invention to enable the user to establish (a) the scope of the security sensitive words, characters or icon, data objects, (b) the future use (or destruction or encryption) of a filter enabling extraction of security sensitive data, (c) the selection of storage locations (local, removable, in an LAN, a WAN or on the Internet) for extracted data and remainder or common data and (d) one or multiple levels of security limiting full reconstruction and enabling partial reconstruction of the extracted data and the remainder or common data.

Ex. 1002, Page 13

[022] It is another object of the present invention to establish and manage the separation of user-based communities of interest based upon crypto-graphically separated, need to know security levels.

[023] It is another object to provide an adaptive system responsive to hacking attempts and hacking attacks.

[024] These steps may be completely automated (after some parameters are set or programmed by a system administrator), may be fully programmable by the user, or may be a combination of automated and manual controls implemented by the systems administrator and/or the user-client.

[025] It is an object of the present invention to parse, disperse and reconstruct the data or data object thereby enabling secure storage of the data. For example, financial data maintained by an institute, can be parsed with an algorithm, the parsed segments dispersed off-site and away from the financial institute, and, upon appropriate security clearance, the dispersed data can be reconstructed to duplicate the data. Large distribution of parsed data is contemplated by this aspect of the invention. The original data remains stable, operable and immediately useful. The securing dispersed data is a back-up of the original data.

[026] It is a further object of the present invention to secure e-mail data transmissions and web browser transmissions by extraction of security sensitive data, facilitating the remote storage of said data and sending remainder data to the e-mail addressee or the recipient.

[027] It is another object of the present invention to use fine-grained selection of security critical data, extraction and encryption and separate storage of the secured data. The parsing or filtering of plaintext, data object, file or data stream thereby bridges the gap between full encryption

of the plaintext etc. and no encryption. The present system is therefore a more efficient use of processing speeds, times, and storage resources.

[028] It is a further object to create a credit card number or financial data scrubber. The scrubber may be employed to remove any security critical data.

[029] It is another object of the present invention to permit the user to decide on and select a level of risk he or she believes appropriate by selecting no, minimal, intermediate or maximum levels of data security.

[030] It is another object of the present invention to permit the user to access data security risks, access data processing resources (processing time, storage facilities, data access time, etc.) and select a security level which balances risks and resources.

Summary of the Invention

[031] The method for securing data in a computer system in one embodiment includes establishing a group of security sensitive words, characters, icons, data streams or data objects, filtering the data input from a data input device and extracting the security sensitive data. The extracted data is separated from the remainder data and is separately stored. In one embodiment on a personal computer (PC) system, the extracted data and the remainder or common data is stored in different, distributed memory segments. In a network implementation, the extracted data may be stored in one computer and the remainder or common data may be stored in another computer. In a client-server implementation, the server may direct storage of the extracted data to a different location than the remainder data, either on the server or on a further memory system (computer) interconnected to the server or on the client computer and in distributed memory segments. A map may be generated by a software module or sub-system indicating the location of the extracted data

Ex. 1002, Page 15

and the remainder data in the network. The filter may be destroyed (via a deletion routine) or may be retained for future use by the user. If retained, encryption is preferred. The map may be stored on the client computer or the user's PC or may be stored on the server. Copies of the map may be removed (deleted) from the user's PC or the client computer. The map may be encrypted. The extracted data and/or the remainder data may be removed (deleted or scrubbed) from the originating computer. Encryption can be utilized to further enhance the security levels of the system. All transfers of the filter between the client to the server may be encrypted, and all data (whether extracted data or remainder data) may be encrypted prior to storage in the distributed memory. Any transfer of extracted data or remainder data or maps or filters may include an encryption feature. Reconstruction of the data is permitted only in the presence of a predetermined security clearance. A plurality of security clearances might be required which would enable a corresponding plurality of reconstructing users to view all or portions of the data. Persons with low level security clearance would only be permitted to have access to low level extracted data (low level security sensitive data) and the common data. Persons with high level security clearances would be permitted access to the entire document reconstituted from the extracted data and the remainder data. A computer readable medium containing programming instructions carrying out the methodology for securing data is also described herein. An information processing system for securing data is also described.

[032] In another embodiment, the method for securing data in a computer network and transparently establishing and managing the separation of user-based communities of interest based upon crypto-graphically separated, need to know, security levels, by necessity, utilizes communities of interest representing a plurality of users having corresponding similar security levels, each with a respective security clearance. In other words, all members of Community A have the same security

Ex. 1002, Page 16

level and security clearance, which is different than the users of Community B which have a different security level and security clearance. The method and the computer media containing programming instructions includes filtering data from the data input computer, extracting security sensitive words, phrases, characters, icons, or data objects and forming subsets of extracted data and remainder data. The subsets of extracted data are stored in one or more computer memories in the network identified as extracted stores. The remainder data is also stored in the network if necessary. Reconstruction of some or all of the data via one or more of the subsets of extracted data and the remainder data is permitted only in the presence of a predetermined security clearance from the plurality of security levels. The cryptographically separated, need to know, security levels correspond to respective ones of the plurality of security levels and the method includes, in one embodiment, encrypting subsets of extracted data with corresponding degrees of encryption associated with the plurality of security levels. During reconstruction, all or a portion of the plaintext data is decrypted only in the presence of the respective security level. The information processing system which secures data per the community of interest security level in the includes a data filter for the data input from the data input computer which extracts the security sensitive words, phrases, icons or data objects. A system and a methodology for storing the subsets of extracted data and remainder data is provided and a compiler permits reconstruction of some or all of the plain text data in the presence of an appropriate security clearance level. Multiple level encryption in one document is also available.

[033] An adaptive method of securing data responsive to a plurality of hacking events utilizes a hacking monitor which generates a corresponding plurality of hack warnings dependent upon the severity of the hacking attack. Based upon respective ones of the hacking or hack

Ex. 1002, Page 17

warnings, data is filtered to extract security sensitive words, phrases etc. and the extracted data and the remainder data (if necessary) is stored based on the degree of hack warning. Reconstruction is permitted of some or all the data utilizing the extracted data and the remainder data only in the presence of the predetermined security clearance level. Automatic reconstruction is permitted after the hack attack terminates. The method sometimes includes encrypting extracted data dependent upon the degree or severity of the hack warning and decrypting that data during reconstruction. A computer readable medium containing programming instructions similar to the method is also provided. The information processing system includes a filter which is adjusted based upon the degree of hack warning to extract security sensitive words. A storage system stores extracted data and remainder data (if necessary) based upon the level of the hack warning and a compiler is used to reconstruct the data in the presence of the appropriate security clearance level.

[034] The parsing and dispersion aspects of the present invention enable the user to parse, disperse and reconstruct the data or data object thereby enabling secure storage of the data. The original data may be maintained in its original state and stored as is customary, encrypted or destroyed. For example, financial data maintained by an institute in its original state, and a copy thereof can be parsed with an algorithm, the parsed segments dispersed off-site, (that is, separated and stored in extract and remainder stores or computer memories), away from the financial institute, and, upon appropriate security clearance, the dispersed data can be reconstructed to duplicate the data. Large distribution of parsed data is contemplated by this aspect of the invention. The original data remains stable, operable and immediately useful in its stored location. The secured and dispersed data is a back-up of the original data. Destruction of the original source is also an alternative embodiment.

Ex. 1002, Page 18

[035] Another embodiment of the present invention operates in an e-mail or a web browser environment. In a specific embodiment, the invention operates as a credit card or financial data scrubber. The e-mail data has one or more security sensitive words, characters or icons and the method or computer program works in a distributed computer system with a remote memory designated as an extract store. The method extracts the security sensitive words, characters or icons from said e-mail data to obtain extracted data and remainder data therefrom. The extracted data is stored in the extract store. The methodology emails the remainder data to the addressee. the addresses is permitted to retrieve the extracted data from said extract store only in the presence of a predetermined security clearance and hence, reconstruct the e-mail data with said extracted data and remainder data. The program and method on the user's e-mail device extracts the security sensitive data, facilitates storage of the extracted data in said extract store and, emails the remainder data to the addressee. Rather than extracting security data, the method and program may parse the data. The method and program for safeguarding data entered via a browser involves extracting security sensitive data, facilitating the storage of such data in the remote store, and forwarding the remainder data to a targeted destination in the distributed computer system. The scrubber may utilize a pop-up window to enable user activation of the scrubber on an email or a web browser communication.

[036] The present invention can be configured in various forms. The following descriptions discuss various aspects of the invention and further advantages thereof.

[037] The present invention enables automatic classification and declassification of documents on the fly. The extraction process downgrades and declassifies documents on the fly (in

Ex. 1002, Page 19

real time) so that they are useless to unauthorized parties. Presentation by a user of a valid security clearance enables substantially instant and seamless reconstitution of the security sensitive content.

[038] The present invention automatically secures unstructured documents and freeform documents for example, e-mail, instant messaging, or Word documents (input documents).

[039] The present invention automatically secures structured documents and transactional documents for example, database records or XML documents (input documents).

[040] The present invention introduces flexibility into security management, risk management of data, data storage, and data flows and enable automatic responsiveness to threats. The innovation enables automatic response to security challenges and threats. The innovation can maintain, upgrade and downgrade the levels of security through implementation of a leveled granular extraction process and a controlled-release mechanism. Attacks or other external events can trigger a response in the form of higher extraction levels, expanding the type of content extracted, and constricting the release of important and critical data control from storage. How much and what to extract depends on the level of threat or perceived risk. In same manner, the amount and type of content released from storage and reconstituted depends on the level of threat or risk perceived by the system. The system delivers a level of security protection specifically matched to meet security needs as dictated by the changing security threats, environment, policy and organizational needs.

[041] The present invention introduces multiple levels and standards of security. It is common knowledge that the highest security is delivered through total separation. Whereas this concept has only been implemented physically or by isolating computer environments, the invention achieves this concept of total separation within open and networked computer environments. The invention can implement a total physical and logical separation of important and critical data from

Ex. 1002, Page 20

its context and can preclude access to that information without a needed granular access permission. The invention is also effective for sounds and images (data objects or data streams with security words, characters, terms, icons or other data objects).

[042] The present invention introduces a new method and apparatus to monitor security sensitive content through a process of analysis and categorization of each word or character, in a document. The invention enables processing of every character, word, number, as they are entered into a document and categorizes each into one of many pre- set categories. Categories can include surnames, locations, currency, defined terminology, and unknown words or phrases.

[043] The present invention introduces a method and apparatus for plain text extraction and dispersion of security sensitive data. Maximum security with traditional methods encumbers free flow of information and business efficiency. Encryption burdens computer systems with high performance overhead, and its use is limited to the parties who have decryption capabilities. The invention offers a new solution. It enables leveled security in plain-text format, in addition to none, some, or all of pre-existing encryption, decryption, firewalls, and other security infrastructure. The level of security is determined by the extent of the security sensitive items, selection process; the extent of dispersal to various distributed storage locations; the rules for controlled-release from storage; and the access rules governing the reconstitution of extracts into the secured document.

[044] The extractions are dispersed to distributed storage on a granular level. The rest of the document can be stored at its original location and/or other storage locations. Dispersal of extractions introduces new barriers not existing in current security. In certain situations, an attacker has first to find the (encrypted) map to the locations, then locate and access the distributed storage,

get the data released from the controlled-release storage, and finally reintegrate the extracts into the appropriate documents.

[045] The present invention introduces a method and apparatus for targeted extraction and encryption of security sensitive items. The extraction capabilities of the system enable different workflow modes. The system enables extraction and encryption of important and critical content. In essence, only the critical content is extracted and/or encrypted, whereas the rest of the document remains as plaintext. This capability enables the following: advantages and flexibility; and the ability to share the document within the organization or transmit it to outsiders while still maintaining security over the most important and critical content of the document. This is an automatic process for controlling the content of outgoing e-mail. The document owner releases the important and critical content by enabling access to it to defined parties at defined times within defined threat modes.

[046] The present invention introduces a method and apparatus for encrypting document or extractions with multiple encryption types. The invention can deliver the highest level of security by using multiple types of encryption (and/or multiple keys) for one line, paragraph or document. Maximum security is delivered through automatic selection of security sensitive items, and encrypting these extractions with one or more types of encryption. The remainder data can also be encrypted. Multiple encryption types within one document statistically precludes deciphering that document regardless of the available computer power. Common encryption methods are vulnerable through existing technologies, social engineering methods, carelessness, and workflow habits. Furthermore, simple encryption becomes more vulnerable (including triple DES) assuming future

Ex. 1002, Page 22

mathematical breakthroughs or quantum computing. Existing methods to crack block ciphers are being improved to compromise the future AES Rinjdael standard.

[047] The present invention to introduce a method and apparatus for content dispersion. The innovation enables control over specific important and critical content items within the general contents of documents or digital files in a computer or within a network. The immediate controlled-release of those important content items according to specific identification and access criteria proactively safeguards the security and the value of documents or digital files. The content control enables broad dissemination of the digital files in closed networks, as well as open networks including the Internet, without compromising the security of the important and critical information in the digital file. The dispersal channels can include any of all of the following: networks, Internet, Virtual Private Channel. Telephone lines, Optical lines, Wireless, Fax, Documents, Verbal communication.

[048] The present invention introduces a method and apparatus for enhancing the survivability capabilities of an organization and its networks. If networks get damaged, the decryption capability, such as PKI, is likely to be compromised, or at a minimum, suspended. In such instances, the invention enables continuation of work on channels, which need not be secure. In addition, the dispersion of information guarantees maximum partial reconstitution to documents and transactions, or total reconstitution to documents and transactions benefiting from backup at distributed storage facilities.

[049] The present invention introduces a method and apparatus for delivering security for inter-connecting networks. It enables security for closed networks connecting to the Internet and other open networks. The Internet infrastructure and open networks are not secure. Even secured

Ex. 1002, Page 23

closed networks, such as VPNs, are not secured enough. The critical content of documents is the critical asset of the organization and must be highly secured, with maximum reliability, full transparency and instant accessibility. To remain competitive, organizations must maximize utility of the critical data within their documents, files, databases and servers. The securing of such documents must not be at the expense of compromising the access or processing speed of such documents. The invention enables work in plain text, as well as with encryption. Working in plain text reduces the computing performance overload.

[050] The present invention introduces a method and apparatus for delivering information flow control in decentralized environments. Protection of privacy and confidentiality of information represents a long-standing challenge, The challenge has become much bigger with the expansion of the Internet, which has created decentralized networks. Parties, who do not know or trust each other, have to exchange information. The invention enables free flow and sharing of information between parties by removing burdening security restrictions and creating top security with a controlled-release of the security sensitive content in the documents. The technology enables top security through introduction of user and organization's ownership and control of the critical granular data in documents.

[051] The system introduces an additional layer of access controls at the granular level of the user document. In order to view the reconstructed critical information the user would need to be verified by additional access controls at the data storage level. The user access code or a physical key enables release of data from the storage. Today's access controls do not stop the user from distributing documents to other parties. The inventions fined grained controlled-release mechanism releases the critical information, only under a required set of circumstances and access validation.

Ex. 1002, Page 24

The invention enables the user ownership of his security sensitive critical data and conditions for its release and dissemination. The user has the option to hide the critical data through declassification process and release through a reclassification process in which the critical data would be reconstituted in the document.

[052] The present invention introduces a method and apparatus for delivering Compartmentalization of security sensitive content by leveled access to users. The invention creates leveled sharing of information, for example such that persons with level 3 access will have keys for encryption type RSA persons with level access 2 will have access to Blowfish encryption within one document.

[053] The present invention introduces a method and apparatus for enabling more use of distributed and dispersed storage including ASPs (application service providers). There is a major human tendency to refrain from sending important documents to web storage locations because of potential security breaches. This cultural issue is both driven by psychological issues and well-founded security concerns. The retention of those documents as is in physical proximity or locked security, provides actual security but precludes gaining any utility from those documents in a functional business setting. Instead the invention enables functional distribution of those documents without the security sensitive data, and a controlled-release of some or all of the extractions in a granular way in order to support business activities while retaining security.

[054] The present invention introduces a method and apparatus for enabling lower storage costs. The extraction process declassifies and downgrades mission critical documents. The downgrading and transformation of a critical document into a non-critical document, enables storage in less secured and lower cost storage. Taking advantage of this security-initiated, extraction process

can yield substantial storage cost savings. The invention enables a high return on investment ROI for system storage cost arbitrage. Splitting the data into critical and non-critical enables 20 to 90% savings on storage cost.

[055] The present invention delivers an automated security risk management system that creates added in-depth security defenses at the semantic-level as well as creation of controlled-release mechanisms at the storage-level with significantly reduced performance overhead requirements.

[056] The present invention presents a technology which answers the security demands as required by Committee on Information Systems Trustworthiness of the National Research Council. The Committee's report, Trust in Cyberspace (1999), defines the security paradigms needed for a safe future. The report states: The substantial commercial off-the-shelf (COTS) makeup of a network information systems, the use of extensible components, the expectation of growth by accretion, and the likely absence of centralized control, trust, or authority demand a new approach to security: risk mitigation rather than risk avoidance; technologies to hinder attacks, rather than prevent them outright; add-on technologies and defense in depth; relocation of vulnerabilities rather than their elimination; none of the existing or security technologies addresses these needs in whole. The invention breakthroughs this barrier by providing a single system which implements each one of those four elements in a unified way. The invention controls information flow in centralized and decentralized environments, through controlled-release of information within distributed systems.

[057] The present invention implements security measures while accommodating the performance needs of a network. The invention provides a method and apparatus to ease overhead performance on congested computer networks. It can adjust the security defenses based on the

performance needs of the network. Many security systems overburden the already burdened computing environment in terms of computational overhead, labor, and training requirements. The invention enables to ease the overhead performance of a network by transforming from high overhead performance, encryption methods, and other security methods, to the method presented by this invention.

[058] The present invention minimizes the time of exposure of the important content within a document. The invention enables to separate the important content from the rest of the document for substantial periods of time, thereby minimizing substantially the exposure to risk. It is possible for example to extract the important content from the document and release it for reconstitution only when the user will open the document. In such situations the important content could for example be time and unexposed for over 99% of the time and exposed for less than 1% of the time, which lowers the risk substantially.

[059] The present invention provides a security risk management method and system to minimize security risks. The invention enables minimization of security risks by: Automatic separation and extraction of granular critical data from the core document. Dispersal of the extracted critical data groups to different secured storage locations. Reconstitution of the critical data in document for limited time, to minimize exposure to risk. Partial reconstitution, of the critical data, in core document, through a controlled release of granular critical data. Granular controlled release of data to specific authorized people only.

[060] The present invention provides a controlled release security mechanism to enable the release of content and granular content from storage locations in a centralized and decentralized

Ex. 1002, Page 27

environment. The controlled release mechanism enables release of the appropriate content to the authorized party at the right time under the right circumstances.

[061] The present invention provides a security solution against damage by insiders. Studies show that insiders cause 70%-85% of the damage. These nine innovations are described in detail as follows: The invention enables insiders and employees to work with documents while managers and owners control the release of the critical prioritized information. The control is granular, thereby enabling continued work with the rest of the content in the document. The objective is to empower the user with the highest security while enabling him maximum sharing and delivery flexibility. This enables free flow of information between closed networks and public networks, such as the Internet, without compromising the security through extraction of important and critical content. The user can transport documents through various networks and e-mail services knowing that the critical information, which is still under control, and is not compromised.

[062] The present invention provides an automatic security system in order to overcome human flaws that create security vulnerabilities. Human engineering flaws are the cause of 90% of security vulnerabilities. For example, passwords are exposed through human fault enabling reading of plain text before it is encrypted. The invention enables an automatic process of appropriate response to security threats in an objective way and on an on going basis.

[063] The present invention provides an automatic security system in order to reduce human labor, and training costs.

[064] The present invention provides protection for important granular content within a document. A feature left out in computer development is the protection and automatic protection of granular important content in a document. In every facet of life critical assets are immediately

protected. For example, credit cards and cash are protected in a wallet, important items at home are placed in closets, wall units, cabinets and safes. The present system extracts the digital equivalent of these items, e.g., extracts all credit card data, and stores the extracted data in secure location(s).

[065] The present invention provides an alternative method to encryption. Mathematical security and encryption could be broken. Discovery of a mathematical equation for a shortcut of the factoring of prime numbers would be make mathematical security and encryption extremely vulnerable.

[066] In 1999 a 512-bit RSA key was broken- at that time 95% of keys in e-commerce were 512 bits long. U.S. government 56-bit Data Encryption Standard was cracked in just 22 hours by the Freedom Foundation.100, 000 PCs were connected with a supercomputer which enabled the testing of 245 billion keys per second.

[067] The invention provides an automated security risk management system. The system automatically responds to attacks by matching the defenses level to the level of threats The system responds to security threats through the following mechanisms: (1) controlled extraction of sensitive security data: in normal circumstances, extractions will take place according to pre-set rules; in threat situations, additional extractions will take place to deliver higher security; in an attack, additional substantial amounts of critical data will be extracted to deliver the highest security; (2) controlled dispersal to storage locations; in normal circumstances, dispersal to different storage locations according to pre-set rules will take place; in threat and attack situations, more dispersal to more storage locations, via additional communication channels will take place; and (3) controlled release of extracts for reconstitution; controlling amount of extracts released for reconstitution; controlling

Ex. 1002, Page 29

time of exposure of extracts in reconstitution; limiting access to specific people; and limiting access to specific times.

[068] The present invention defends against devices like keyboard sniffers and mouse sniffers that can read information keyed into the computer and transmit it to an adversary. The invention enables to input security sensitive items through data input devices other than the keyboard. For example credit card numbers can be inputted through a hand held wireless devise. The inputted data would be transferred to storage for possible reconstitution.

[069] The present invention defends against as devices that intercept electromagnetic signals from computers, monitors, printers, and keyboards. For example the Van Eck receptors which can read information off the screen the display screen. The invention enables separation contents of document into two or more displays thereby limiting the potential damage of electromagnetic eavesdropping.

[070] The present invention enables the controlled release of data objects, full or partial release of plaintext source documents to persons or organizations with the appropriate security clearances.

[071] Another object of the present invention is to enable the control of information flow over a PC, a network, a LAN, a WAN and over the Internet.

[072] A further object of the present invention is to enable the interoperability of several secured networks based upon the relative security clearances of each network.

[073] It is another object of the present invention to provide a process for synthesizing a document.

Brief Description of the Drawings

Ex. 1002, Page 30

[074] Further objects and advantages of the present invention can be found in the detailed description of the preferred embodiments when taken in conjunction with the accompanying drawings in which:

[075] FIG. 1A diagrammatically illustrates a basic system diagram showing filtering and storing extracted data and remainder or common data and, in an enhanced embodiment, generating and storing a map.

[076] FIG. 1B diagrammatically illustrates a system diagram showing reconstruction of the data, various security clearances and both electronic reconstruction and visual reconstruction.

[077] FIG. 2 diagrammatically illustrates a system showing major components of a single personal computer (PC) system, a networked system with several PCs (a LAN or WAN) and the network coupled to a telecommunications system and the Internet and shows the interconnection with a server and multiple, Internet-connected memory units.

[078] FIG. 3 diagrammatically illustrates a basic flowchart showing reconstruction for various security levels.

[079] FIG. 3A diagrammatically illustrates interleaving distinct data into different memory locations in a video memory.

[080] FIG. 4 diagrammatically illustrates a flowchart showing one embodiment of the principal portions of the data security program.

[081] FIG. 5 diagrammatically illustrates a flowchart showing the basic elements of the reconstruction process for the data security program.

[082] FIG. 6 is a computer network diagram showing various user communities.

[083] FIG. 7 diagrammatically illustrates a flowchart showing the key component steps for the multiple layer security program for the community of users.

[084] FIG. 8 diagrammatically illustrates a flowchart showing the key components of an adaptive security program adaptable to various levels of electronic attacks, hacker or hack attacks.

[085] FIG. 9 diagrammatically illustrates a flowchart showing the key components of a multiple encryption program using multiple types of encryption in one document or data object.

[086] FIG. 10 diagrammatically illustrates a chart showing the key components of the parsing, dispersion, multiple storage and reconstruction (under security clearance) of data.

[087] FIGS. 11A and 11B diagrammatically illustrate a flowchart showing the key components of one embodiment of the e-mail security system (jump points 11-A and 11-B link the flow charts).

[088] FIGS. 12A and 12B diagrammatically illustrate a flowchart showing the key components of one embodiment of the invention implements the security system on a web browser (jump point 12-A links the flow charts).

[089] FIG. 13 diagrammatically shows several revenue systems which may be employed with the data security systems described herein.

Detailed Description of the Preferred Embodiments

[090] The present invention relates to a data security system, a methodology of securing data on a personal computer (PC) system, on a computer network (LAN or WAN) and over the Internet and computer programs and computer modules and an information processing system to accomplish this security system.

Ex. 1002, Page 32

[091] It is important to know that the embodiments illustrated herein and described herein below are only examples of the many advantageous uses of the innovative teachings set forth herein. In general, statements made in the specification of the present application do not necessarily limit any of the various claimed inventions. Moreover, some statements may apply to some inventive features but not to others. In general, unless otherwise indicated, singular elements may be in the plural and vice versa with no loss of generality. In the drawings, like numerals refer to like parts or features throughout the several views.

[092] The present invention could be produced in hardware or software, or in a combination of hardware and software, and these implementations would be known to one of ordinary skill in the art. The system, or method, according to the inventive principles as disclosed in connection with the preferred embodiment, may be produced in a single computer system having separate elements or means for performing the individual functions or steps described or claimed or one or more elements or means combining the performance of any of the functions or steps disclosed or claimed, or may be arranged in a distributed computer system, interconnected by any suitable means as would be known by one of ordinary skill in the art.

[093] According to the inventive principles as disclosed in connection with the preferred embodiment, the invention and the inventive principles are not limited to any particular kind of computer system but may be used with any general purpose computer, as would be known to one of ordinary skill in the art, arranged to perform the functions described and the method steps described. The operations of such a computer, as described above, may be according to a computer program contained on a medium for use in the operation or control of the computer as would be known to one of ordinary skill in the art. The computer medium which may be used to hold or

Ex. 1002, Page 33

contain the computer program product, may be a fixture of the computer such as an embedded memory or may be on a transportable medium such as a disk, as would be known to one of ordinary skill in the art.

[094] The invention is not limited to any particular computer program or logic or language, or instruction but may be practiced with any such suitable program, logic or language, or instructions as would be known to one of ordinary skill in the art. Without limiting the principles of the disclosed invention any such computing system can include, inter alia, at least a computer readable medium allowing a computer to read data, instructions, messages or message packets, and other computer readable information from the computer readable medium. The computer readable medium may include non-volatile memory, such as ROM, flash memory, floppy disk, disk drive memory, CD-ROM, and other permanent storage. Additionally, a computer readable medium may include, for example, volatile storage such as RAM, buffers, cache memory, and network circuits.

[095] Furthermore, the computer readable medium may include computer readable information in a transitory state medium such as a network link and/or a network interface, including a wired network or a wireless network, that allow a computer to read such computer readable information.

[096] In the drawings, and sometimes in the specification, reference is made to certain abbreviations. The following Abbreviations Table provides a correspondence between the abbreviations and the item or feature.

Abbreviations Table

A-com	computer or memory store for common or remainder data
ASP	application service provider - server on a network

Ex. 1002, Page 34

B-ext	computer or memory store for extracted data
bd	board
CD-RW	compact disk drive with read/write feature for CD disk
comm.	communications, typically telecommunications
CPU	central processing unit
doc	document
dr	drive, e.g., computer hard drive
DS	data storage
e	encryption
ext-data	extracted data
I/O	input/output
I-com	Internet storage for common or remainder data
I-ext	Internet storage for extracted data
loc	location
mem	memory
obj	object, for example, a data object
pgm	program
re	regarding or relating to
recon	reconstruct
rel	release
req	request
rev	review

Ex. 1002, Page 35

sec	security
sys	system
t	time
tele-com	telecommunications system or network
URL	Uniform Resource Locator, x pointer, or other network locator

[097] FIG. 1A diagrammatically illustrates the basic processes for establishing a secure storage of information, generally identified herein as "data." "Data," as used herein, includes any data object, e.g., text, images, icons, moving images, multiple images, data representing sound, video etc. Sound bites and video images may also be extracted data. A source document 100, sometimes referred to as a "plaintext," is passed through a filter 102. Although it is convenient to discuss and understand the invention herein in connection with a plaintext document, the document 100 is a data object. It is not limited to an electronic document representing words. The document 100 represents a data object that may be e.g., text, images, icons, moving images, multiple images, data representing sound, video etc. The term "data object" as used in the claims is broadly defined as any items that can be represented in an electronic format such that the electronic format can be manipulated by a computer as described herein. The data object, or as discussed herein, the "plaintext" is sent to a filter. Filter 102, in a most basic sense, separates out common text or remainder data 104 from uncommon text, words, characters , icons or data objects. The security sensitive words, characters, icons or data objects are separated from remainder or common text 104 as extracted text 106. It should be noted that although the word "text" is utilized with respect to remainder text 104 and extracted text 106, the text is a data object and includes words, phrases, paragraphs, single characters, portions of words, characters, whole or partial images, , icons or data objects. In a basic

Page 1 of 52 - Description

implementation, filter 102 may utilize a dictionary such that words present in the dictionary (common words) are separated from the source plaintext document 100 and placed into remainder document or common data file 104. The uncommon words (extracted-security sensitive words), not found in the dictionary, would be placed in an extracted text or extracted data file 106. For example, a business may wish to impose a security system on a contract document such that the names of the contracting parties (not found in the dictionary) and the street names (not found in the dictionary) would be stored in extracted data text file 106. The common text or remainder data would be stored in remainder data file 104. In the illustrated embodiment, remainder data file 104 also includes place holders which enables the extracted data to be easily inserted or set back into the remainder data file.

[098] The security sensitive words, characters, icons or data objects may be any word, phrase, letter, character, icon, data object (full or partial), image or whatever, as pre-defined or as established by the user. The user may specifically design the filter, begin with a dictionary to define common terms, identify any additional security sensitive words, letters, images, icon, data objects, partial versions of the foregoing or any other granular aspect of the plaintext. After defining the filter and accepting the data input, the system filters the plaintext and separates extracted data (security sensitive items) from the remainder data. The filter may also include elements of artificial intelligence (AI). For example, the user may select one word as a security word and the AI filter may automatically select all synonymous words. The AI filter may enable the user to define a filter is real time at the entry of data via a keyboard. For example, the user may select to secure (i.e., extract and store) some proper names and may instruct the filter to secure names such as Block, Smythe and Cherry. During input of the plaintext, the system may detect Smith and ask the user if he or she wants to secure (a) all proper names in a common name dictionary collection and/or (b)

all names with spellings similar to the filter input data, Block, Smythe and Cherry. As is known in the art, AI typically uses inference engines to define one pathway or to outline a course of action. The filter or extraction engine discussed herein can be configured with AI, inference engines, neural network systems or other automatic systems to carry out the functionality described herein for the dynamic operation of the security system.

[099] The system and methodology described herein also encompasses parsing the plain text document by bit count, word, word count, page count, line count, paragraph count and parsing based upon any identifiable document characteristic, capital letters, italics, underline, etc. Algorithms may be implemented to parse the plain text document. The target of the parsing algorithm (a bit count, word, letter, etc.) is equivalent to the "security word, character or icon, data object" discussed herein. The parsing occurs with the filtering of the plain text source document 100 and the subsequent storage of extracted data apart from remainder data.

[100] In a basic configuration, the common text or the remainder data is stored in common storage memory 108. This common or remainder data store is identified as A-com generally referring to a segmented memory in a PC or a computer A in a network (LAN or WAN). Remainder data storage 108 may include a confirm storage signal function 111 to send back a confirm storage signal to the data input device generating source plaintext document 100. The extracted data file 106 is stored in a different memory computer storage 110 (B-ext). In a preferred embodiment, memory segment 108 (A-com) is at a different location than computer storage memory segment 110 (B-ext). In a PC embodiment, memory A-com is a different memory segment than memory B-ext. In a networked embodiment, computer storage 108 may be on a different computer as compared with computer storage 110. In an Internet embodiment, common text or cleansed text storage is at one

Ex. 1002, Page 38

web site (which may be one computer) and the extracted, high security data is stored at another web site, buried web page or other Internet-accessible memory store location. In any event, the remainder text is stored in a memory A-com and the extracted data or high security words, characters , icons or data objects are stored in memory B-ext. After storage of the extracted data in memory 110, a confirmation indicator 113 may be generated to the client computer or the computer handling source plaintext input document 100 (the originating computer system).

[101] As a simple example, the program configured in accordance with the present invention, could automatically detect entry of all credit card numbers types into a user's computer. The filter is set to detect the unique credit card sequence and data string. Assuming that the user's computer is operating a browser and the user is communicating with a server on the Internet, the user's computer would filter out the credit card number and send the number to a secure storage site. The secure storage site is owned, operated or leased by a trusted party. The extracted data, i.e., the credit card data, is stored at the trusted site. The URL or other identifying data is sent to the vendor from which the user wants to purchase goods and services over the Internet. When the vendor seeks to complete the transaction, the vendor sends a request code to the secure site, the trusted party at the secure extracted data storage site debits the user's credit card account (or otherwise debits the user's bank account) and sends an approval code to the vendor. In this manner, the vendor is never given the user's credit card – the card number is sent to a trusted party automatically by the filter in the security program described herein. The security program may be incorporated in a browser to automatically protect credit card data, personal data (as a method to become anonymous on the Internet), etc. from being deliberately broadcast to others on the Internet or to block others from snooping into the user's personal data while the user communicates over the Internet.

Ex. 1002, Page 39

[102] In a further enhancement of the present invention, the computer or data input device handling source plaintext document 100 may also record the location of A-com 108 and B-ext 110. The location data is called herein a "map." A memory mapping function is utilized. The map may be stored in a third memory location 112. Memory location map 112 may be a segment of the memory of the data input computer originating plaintext 100. The map may be encrypted for security reasons.

[103] As a further enhancement of the present invention, the user, prior to initiating the security system, may be given a choice of filtering out all the uncommon words or words not found in the dictionary and adding certain security sensitive words, characters , icons or data objects to filter 102. The added words or terms are filtered out with the uncommon words. Of course, the user may be required to manually input all security words or download the security word filter from the Internet or another system on the LAN. For security systems having multiple security levels, a plurality of filters would be created, each filter associated with a different security level. Further, multiple security levels would require, in addition to remainder text document or data 104, a plurality of extracted data documents 106. The common or remainder text document or data 104 would still be stored in remainder computer storage A-com 108. However, each extracted data document 106 would be stored in a respective, separate computer memory segment or computer B-ext 110. Separate storage of a plurality of extracted data at multiple, separate locations in B-ext is one of the many important features of the present invention.

[104] In view of increasing levels of security relating to (a) the storage location A-com; (b) the transfer of remainder text document 104 to memory computer storage A-com 108; (c) the storage of map 112 (possibly encrypted); (d) the creation, storage or transfer of filter 102 (possibly

Ex. 1002, Page 40

encrypted); (e) the storage of extracted data at memory storage B-ext (whether singular or plural storage sites); and (f) the transfer of extracted data thereto, the system may include an encryption e feature. The encryption e function 115, 117 and 118 is diagrammatically illustrated in FIG. 1A.

[105] The program of the present invention can be set to extract critical data (a) when the plaintext or the source document (data object) is created; (b) when the source document or data object is saved; (c) on a periodic basis; (d) automatically; (e) per user command; (f) per ascertainable or programmable event; and (g) a combination of the foregoing. Timing for storage of the extracted data is based on these aspects. Reconstruction of the data object or plaintext may be (a) automatic and substantially transparent to the user; (b) based upon manual entry of security clearance data; (c) periodic; or (d) a combination of the foregoing dependent upon outside events and who is the author of the data object or other definable aspects of the data object, its environment of origination, current and anticipated security threats and its environment of proposed reconstruction. The timing for the extraction, storage and reconstruction is oftentimes dependent upon the level of security required by the user and/or his or her organization.

[106] FIG. 1B generally diagrammatically illustrates the major features of a reconstruction routine or system. The user, typically at a computer terminal, inputs a reconstruction request 120. The system first executes a security clearance protocol routine 122 in order to determine whether the user has the proper security clearance. The security clearance may be thought of as a security clearance control. If multiple users are permitted access to the documents and those multiple users have different security clearances, the security clearance protocol determines the level of security clearance and, hence, the full or partial reconstruction of the plaintext. The security code input by the user is checked against a security code database or list 124. Clearance is provided in step 126.

Ex. 1002, Page 41

The location of the map and, hence, the location of the remainder data A-com 108 and extraction is provided to the user's computer in step 128. This may include obtaining a copy of the map 130 showing the location of memory segments in (a) the local computer; (b) the LAN or WAN; or (c) the Internet storage sites. The storage segments are A-com 108 and B-ext 110. The common or remainder data is downloaded or transferred or made available to the user's computer as shown at the output of map location and data step 128. Typically, the extracted or security sensitive data from B-ext is downloaded. As described hereinafter, the data can be reconstructed as a complete electronic document in function 130 or may be reconstructed only as a visual reconstruction in step 132. Visual reconstruction is discussed later. Function 130 operates as a compiler to gather the extracted data and remainder data into a single plaintext document. If the data object represents sound or audio signals, reconstruction and play back may require a speaker output in function block 130. In a telecommunications implementation of the present invention, the input would include a microphone or audio detector (supplemental to the input device for document 100), an analog to digital converter (possibly with a voice to digital converter), the filter, extractor, storage facilities at least for the extracted data, and at the output of the system, a converter to audio and an audio announcer. The recipient of the secured data stream or message would be required to clear a security clearance and possibly obtain a decoding key prior to listening to the entire, decoded message. The key and the security data is separately downloaded to the recipient's device.

[107] If remainder data in A-com memory 108 and extracted data in B-ext computer memory 110 is encrypted, the reconstruction process includes a decryption step. Encryptors and decryptors are relatively well known by persons of ordinary skill in the art. Further, the filter 102 (FIG. 1A) may include some encryption routine operating on the data object (plaintext) during the

Ex. 1002, Page 42

filtering. A simple encryption may include substituting "dummy" text or images for the security words and keeping a pointer to an encryption key document mapping the security words with the dummy words. The filter may be stored or may be destroyed at the option of the user. Storage of the filter impacts the degree of security of the entire data system but storage of the same filter enables the user to reuse the filter at a later time. Encryption of the stored filter increases the security of the data. Creation and storage of map in memory 112 also impacts the degree of security of the system. However, if the filter 102 is destroyed and all copies of the map are destroyed on the user's computer originating plaintext document data 100, and the map is stored offsite in a third computer memory location 112, this offsite map storage may enhance the degree of security of the data. The originating computer processing plaintext 100 may be scrubbed to remove all reference and copies of the plaintext, remainder text, extracted data map storage data, etc., i.e., a deletion routine may be employed on the data input computer.

[108] FIG. 2 diagrammatically illustrates a personal computer or PC computer system 140, a second PC or computer 142, and a third PC -3. PCs 140, 142 and PC-3 are connected together via a network 145(LAN or WAN) and are also connected to an input/output device 146 that may be generally described as a router or a server to an outside communications system. The input/output device 146 is connected to a telecommunications system 148 which leads to Internet 150. The Internet is a global computer network. Internet 150 is coupled to a plurality of servers, one of which is server 152. Server 152 may be designated as an application service processor ASP. Internet 150 also includes various computer memory storage devices such as computer storage I-com 154, computer storage I-ext 156 and computer storage map 158. Computer storage enabling the store of

extracted data includes a security level clearance module 157. Similarly, map computer storage 158 includes security level clearance module 159.

[109] As stated earlier, the present data security system can be implemented on a single personal computer 140. In this case, different memory segments or hard drive 168 may be used for A-com and B-ext. Typically, PCs include a keyboard or data input device 161, a display 163, a central processing unit CPU 165, a video board 167 having video board memory 169, a fixed disc hard drive 168, a RAM 166, and input/output device 164, a removable memory media floppy drive 162 and a removable compact disk (CD) read-write (CD-RW) device or drive 160. The system may include other removable disk drives, tape drives, or flash memory units. Internal units CPU 165, video board 167, hard drive 168, RAM 166 input/output device 164, floppy drive 162 and CD-ROM device 160 are all coupled together via an internal bus 171. Bus 171 represents a plurality of buses as is known to persons of ordinary skill in the art.

[110] One methodology of implementing the present invention utilizes distinct memory segments which may be designated in one or more of the following: hard drive 168, memory in a removable disk in floppy drive 162, memory in a removable CD disc in CD-RW device 160, and, to a very limited extend, RAM 166. In this manner, the user may select, generally at the outset of the process, that the extracted data memory storage B-ext 110 be stored on a floppy (removable memory) via floppy drive 162 or a CD via CD-RW drive 160. The user can then simply remove the floppy or the CD and carry it with him or her. To reconstruct the data, the operative program, generally discussed above would have access to the floppy or the CD and particularly the memory location of the data on the floppy and the CD in order to reconstruct the entire plaintext document 100 (see FIG. 1A). Alternatively, different portions of hard drive 168 may store A-com and B-ext.

Ex. 1002, Page 44

Of course, the computer system may utilize tape drives and memories or flash card, programmable memory.

[111] In a local area network or wide area network implementation, PC 142 includes memory similar to memory units described in PC 140 and a memory segment may be set aside in PC 142 separate from the common data or remainder data storage typically placed on hard drive 168 in PC 140. As a further expansion of the present invention, the extracted data (that is, the high security data), may be stored on computer storage I-ext memory unit 156 via Internet 150, telecommunications system 148 and router/server 146. In this manner, the common data or remainder data is stored on hard drive 168 and the highly sensitive data is stored off site in a secured location. Access to that secured location may be limited via security layer 157. If the user implements an encryption system (see encryption e 118 in FIG. 1A), the extracted data is further secured by the encryption during the transfer from computer 140 through network 145, router/server 146, telecommunication system 148, Internet 150 and ultimately to computer storage I-ext 156.

[112] The present invention may also be embodied utilizing an Application Service Provider on server 152 and in a client-server network.

[113] An implementation of the present invention over Internet 150 most likely includes the use of a uniform research locator or URL for map memory computer 158, computer storage I-ext 156, computer storage I-com 158 and ASP server 152. In a client-server environment, server 152 acts as a server generally commanding the operation of client computer 140. Of course, persons of ordinary skill in the art recognize that the server may be located on the local area network 145 rather than being interconnected with Internet 150 as shown in FIG. 2. The claims appended hereto are meant to cover the alternative embodiments.

Ex. 1002, Page 45

[114] As an example of a client-server or web-based implementation of the present invention, the user at computer 140 may define the filter 102 as described above, and input data (plaintext) via keyboard 161 or load plaintext data from floppy drive 162 or CD-ROM drive 160 into RAM 166. In any event, whether the plaintext data is input via keyboard 161 or copied or accessed from floppy drive 162 or CD-RW drive 160, the plaintext data is filtered as discussed above in connection with FIG. 1A. Prior to filtering, it would be appropriate for the user at computer 140 to identify where the remainder data or common data will be stored and where the extracted or high security data would be stored. A simple program may automatically select the secure store location. The system is sufficiently flexible to enable the user to select local storage on different memory segments of PC 140 (hard drive 168, floppy drive 162, CD-RW drive 160) or be flexible enough to enable user at computer 140 to designate off site storage of the high security data (extracted data) and/or the common or remainder data. An automatic store routine may only require the user to accept or reject to preferred first security level, second security level and higher security level stores. The off site data storage process may include activating server 152 and enabling the server to take over the process directly from user 140. In other words, the user at computer 140 could call up the URL of the server 152, the server could request certain user information (user name, password), and would request data from the client computer to establish the filter pursuant to input selected by the user. The client computer may (a) filter the plaintext thereat or (b) send the data to the server for filtering. The server could store data either locally on computer 140 or remotely at computer memories 154, 156. After storage of the data at any of these locations, the server 152 may establish a map and store the map in memory location 158. Of course, remainder data (cleansed, plaint-text data)and the map may be stored at ASP 152 or client computer 140. The map, if stored at map

storage 158, may be downloaded to the user at computer 140. The filter may be stored at computer 140 or may be stored at a secured location on server 152. Alternatively, the map could be destroyed on user computer 140. The filter could also be destroyed on user computer 140. Of course, the filter could be stored in a fourth remote location (not shown), different from I-com 154, I-ext 156 and map computer memory 158. Storage of the map and decryption keys is a critical, high security task. Appropriate security measures should be utilized to protect those items. Local removable memory storage on disc in floppy drive 162 or disc in CD-RW 160 may be reasonable. All traces of the map, the filter, the encryption key, the extracted data, and possibly the remainder data may be scrubbed or deleted from all computer memories (by write-over or disc reformat routines) other than the "com" and "ext" storage sites. Deletion of all URLs, links, x-pointers, etc. is also recommended for high security applications. Deletion systems are known to persons of ordinary skill in the art. For multiple security levels, multiple web site for storage of cleansed plaintext, first, second, third and higher security level extract text is preferable. Where the community of interest has access to the targeted and protected data via the Internet, multiple secured storage locations, multiple stores for filters, for encryption keys and for maps locating the secured stores is provided by multiple storage locations distributed throughout the Internet.

[115] To reconstruct the document, the user at computer 140 would be required to call up the URL of server 152 and input the appropriate security code. The server 152 would then call up and download data from various memory locations whether they be memory locations on computer 140 or memory locations I-com 154, I-ext 156 and map memory 158. The system compiles the entirety of the plaintext document by gathering the dispersed components thereof or compiles partial reconstructious for different levels of security. By implementing different security levels, the system

Ex. 1002, Page 47

is dynamic enough such that server 152 can easily locate the various extracted data levels based upon various security codes representing different security levels, as those codes are input by the user at computer 140. Multiple security codes, at the inception and during the process, may be utilized. The user may be required to input security codes at multiple times during the reconstruction or compilation process.

[116] It should be noted that computer storage 154, 156 and 158 may be located on the same computer or may be located on different computers spread throughout the Internet. If the storage units are different computers spread throughout the Internet, computer storage 154, 156 and 158 would each have their own URL or Uniform Resource Locator. In any event, during reconstruction, the server 152 gathers the information and downloads the information into RAM 166 of computer 140. This download may include a first download of the common or remainder data from I-com 154. At a separate time, which may or may not include a decryption routine, the extracted from I-ext 156 is downloaded. Preferably, other than inputting initial security codes and any required or desired intermediate security codes, the system operates automatically without further input from the operator at client computer 140. The download of both data sets may be simultaneous in that the download is not humanly perceivable. This is especially true if storage in different memory locations in PC 140 is utilized.

[117] The role of server 152 may be expanded or reduced dependent upon the desires of the user and the degree of security necessary. For example, server 152 may only enable separate storage of extracted data in I-ext 156. In this limited role, server 152 would require the input of a proper security code and clearance prior to identifying and enabling the download of extracted data from I-ext 156.

Ex. 1002, Page 48

[118] In an expanded mode, server 152 may be involved in filtering the data, extracting the security sensitive words, characters , icons or data objects to obtain extracted data and remainder data thereat, separately storing the extracted data from the remainder data (extracted data being placed in computer memory I-ext 156 and remainder data being stored in common remainder data memory I-com 154) and then permitting reconstruction via separate or combined downloads of the remainder data and the extracted data into computer 140.

[119] FIG. 3 diagrammatically illustrates a system diagram for various reconstruction routines. A complete reconstruction is shown as security level path A. This involves an electronic integration of plaintext in step 202 resulting from the complete electronic reconstruction of document 100. For example, a merge may occur between the extracted data and the remainder data or common text data. The document is completely compiled in this process. Placeholders in the remainder document are utilized to locate and insert the extracted data. Most likely, there will be no process controls imposed on the integrated document as shown in step 204. In other words, if the user at computer 140 has the proper security clearance, he or she could download or recreate the entire original source, plaintext document and the user would be entitled to edit the document or change it in any way or copy it and reproduce it.

[120] The second level of security, path B, results in storage of the common or remainder data in a different memory location on the hard drive 168 as compared with the extracted data. This is noted in step 206. Another words, in a simple example, hard drive 168 or RAM 166 would hold a copy of a remainder data document and another copy of the extracted data document, that is, two separate documents. Since two documents are available in RAM 166 or hard drive 168, these documents are stored in different locations in the memory. In step 208, a map showing the memory

Ex. 1002, Page 49

location of the common or remainder document and the extracted data document is provided to computer 140. Step 210 commands the processor CPU 165 in computer 140 to interleave the extracted data with the common or remainder data in the video board memory. In this process, the extracted data would typically have placeholders for the missing remainder data. Otherwise, control codes to locate the extracted data into the remainder data would be executed by CPU 165 to properly place the extracted data into the "visual space" of the placeholders in the remainder data document. The extracted data document may have placeholder for the remainder data. Some type of register between the two image documents may be necessary. The compiler, in this embodiment, gathers the document elements and visually compiles and presents the plaintext to the user.

[121] FIG. 3A diagrammatically shows that video board memory 169 is loaded with remainder or common data 1 and a different location of the video memory is loaded with extracted data 1. The next video memory location is loaded with common data 2 and then a different video memory location is loaded with extraction data 2. Since the refresh rate of computer monitor 163 is fast, the display 163 will show the common or the remainder data and then show the extracted data such that the user could not humanly perceive a difference in the document. However, the user could not copy the document from display screen 163 (a "screen shot") since the document is never electronically integrated into a single document. There is only a visual presentation of the combined document by interleaving the extracted data with the common or remainder in the video memory 169. Step 212 notes that the user may be limited in his or her ability to process, edit and store the reconstructed and presented plaintext document.

[122] Security level path C recognizes in step 214 that the data is stored in different memory or computer locations. In this situation, two video boards, video board A and video board

B are shown as board 216 and 218. Video board 216 drives display monitor 220. Video board 218 drives display monitor 222. Display screens 220, 222 are overlaid atop each other. Video board 216 is fed with common or remainder data from the remainder data store (see I-com store 154 in FIG. 2) and video board 218 is fed with the extracted data from the extracted data store, for example, I-ext store 156. In this manner, as noted in step 224, the user is presented only with a visual presentation or compilation of the plaintext. Since there was physical separation between video monitor 222 and video monitor 220, there is no electronic integration at all of the plaintext document. Hence, the ability for the user to do any significant editing on the plaintext document is blocked or prohibited because the user only has access to either the data on video board 216 or the video board 218.

[123] Security level path D shows that the extracted data may be parsed or further separated based on a plurality of security clearances in step 226. Step 228 recognizes that the system can repeat process and security process paths A, B and C only with portions of the extracted data presented to the user based upon the user's security clearance.

[124] FIG. 4 diagrammatically illustrates the major components of a flowchart for the data security program. It should be noted that this flowchart may be truncated to limit user selection of certain items. The system would be pre-set to contain these features. Step 230 initializes the system. Step 232 enables the user to designate various levels of security for the activity which he or she will soon engage. The system, in step 234, enables the user to define the levels of security parameters. The following Security Table gives some examples of the type of security that may be available to the user.

Security Table

to whom

Ex. 1002, Page 51

to where

when (time of day, day of week, month, floating but predetermined time frame)

why (purpose, match purpose to other security parameters or to certain predetermined criteria)

how (through what medium (LAN, WAN, Internet, direct dial link), download to what site or destination)

how long (duration) the reconstruction process will be permitted per each security clearance level

how much (different security levels enable reconstitution of documents and data with different amounts of secure data therein)

timing systems may require synchronization for a standard clock (i.e., atomic clock)

[125] As an example of a truncated or pre-set program, a client-server system over the Internet may have URLs designating storage sites and an ASP 152 (FIG. 2) controlling storage. In this pre-set system, the user does not select the sites. The sites may be randomly selected by ASP 152. The ASP may use artificial intelligence AI to locate secure extract data storage sites. AI or inference machines can ascertain (a) traffic on communications channels, (b) storage limit issues, (c) transmission failures in the communications links, and (d) the degree of security necessitated by exterior events, i.e., terrorism alerts, virus alerts, war, data security warnings posted by trusted sources, MicroSoft, Norton, NASA, DoD, CDC, FBI, etc. Higher security alerts trigger the AI configured storage locator and facilitator to locate memory stores in higher secured places. These

Ex. 1002, Page 52

higher security facilities may be more costly, may be located in more stable countries or on more stable servers and may have greater degrees of encryption capabilities.

[126] The user, in step 326 can designate the location of the filter, the common storage area for the remainder data, the extraction data storage and potentially multiple data storage areas or segments. The user may enable an AI filter design. Step 238 permits the user to engage or disengage encryption and, if engaged, establish the degree of encryption for the system. Step 240 enables the user to define the parameters of the filter. The user can retrieve a preexisting filter or may define a new filter for each data security session. These filters may consist of dictionaries or any type of compilation of words, characters, icon, data objects or pixel formation or any indication that can be perceived by the computer system. Granular extraction of data elements in a data object may be permitted. Step 242 recognizes that the user either inputs a preexisting plaintext document or types data into the system. In any event, the plaintext document is fed through the filter. Step 246 extracts the security data from the input document. Step 248 stores the extracted data. The extracted data may be encrypted prior to storage. Step 250 conducts an error check on the extracted data. This error check is helpful in discerning problems in the storage of the data prior to closing down the data security system. Step 252 stores the common data or the remainder data. Step 254 conducts an error check on the common or remainder data. The decision step 256 determines whether the user has selected a "destroy filter" command. If not, the filter is stored with or without encryption in step 257. If YES, the filter is destroyed with a deletion routine. Typically, deletion is complete erasure of all traces of the file including, in high security systems multiple write-overs or disc reformatting. Step 258 stores a map. The map may be stored locally or remotely as described earlier. The system ends in step 260. All traces of these data elements or objects may be swiped clean or removed from

Ex. 1002, Page 53

whatever computer system generated the data objects or processed them, other than the memory storage locations. Deletion of data also includes the concept of deletion of data transmission paths, URLs, storage site locations and all temporary memory stores. Deletion of file location in the root directory of hard drive 168 of computer 140 is preferable in high security systems.

[127] FIG. 5 diagrammatically illustrates basic flowchart features for the reconstruction process. Step 302 accepts a request to reconstruct the secured data. Step 304 queries a local map and the security system or protocol. In a preferred embodiment the user would have to input several passwords, one of them being a local password on computer 140. A local map which may be accessed only through the password, may simply identify the URL of server 152. Decision step 306 determines whether the local password is acceptable. If not, and error step is indicated in step 307, the attempt to log on to the security system is noted in step 309 (an audit trail), and the system either branches to repeat step 311 or bars the user from further activity in step 313.

[128] Returning to decision step 306, if the password is locally acceptable, the YES branch is taken and the system executes step 308 which releases a reconstruction request to the common storage facility I-com 154 or A-com 108 (FIGS. 2 and 1A-B). The system in step 310 logs the user in, as well as time and date and the data regarding the request. In step 312, a download from the common data storage is provided to RAM 166 or hard drive 168.

[129] In step 314, a query is made to obtain the remote map from the remote security system. The decision step 316 indicates that the user again successfully inputs his or her security code. If not, error routine 317 is activated, the password failure is noted in step 319 (an audit trial), and the user is given an opportunity to repeat in step 321 or is barred or prohibited from further activity in step 323. If the user has correctly input the security code, the system in step 318 releases

Ex. 1002, Page 54

the keys (to decrypt) and the map and releases the reconstruction request to the remote storage for the extracted data. This could be computer storage I-ext 156 or computer storage B-ext 110. In step 320, the user's access to the extracted data is logged in along with the time and day and type of data request. In step 322, the system downloads the extracted data into RAM 166 and/or hard drive 168 of computer 140. In step 324, an error routine is operated on the extracted data in order to insure that the extracted data properly matches the common or remainder previously stored. Decision step 326 determines whether the error routine properly generates the correct count or output. If not, the system in step 327 indicates an error, in step 329 the system deletes the common files and the extracted files and the system in step 331 logs in the failed attempt. If the error checking routine on the extracted data is acceptable, the YES branch is taken from decision step 326 and the system, in step 328, proceeds to display the plaintext document or to integrate the plaintext document pursuant to the security clearance initially input by the user. Step 330 ends this process. The end process may entail encrypting the data again and swiping clean all traces of data objects from the memory stores and computer handling units. Of course, every use of encryption requires decryption of the data prior to reconstruction.

[130] The system may incorporate various types of security systems or routines.

pass word

pass phrase

multiple choice questions and answers

initial, intermediate and subsequent security clearance routines

biometric security routines (voice, fingerprint, signature, eye or retina scan)

Ex. 1002, Page 55

[131] The reconstruction routines may be interrupted or the security system automatically activated or initiated upon the occurrence of externally generated triggers or upon certain predetermined conditions or conditional events. Limited extraction, security clearance, release of data and reconstruction limits may be imposed. Artificial intelligence (AI) engines, inference engines or neural networks may be implemented to vary the permitted level of reconstruction via the security clearances. In other words, the AI system, as applied to reconstruction, may, relatively independent of the filter and storage processes, increase the necessary security levels permitted to access and generate full or partial plaintext recreation.

[132] The display systems 220, 222 in FIG. 3 include CRT monitors, LCD screens, projection screens and combinations of those systems.

[133] The audit trail to monitor reconstruct and reconstruction attempts may include adding a time/data stamp to the remainder data and/or the extracted data prior to storage and a cross-check to the audit trail log during the reconstruction process.

[134] Placeholders in the remainder document may be:

blank spaces

data symbols or elements "---" or "xxx"

false data

clearly erroneous data "ABC Company" or "Baker"

chaff or hash marks

messages

bar code

serialization data

Ex. 1002, Page 56

alerts

links to other data objects

null set indicators "[ ]"

URL or website addresses

[135] It is believed that the present invention is faster, during reconstruction, than standard encryption techniques, on the order of 100 to 1,000 faster.

[136] The system and method described herein may operate substantially automatically, that is, without operator intervention, other than the security clearance function. The clearance function does require some type of operator authentication prior to retrieval of the extracted and remainder data.

[137] The system and the method may operate automatically in that the plaintext or originating data could be identified by a party desiring security. The system could obtain that data from any data input device (hard drive memory, floppy drive memory, flash card memory, personal data assistant (PDA), or any other type of data input device), filter the data, separate the extracted text or the remainder text, encrypt (or not encrypt) the data, separately store the extract and remainder data (all automatically, that is, without operator intervention). Hence, it is not necessary that the system operate with significant operator or manual intervention. Of course, the system may also operate on a plaintext document or data object that is being created "in real time" by an operator and keyboard, mouse or other type of data input device.

[138] The automatic operation of the system and the method can be caused by a triggering event. This triggering event may be a security attack (generating a trigger to start the gathering of plaintext, filtering, extraction and storing) or may be any other type of trigger such as a building

Ex. 1002, Page 57

burglar alarm, door alarm, fire alarm, or virus detection algorithm trigger. The event may be a time of day, week or month. It may be n seconds after the user stops typing on a keyboard. It may be a timed back-up feature.

[139] Multiple filters may be utilized in the system and in connection with the method. These multiple filters may be useful in the operation of the system with a plurality of security levels. Each filter could filter out different levels of security sensitive items and each bundle or group of security sensitive items (from each distinct filter) could be stored at different computer storage locations. Multiple filters, multiple security levels and multiple storage areas may also include multiple encryption routines and decryption routines. Encryption and decryption routines can be related to the level of security of a particular group of data.

[140] Multiple maps may also be provided for singular or multiple storage of extracted data and remainder data. These maps may or may not indicate the originating point of the data. Maps can be parsed such that an intruder, upon discovery of a single map or map portion, could not locate the storage locations of all piece of the extracted data and remainder data. maps may also be encrypted. The map may also be stored at a distinct map store location.

[141] The concept of partial reconstruction also includes the concept that a portion of the plaintext would be reconstructed and the unreconstructed portions of the plaintext could be encrypted or could show blanks or other symbolic indicators. See the placeholder table above.

[142] Partial reconstruction of the plaintext also includes a concept that the security sensitive items or materials may be subject to different types of encryption. Hence, a single plaintext document may have multiple levels of security and multiple levels of encryption wherein each encryption has a different level of security assigned to it.

[143] The present invention can also be configured to provide a computer network which transparently establishes and manages the separation of user-based communities of interest. The separation is accomplished by extraction pursuant to security levels, dispersion of data into secure storage facilities (memory stores) and reconstruction based upon the assigned security level. A low level security clearance results in only partial reconstruction of the plain text or source document. These user-based communities of interest are a plurality of users each having respective security clearances. As described above, each successively higher level of security clearance permits the user to see greater degrees of reconstructed plain text obtained from the extracted data stored in extract stores and the remainder data from the remainder stores. By integrating encryption (and necessarily decryption), separation of user-based communities of interest are established such that the users in a particular community are permitted access to some or all of the plain text data based crypto-graphically separated communities and need to know security levels.

[144] FIG. 6 is an exemplary computer network diagram showing various user communities. The telecommunications network 402 is connected to the server application server provider ASP 452 and to various networks and personal computers or PCs. the PCs may be computer work stations. Network A 404 is coupled to telecommunications network 402 via an input/output unit 406. Network A is coupled to various PCs identified in FIG. 6 as PC-4, PC-5 and PC-6. Of course, Network A could be coupled to other PCs not illustrated in FIG. 6. As described earlier, server 452 can facilitate remote or offsite storage of extract data and remainder data in store 1, store 2 and/or store 3. Further, the map showing the storage location may be encrypted and stored in any one or more of these stores. Also as described earlier, the memory in one of the PCs, for example PC-4, PC-5 could be utilized to store extract data and remainder data from PC-6 and PC-6 can be configured as the input data computer. Hence, the present system and methodology

Ex. 1002, Page 59

encompasses the concept of local storage and remote storage. On the local level, the storage begins by storing the extract data at different locations in the hard drive of the PC. The next level higher is storing the extract data in removable computer media such as floppy disk, removable tape drives, CDs etc. associated with the PC accepting data or associated with a server on Network A. The next higher level of extract store is storage of the extract data on a server or other computer in a particular network. If PC-6 is designated as the input computer, the extract data may be stored on PC-4. Of course, PC-4 could be designated as the server for Network A.

[145] PC-7, PC-8 and PC-9 are coupled to telecommunications network 402. Network C 408 and Network B 410 is coupled to communications network 402. The lines, one of which is line 409 extending from Network C 408, represent a plurality of computers or workstations coupled to Network C. Line 411 represents a plurality of workstations or computers coupled to Network B 410. In an e-mail implementation of one embodiment of the present invention, PC-7, PC-8, etc. may represent computerized devices accepting e-mail (personal data assistant, pager, cell phone, etc.). The sender and the e-mail addressee may utilize simple computerized systems to communicated via e-mail. Further, the network may be any telecommunications network including wire, cable, cellular, wireless, satellite, IR or RF systems.

[146] FIG. 7 diagrammatically illustrates a flow chart showing the key component steps for the multiple layer security program for the community of users. The "community of interest" system described herein enables persons and organizations at the same security level to share data on a peer to peer level. Further the security system may operate automatically, with respect to extraction, storage and reconstruction, such that the peer to peer dissemination of data objects is quickly and readily available to all at the same or higher security levels. Step 420 initializes the program. Step 422 enables the user, administrator or system operator to designate multiple levels of security, that

Ex. 1002, Page 60

is, multiple words, characters, icon, data objects, or whatever, for each security level and further to define encryption for each security level. The designation step 422 also includes identifying the communities of interest and the particular security level and security clearance for each community of interest. One example of various security levels for communities is set forth below in the Community Security Level Table which is keyed to the computer network diagram of FIG. 6.

Community Security Level Table

Security level	Community Group
High	PC-7; PC-8
Medium high	all high group plus Network B
Medium	all above plus Network A
Low	all with nominal clearance
Special set medium	PC-7; PC-9; Network B

[147] Further, designation step 422 will include identifying the words, phrases, icons or data objects subject to security concerns and the potential location of the extract data and, if necessary the remainder data and the degree of encryption. The following Selection Table provides some examples.

Selection Table

Level of encryption/storage	type or category of word or phrase; input specific word, phrase
High, web-based storage	dollar values, names of streets, countries, "Smith" and 5 words about "Smith," "avocado"
Medium high, remote storage	all addresses, all names
Medium network storage	all family names, all client names

Ex. 1002, Page 61

Low, encrypt and separate store in local memory — all items not in dictionary

[148] As an example of various encryption methodologies, the following Encryption Table is illustrative.

Encryption Table

DES, random pad A ("r. pad A")

Huffman, r. pad B

Crypto API, r. pad 7

Two fish, r. pad C-2

Blowfish

RC4

Skipjack

Ghost

[149] In FIG. 7, step 424 executes or enables the security program with multiple filters, multiple encryption levels and multiple storage levels. Each one of these filters, encryption levels and storage levels correspond to the security level for the various communities of interest. Step 425 responds to an inquiry from a user to reconstruct the document. Step 426 accesses the user's security clearance and the particular inquiry. Decision 428 determines whether the inquiring party is entitled to full or partial access to the source document. If not, the NO branch is taken and the system, in step 429 adds placeholder substitutions. Step 429 may be optional. If YES, the system reconstruct pursuant to the clearance level in step 430. The following provides an example of multiple level encryption utilizing placeholder substitution.

Ex. 1002, Page 62

Multiple Level Encryption

Applicants must be _____ zzxx xx _____xxx ____ _____ _____ citizens and have a high school diploma or equivalent. They must possess a valid subsubsub driver's license and qualify for top SUBWORD_____ clearance.

[150] With this multiple level encryption, substitutions may be utilized "subword" to indicate to the user with a less than superior security level that a certain word, term or phrase has been extracted and stored by he or she is entitled to know that substitute word, term or phrase has been inserted into the plain text document. Of course, any type of substitution character may be used for the placeholder.

[151] In step 432, the system displays the plain text in a normal format or utilizing a split or bifurcated video memory or utilizing overlay display screen. FIG. 3 and the description of that figure set forth above describes the normal display in steps 202, 204, the split video memory display in steps 206, 208, 210 and 212 and the overlay display system in steps 214, 216, 218.

[152] The system, in step 434, monitors and logs the location of the user making the inquiry, the type of inquiry, the time, day, date, clearance level and access level and logs all modifications to the plain text source document. One example of the log is set forth below in the Security Report Table.

Security Report Table

Privacy Scrubber Report

source file: path\filename

scrubbed file: path\filename-scrub

source file: date, time, size

process: date, time

Ex. 1002, Page 63

user: name

system: name

Recovery File

(a) storage location, type of encryption, random key

(b) storage location B....

(c) store C .....

(d) store D .....

[153] Step 436 enables the security program and parses and extracts the data per the security program, filters the data, extracts it and codes it disperses it and stores it as discussed above. The multiple layer security program ends in step 440.

[154] The following Security Level Access Placeholder Table is another example of the type of placeholder substitutions that may be available. The example in the Security Table Access Placeholder Table may be used in conjunction with step 429.

Security Level Access Placeholder Table

[security level 2] intelligence located [security level 4] 20 miles from [security level 4]. He is using the name [security level 4], and dressed as a [security level 4] preacher. With him are his lieutenants, [security level 4] and [security level 4]. He is communicating with the international media through Mr. [security level 4], who resides at [security level 3], [security level 4], [security level 4]. Telephone is [security level 1] and Facsimile is [security level 1].

[155] It should be noted that in order to reconstruct some or all of the plain text source data, some or all of the subsets of extracted data from the extract stores will be utilized dependent upon the respective security level of the inquiring party or user.

Ex. 1002, Page 64

[156] The present invention can also be configured as an adaptive security program which adapts and adjusts the security provisions based upon intrusion into a particular network or attempts to electronically attack or hack into that network or successful hack events. Programs are available to track electronic attacks or hacking attempts. One of these programs is manufactured by Cisco and identified as the Cisco Intrusion Detection System (IDS). The Cisco IDS system can work on a server or on PCs in a network. The Cisco IDS is an electronic intrusion detector, or an electronic attack detector or a hacking monitor. The hack or attack monitor is software loaded into a designated computer.

[157] The output of the electronic attack or hacking monitor loaded into PC 142 (FIG. 2) for example, or loaded into PC-6 acting as a server for Network A 404 in FIG. 6, generates a plurality of attack warnings. The attack warnings progressively and incrementally indicate the severity and degree of intrusion and hacking attacks directed to the computer system. The following Security Level Table illustrates an example of various responses to increasing levels of attacks. These increasing security responses include engaging the filter and extracting critical data and storing it locally; the next level involves storing the critical data on removable storage media; the next higher level involves offsite storage of all security data; the subsequent security alert results in multiple offsite storage for multiple levels of security or critical data and the highest level involves offsite storage of both common data (remainder data) and security data. Of course, other combinations responsive to the hack attack may be provided. The electronic attack monitor may use artificial intelligence AI to (a) assess the severity of the attack, (b) plan an appropriate "secure data" response, (c) select the degree of filter, extraction and/or encryption, and (d) locate secure extract data storage sites. AI or inference machines can ascertain (a) traffic on communications channels, both intra and inter network, (b) storage limit issues, (c) transmission failures in the

Ex. 1002, Page 65

communications links, and (d) the degree of security necessitated by exterior events, i.e., terrorism alerts, virus alerts, war, data security warnings posted by trusted sources, MicroSoft, Norton, NASA, DoD, CDC, FBI, etc. Higher security alerts trigger the AI security monitor to heighten the security level (or to decrease that security level in view of a reduction or withdrawal of an electronic attack). Aspects of AI systems, inference engines and neural networks are discussed above in conjunction with the AI configured filter. These AI aspects can be utilized with an AI configured security sensor.

Security Level Table

Attack (low threat level) Level One

engage filter

local storage - disk drive

encrypt map

Attack (moderate threat level) Level Two

same as Level One but use removable storage media (local)

Attack (nominal attack) Level Three

Engage higher level filter

Off site storage, single storage for all security data

Attack (moderate attack) Level Four

Multiple off site storage, multiple levels of security data

Attack (severe attack) Level Five

Off site storage both common data and security data

[158] Hence, the filtering of data is based upon respective ones of the plurality of attack or hack warnings and the extraction of data and degree of extraction is dependent upon respective ones of the plurality of attack - hack warnings. Storage of the extracted data and the remainder data is

Ex. 1002, Page 66

also based upon the degree of attack which is reflected in the attack - hack warning issued by the monitor.

[159] FIG. 8 diagrammatically illustrates a flow chart showing the key components of the adaptive security program adaptable to various levels of hacker of electronic attacks. Step 460 senses all intrusions and attempts, that is, electronic attacks, hack attacks or hacking actions on a computer or a computer network. This step is equivalent to the output of the attack - hack monitor. Step 462 assesses the current network performance, adjusts the storage location for the extract data (the location of the extract store), the encryption level (the degree of encryption) and the storage of the map showing the extract data storage (if necessary) and storage of remainder data, if necessary given the severity of the attack. For example, during high utilization of the computer network (high utilization in a server computer in a server-client environment), local storage of extracted data may be preferable as compared with offsite storage of critical data. However, if the attack occurs during non-working hours, the performance of the network is very high, and the security system could utilize all the resources in the computer network to achieve the security goal of safe guarding the data during the attack. System resources include processing resources (for encryption/decryption), bandwidth resources to store extract data and any other resources that are critical for the utilization of the security system described herein. Decision step 464 determines whether a threat or attack as occurred. If not, the system takes the NO branch returns to step 460. If YES, the system in step 466 assigns an attack level or a hack warning level to the threat or attack. The system in decision step 468, monitors the network during the attack. If the network performance or the computer performance does not change, the YES branch is taken. If the computer performance or network performance changes based upon or during the attack, the NO branch is taken and the system returns

Ex. 1002, Page 67

to step 466 which reassigns an attack level or a warning level to the next higher or significantly higher warning levels.

[160] After decision step 468, the system executes step 470 which assigns the security level and implements the security program based upon the attack. It should be noted that the administrator establishes the degree of security level, the encryption, the extract store and remainder store (if necessary) for various levels of attacks or hack warnings. The security level assigned to a particular attack warning is implemented in step 470. Decision step 472 determines whether the security program's communication path is clear. For offsite storage of extract and/or remainder data, a communication path is important. If the path is blocked or compromised by the attack, the NO branch is taken and the system in step 473 reassigns the security level to a next higher level or a different, safer security level and returns to step 470. If the security and communications path is clear, the YES branch is taken from decision step 472 and, in step 474, the system maintains the security program. Decision step 476 determines whether sufficient time has passed from the attack. If not, the system loops to step 474. If YES, the system executes step 478 which either permits reconstruction of the user operating the plain text or source document or automatically reconstructs those documents that were filtered, parsed, extracted, and subject to outside storage. The system ends in step 480. To provide additional security, the attack monitor can be configured to monitor security warnings from trusted parties such as MicroSoft, Norton, NASA, DoD, CDC, FBI, etc. Emails or electronic communications from trusted parties can trigger higher levels of security. the attack monitor described above can be configured to accept messages from trusted parties. These messages are equivalent to detecting an electronic attack.

[161] Further, the attack - hack monitor can be configured to monitor and assess other environmental conditions such as fire, power failure, equipment failure, unauthorized physical entry

Ex. 1002, Page 68

into the building, plant, or computer room. These exterior threats or events are monitored by the attack monitor since they may quickly develop into an electronic attack on the secured data retained by the computer system. In response to these exterior events, the attack monitor generates corresponding attack warnings similar in nature to the hack attack warnings discussed above.

[162] There are various methodologies that may be utilized in the adaptive system. The tables that follow set forth these various security methodologies.

Apple Inc. Exhibit 1002

Standard Automatic Defenses Matrix

Mode	Normal	Threat	Attack
Encryption	Targeted Encryption	Full Encryption	Multi Type Encryption
Extraction	Plain-text Extraction	Extraction of Encrypted Data	Extraction of Multi Type Encryption
Distributed Dispersion	Single Storage Location	Several Storage Locations	Many Storage Locations
Display	Single display	Color/Dither Protection	Multiple Displays

Optional Automatic Defenses Matrix

Mode	Normal	Threat	Attack
Substitution of Code Words	None	Partial	Many
Substitution of Misinformation	None	Partial	Many
Controlled Release-Storage	Full	Partial	Conditional
Storage Locations	2	4	10 or more
Time for release	Anytime	Working Hours	Conditional
Authorized Users	Many	Partial	Conditional
What to Release	All	Partial	Conditional
Secret Sharing	None	Two Users	As Configured

Petitioner Apple Inc. - Exhibit 1002, Page 70

Security Meter Module Table

	Normal Mode	Threat Mode	Attack Mode
ENCRYPTION	Targeted encryption	Full encryption	Multi layer encryption
	(Secret sharing)	(Secret sharing)	(Secret sharing)
EXTRACTION	Plain-text extraction	Extraction of encrypted Data	Extraction of multi encryption
Distributed Storage	1 critical storage	few critical storage	many critical storage
Controlled Release-Storage			
	Storage # ID		
	Time for release		
	Authorized Users		
	What to release		
	Special conditions	2 users online	3 or more users
Display	single display	single display	multiple displays
Substitution of code words	No	No	No

20120501 26154101

Normal Work Mode

	Extraction				Storage				
	Level 1	Level 2	Level 3	Level 4	Web	Offline	Remote	Removable	Local
social security	X						X		
credit card	X						X		
included	X						X		
last name	X						X		
number	X						X		
telephone			X					X	
name			X					X	
URL			X					X	
e-mail			X					X	
uppercase			X					X	
initial capital			X					X	
currency				X					X
postal code				X					X
address				X					X
location				X					X
date				X					X

Ex. 1002, Page 72

Threat Mode

	Extraction				Storage				
	Level 1	Level 2	Level 3	Level 4	Web	Offline	Remote	Removable	Local
social security	X							X	
credit card	X							X	
included	X							X	
last name	X							X	
number	X							X	
telephone		X						X	
name		X						X	
URL		X						X	
e-mail		X						X	
uppercase		X						X	
initial capital		X						X	
currency			X						X
postal code			X						X
address			X						X
location			X						X
date			X						X

Petitioner Apple Inc. - Exhibit 1002, p. 71

Attack Mode

	Extraction				Storage			
	Level 1	Level 2	Level 3	Level 4	Web	Offline	Remote	Removable
social security	X					X		
credit card	X					X		
included	X					X		
last name	X					X		
number	X					X		
telephone	X						X	
name	X						X	
URL	X						X	
e-mail	X						X	
uppercase	X						X	
initial capital	X						X	
currency		X					X	
postal code		X					X	
address		X					X	
location		X					X	
date		X					X	

10156102 - 052309

[163] FIG. 9 diagrammatically illustrates a flowchart showing the key components of a multiple encryption program using multiple types of encryption in one document or data object. Multiple levels, types or modes of encryption are utilized in the same document or data object to enable securing data and transparently managing the separation of user-based communities of interest based upon crypto-graphically separated, need to know security levels. These security levels are associated with a plurality of encryption types or with different cipher keys using the same encryption. An example of a multiple level encrypted document is shown above in the Multiple

Level Encryption sample. Different levels or modes or types of encryption are listed in the Encryption Table above.

[164] Step 510 in FIG. 9 initializes the system by organizing different security levels with different encryption types and cipher keys. Also, the program sets filters to create the multiple encryption or ML document or data object. Step 512 filters the document or data object. Step 514 encrypts the extracted data for each security level. These steps 510, 512 and 514 utilize many of the routines discussed above in connection with FIGS. 4 and 7, steps 232, 234, 236, 238, 240, 422 and 424. Step 516 recognizes that the secured document or data object may be stored for later use (with associated multiple decryption), published, distributed, or otherwise utilized to achieve the primary purpose of the document, i.e., to communicate information or to safely store security critical information. Step 518 permits the user, with the proper security clearance to retrieve the document or data object. Step 520 illustrates that the user must retrieve his or her cipher key to decode all or a portion of the ML encrypted document or data object. This step may be manual which engages the user to into certain codes or may be automatic such that the user's computer automatically, without operator input, decodes all or part of the document or data object. Step 522 decrypts the document pursuant to the user's security clearance. Step 524 recognizes that the user may review, re-publish, store, comment on, re-encrypt or otherwise deal and handle the full or partially decoded document or data object. The program ends or otherwise continues with other programs set forth herein. It should be noted that storage of the extracted data may be included in the flow path of the program in FIG. 9 is necessary.

[165] FIG. 10 diagrammatically illustrates a chart showing the key components of the parsing, dispersion, multiple storage and reconstruction (under security clearance) of data. Document or data object 100, in function element 550, is created or obtained by the input computer device. The

Ex. 1002, Page 75

document is stored in a normal manner in customary data store 552. A parsing algorithm function 554 is utilized in parsing step 556. The parsing algorithm, as stated earlier, targets the plaintext document or data object 100 and splits, cuts and segments (that is, parses) the document by bit count, word, word count, page, line count, paragraph count, any identifiable document or icon characteristic, or other identifiable feature such as capital letters, italics, underline, etc. Hence, the parsed document 100 constitutes at least remainder data and data which is extracted or parsed or segmented out. A plurality of data extracts may be obtained. The parsed data (which is both the extract data and remainder data) is then dispersed into storage facilities data store DS 1, 2, 3, 4, etc. Preferably, the parsed documents are encrypted as shown by "e" in FIG. 10. In order to facilitate the potential reconstitution of document 100, a map is stored in a map storage 558. Hence, the dispersement 560 largely spreads out or distributes the parsed document 100 to a plurality of memories in the distributed computer system. These memories may be removable memory devices (floppy disc, removable tape drive, CDs) or may be more fixed devices such as hard drives, Internet storage facilities, etc. Preferably, the map is also encrypted.

[166] Reconstruction step 562 enables a person with the appropriate security to obtain the map from map storage 558, decode the map, gather the dispersed, parsed segments of document 100 and compile the document. This is noted in function 564.

[167] Since the original document 100 is stored in a customary manner in data storage 552, the parsed document stored in multiple data storage units DS1-DS4 provides a unique backup for document 100. The algorithm can employ many different mathematical constructions but is, in the current embodiment, primarily based upon one or more of a bit count, a word, a word count, a page count, a line count, a paragraph count, and identifiable document characteristic, and identifiable word characteristic, and identifiable icon characteristic and identifiable data object characteristic,

Ex. 1002, Page 76

capital letters, italics, and underline found in the plaintext document or data object. Further, the parsing algorithm can generate different security levels wherein parsed segments are stored at different storage facilities having various degrees of security clearance. This establishes a hierarchy of data storage units and corresponding degrees of security clearances. The parsing algorithm may identify unique words or strings of data, i.e., credit card numbers. The hierarchy of security clearances may involve first a password, second a biometric confirmation such as a voice match and a third highly unique biometric characteristic such as a fingerprint or retinal scan. The parsing system enables a large distribution of data in a secured environment. In this manner, if the original data object 100 at customary data storage 552 is destroyed, a person with an appropriate security clearance can reconstitute the original data document 100 due to the secured parsing and dispersal of document 100 through data storage units DS1-DS4 and map storage 558. The parsing may occur on a granular level. In particular, the parsing may occur on a financial document in electronic form.

Financial Document Table

Startcode; Abel, Robert, NMI; 100567; TRANSFER803; To8900586943; FROM3897622891; $700.00; endcode

[168] In the Financial Document Table, the start code and end code is typically represented by a digital code unique to the communications channel, the name on the account has no middle initial (NMI) and the various words "transfer 803" and "to 8900586943" and the words "from" and "$" are represented by predefined numeric or alpha numeric codes. The electronic financial document complies with an established protocol. In any event, financial documents are often times transmitted through electronic communications and telecommunications channels. The present invention, in one embodiment, enables a higher level of security by parsing the financial document or data stream.

Ex. 1002, Page 77

Further, a higher level of security may be employed by extracting identified text or characters and storing the extracted text as discussed above in connection with FIGS. 1A, 1B and 2.

[169] To some extent, the present system can also be utilized for key management and encryption systems.

[170] In a broad sense, the parsing methodology disclosed herein is not based upon the separation of critical versus non-critical or classified versus non-classified security information. The primary focus of the parsing methodology is (1) automatic transparent parsing of data content into granular data groups which are thereafter dispersed to different storage locations in order to maintain a very high level of security with or without encryption; (2) dispersal of the segmented data to different storage locations each which, potentially, demand additional identification or security clearance prior to the release of the stored segmented data, including, possibly, the creation of a digital bureaucracy, in order to hinder or circumvent digital attacks on the plaintext document or data object; (3) proposing and implementing a system wherein the user has a very basic appliance since most of the user's data is stored both locally (customary data storage 552; FIG. 10) and parsed and stored in a distributed system (DS1-DS4) and wherein an important asset is the map stored in map location 558; (4) enabling an institutional system to parse highly confidential information and extract the same in granular form and disperse the same throughout the Internet or other storage locations with or without encryption without compromising the document's security privacy and integrity.

[171] The process involves parsing the documents or content into granular data groups and optionally creating small groups of data wherein the data segments cannot be recognized even to the level of providing 2-4 data objects in each file; dispersing the granular data groups into different storage locations; creation of a map of dispersal to the different storage locations (wherein the map is secured and encrypted and stored); and reconstructing the documents or data content. The

Ex. 1002, Page 78

reconstruction utilizes the map of dispersed and distributed storage and requires the presentation of security clearances such as passwords, biometric information and/or physical identifiers for access at the storage level and potentially at all the other data storage sites. The data is compartmentalized through distributed storage and sometimes requires separate security clearance. This need for presenting additional security clearance at different storage locations (DS1-DS4) creates a digital bureaucratic process which enhances the security level of the entire system. The selection and extraction of data and dispersal of that data to select storage locations can be established under different criteria. For example, one level of criteria extracts last name, address and social security numbers. Another criteria extracts every other line, every third word, etc. The parsing algorithm can utilize random selection or systematic selection as long as the parsing algorithm is documented and utilized in reconstruct step 562. The parsing algorithm may be stored with map and map store 558 or may be stored separately. An additional feature, as discussed above, involves utilizing place holders or adding substitute content to the remainder data of the parsed document 100. The use of place holders and substitute content may be thought of as an algorithm for the parsing. By using place holders and substitute data, private or highly confidential data is masked insuring privacy, security, and confidentiality. The ability to parse the information and/or extract security information is important for financial transactions. The transactions which require account numbers (see Financial Document Table above) are useless without the account numbers. The security of the account numbers, whether identified and extracted or severely parsed and segmented, stored and reconstituted under security clearances, is enhanced by the present system.

[172] To achieve a very high level of security, the system can optionally incorporate a two-men key system. The system automatically separates the selected data stream into one or more data groups and extracts one or more of these data groups and disperses them into data storage DS1-DS4.

Ex. 1002, Page 79

To release the extracted data groups and/or critical content, the reconstruct step 562 may require two persons submitting identification credentials or security clearances. This two-man key method is a further protection against identity theft and insider attacks. The two-men key system can be implemented on a regular basis or on an emergency basis when there is need for a higher level of security.

[173] Financial documents sometimes include substantial amounts of numerical data such as financial projections, balance sheets, electronic funds transfer messages, etc. It should be noted that the extraction may be based upon a particular item such a digit and a nine digit number representing money or may be parsed automatically based upon some parsing facility. Of course, the financial document may also be viewed as a data stream with delimiters ";" separating fields in the data stream. The parsing algorithm may work on the data in each field as well as different fields in the entire data stream.

[174] Most storage facility systems require a map in order to reconstruct the original plaintext document 100. The map may be encrypted and may require a secret key sharing scheme for access thereto. Further, the map may be a physical map (a printout) or may be stored on a removable data storage medium, rather than be an electronic representation. In some instances, a map is not necessary. For example, if the security data or the parsed or segmented data were automatically stored on a floppy disc, the originator of plaintext document 100 could move the floppy disc from the computer system thereby physically safeguarding the security data or the segmented, parsed data. Without the disc, another person or the originator of plaintext document 100 could not reconstitute the document. The originator may deliver the floppy disc to another in order to permit reconstitution. The same is true regarding removable tapes and CD-ROMs.

Ex. 1002, Page 80

[175] Advantages of the present parsing system, methodology and program, include the ability to connect to unsecured networks without adversely affecting the overall security of the plaintext document 100; less dependence on existing security system including fire walls; the reduction of the requirement to keep daily updates regarding vulnerabilities of the computer system originating plaintext document 100; the security of plaintext document 100 is not dependent upon the number of access points into the network or number of users located on the network originating plaintext document 100; there is no damage to the parsed and stored backup version of plaintext document 100 if new security systems are installed wrong or misconfigured and there is no damage if system administrators turn OFF the existing security systems or improperly install or operate the security systems.

[176] The parsing system can operate as a main security operation or an emergency backup system or as a customary backup system. The plaintext source document or data object may be preserved with or without encryption, or destroyed as a further data security step. The parsing and disbursement of data protects plaintext document 100 and insures the survivability of plaintext document 100 if the system originating plaintext document 100 comes under significant electronic or physical attack. That is, if customary data storage 552 is destroyed electronically or physically, the survivability of data in the plaintext document 100 is established by the present system. The storage of granular data groups most likely would defeat any attempt to view the entire content of plaintext document 100. Only verified user users with a confirmed security clearances or identifications verified at reconstruct step 562 and in data storage sites DS1-DS4 are permitted to reconstruct plaintext document 100. Further, the parsing of the system can be triggered based upon an electronic attack, an electronic hack or a physical environmental detection scheme. This system

Ex. 1002, Page 81

immediately protects of the critical data plaintext document 100 with a transparent, automatic parsing, dispersal and storage system.

[177] It should be noted that various aspects of the methodology and program described above in connection with FIGS. 1A-9 can be incorporated into the parsing methodology and program in order to enhance or modify the system.

[178] FIGS. 11A and 11B diagrammatically illustrate a flowchart showing the key components of one embodiment of the present invention, that is, an e-mail security system. FIG. 11A is linked to FIG. 11B via jump points 11-A and 11-B. The method of securing e-mail data operates on a distributed computer system which at least includes a remote memory designated as an extract store. Of course, the extract store may comprise a plurality of extract stores operative in conjunction with a plurality of security clearance levels. A singular security level is identified in FIG. 11A. Further, the e-mail may be subject to a parsing algorithm which, as discussed above, is generally independent of the identification of security sensitive data. However, with respect to the parsing aspect of the present invention, the original e-mail data is split into extracted data and remainder data and the extracted data is stored in an extract store. Hence, the parsing algorithm operates essentially independent of the content whereas the secured e-mail program operates based upon content identification. Although FIGS. 11A and 11B primarily relate to identification of security data, the same is true regarding the use of securing e-mail data with a parsing algorithm.

[179] The e-mail security system begins with step 602 wherein the system or program is turned ON or is activated. Step 603 recognizes that the user originating plaintext document 100 (not shown) has set a security filter identifying one or more security sensitive words, characters or icons. In step 604, the user composes the e-mail representative of plaintext document 100. In step 606, the user selects the "send" command in the typical e-mail program. As is customary, the system in step

Ex. 1002, Page 82

608 conducts a spell checking routine prior to sending the e-mail. In step 610, the system conducts a security check on the plaintext document or composed e-mail generated in step 604. The filter is used in step 604. In step 612, security words are highlighted or distinguished in the e-mail prior to the actual sending of the e-mail to the addressee. This step 612 is optional. In step 614, the user selects the security words for data to be extracted out. The highlighting step facilitates this selection. In step 616, the system extracts the security data and, preferably, in step 618, the security data is encrypted. Step 618 is optional. In a parsing application to secure e-mail, the parsing algorithm operates automatically at step 610 thereby eliminating steps 612 and 614. The extracting step 616 simply represents that the segmented data obtained from the original plaintext e-mail generated at step 604 is separated from remainder data.

[180] After encryption step 618, the e-mail security system generally operates in one of three manners. Other systems may be formulated based upon the systems and subsystems discussed herein. In one methodology, a second e-mail is created (see step 629), in a second methodology the secured data in encrypted form is attached or appended to the original e-mail containing remainder data (step 621) or, in a third methodology, the encrypted security data is simply added to or inserted into the end of the remainder data of the e-mail (step 623). The methodology of generating a second e-mail is initially discussed.

[181] A second e-mail having encrypted security data is created in step 620. Further, the system in step 622 adds a hyperlink to the remainder data in the original e-mail created in step 604. The hyperlink presents a pointer for the addressee to a secured application service provider or ASP. See the discussion of FIG. 2 above. The ASP represents a data storage facility for the secured e-mail data. In step 624, the remainder data from the original e-mail is sent to the addressee in a normal manner. This step also includes the concept that the second e-mail containing the encrypted security

Ex. 1002, Page 83

data is sent to the ASP. In step 626, the addressee receives the remainder e-mail which includes a hyperlink to the secured data ASP.

[182] The system jumps at jump step 11-A from FIG. 11-A to FIG. 11-B.

[183] In step 628, the addressee receives the remainder e-mail, visits the ASP via the hyperlink and clears the security levels at the secured ASP. In step 630, the secured data ASP obtains a map for each secured data e-mail (since the original e-mail may be broken up into a plurality of extracted, secured data e-mails) obtains all secured data e-mail and decrypts the same. In step 632, the secured ASP downloads the secured data as an e-mail to the addressee. In step 634, the addressee system compiles the original plaintext e-mail 100. A reconstruction program may be necessary to decode the secured data and insert the data into the document via the placeholders.

[184] Optionally, the decryption could occur at the recipient's e-mail device somewhat prior to the reconstitution of the e-mail plaintext document 100 during step 634. This requires the addressee to have the encryption routine and the correct key or decrypt code. The e-mail security system described above may include many of the features discussed earlier in connection with FIGS. 1-9. For example, both the security data and the remainder e-mail data can be encrypted prior to transmission to the addressee and the secured data ASP. The encryption may include multiple levels of encryption and decryption may require multiple levels of security clearance. The encryption may be mixed in the remainder e-mail. Partial as well as full reconstruction is enabled as discussed above in connection with FIG. 3.

[185] From the senders or originator's viewpoint, the e-mail facility described herein facilitates the storage of the extracted data at one or more secured sites.

[186] Another implementation of the secured e-mail system attaches the encrypted and secured data to the remainder e-mail data as indicated in step 621. E-mail attachments are well known.

Ex. 1002, Page 84

Alternatively, the encrypted secured data may be embedded or copied in encrypted form at the end of the remainder data in the original e-mail as indicated in step 623. In either case, in step 625, the e-mail is sent to the addressee. In step 627, the addressee opens the attachment. In step 629, the system of the recipient decrypts the secured data attachment or the embedded data attachment. In step 631, the recipient's system integrates the now decrypted secured data with the remainder data. Of course, this a compilation step. Place holders or other position indicators are customarily utilized. Appending the encrypted security data is generally equivalent to attaching a file to the original e-mail which constitutes, after extraction, the remainder data. Including the encrypted security data is adding the security data to the original e-mail at a predetermined location (either the top of the e-mail, the bottom of the e-mail or some predetermined line number).

[187] It should be appreciated that the e-mail security system may work automatically or may be selected manually by the user. The highlighting or special distinguishing manner for the security words in step 612 is optional. By highlighting the security words, the user may select or deselect those words for extraction. At the addressee's side, the addressee's system may be configured to automatically seek out the secured data ASP, enter security clearance data, download the secure data and integrate the secure data in the remainder data e-mail. The present invention contemplates automatic as well as manual steps in steps 626, 628, 630, 632 and 634. The hyperlink with the original remainder e-mail essentially maps the remainder data to the secured data and the remote storage locations handling the secure data. Multiple security clearances may be required of the recipient or addressee. The e-mail system can be combined with other features of the security system discussed above such as multiple security data locations, secret key sharing schemes, multiple encryption of the data in a single document, multiple security clearance levels required for a plurality of storage facilities, the two man key system, automation of key management and a

Ex. 1002, Page 85

plurality of levels of access to the data such as partial reconstruction in step 634 and full reconstruction.

[188] FIGS. 12A and 12B diagrammatically illustrate a flowchart showing the key components of one embodiment of the system and the invention which implements the security system on a web browser. Jump point 12-A links FIG. 12A to FIG. 12B. The system, at step 700 is ON. The filters establishing either the parsing or the identification of security data are established in the filter set step 701. In step 702, the user inputs data into open field of an HTML display page which the user has previously downloaded from a web server. In step 704, the user may select "secure now" turning ON the system or the system may automatically be ON such that the filter is screening all the data input by the user in the open field. In step 706, the system scans all the open field data, locates security data and extracts security data. In step 708, place holders are added to replace the extracted security data in the remainder data and a hyperlink is added to the open field remainder data providing a link to the secure data ASP. In step 710, the user selects the "send button" or any other indicator on the HTML page triggering an operation which transmits the open field data (which is now remainder data) to the web server. In step 712, the web server and particularly the common gateway interface (CGI) receives the remainder data fields, identifies the place holders in the data and the hyperlink to the secure data ASP. In step 714, the web server receiving the data from user's browser goes to the secure data ASP, inputs and clears any security level, and obtains the secured data. In step 716, the web server reconstructs the open field data which generally is represented by plaintext document 100. In step 718, the web server processes the data as necessary. Many of the features discussed above in connection with FIGS. 1A-11A may be implemented on the browser system.

Ex. 1002, Page 86

[189] The credit card scrubber or financial data scrubber operates in a similar manner to the email and browser data security system described above. The credit card or financial data scrubber (herein collectively "CC scrubber") typically operates on a defined sequence of numbers. For example, if a credit card number is 17 digits, whenever the email or browser security system or program detects17 sequential numerical digits (a pre-set filter), a pop-up window may appear enabling the user to select or turn ON the scrubber. If ON, the data security program strips or parses the credit card number and sends, for example, five of the 17 digits to a secure store. Placeholders or substitute characters may be inserted into the remainder CC data. To reconstitute the entire CC data, the intended recipient would be required to pass security clearance levels at the secure store. Of course, the CC scrubber could be set to detect bank account numbers, personal or business account holder names, pre-set passwords, etc. In an OFF state, the CC scrubber would let pass the CC number, account number or pre-set data stream or string. The user may select (i) always ON; (ii) pop-up window, select ON or OFF per transaction; (iii) pop-up window to select OFF (default being ON); or (iv) always OFF but minor reminder (audible sound, icon appearance, etc.) of data security risk. The CC scrubber may encrypt the extracted data for security. Other visual ques may rather than a pop-up window may be used (for example, a drop down menu).

[190] FIG. 13 diagrammatically shows several revenue systems which may be employed with the data security systems described herein. Many types of revenue systems may be employed in conjunction with the present invention. FIG. 13 shows two basic systems, one at the data input stage and the second at the data output or reconstruction phase. Within each revenue subsystem are two types of revenue generators, an advertising revenue generator and a user charge generator. The user charge system contemplates charging or assessing a fee to the user's employer or organization. Therefore, the system operator may select up to four (4) revenue generation systems (ads at the input,

Ex. 1002, Page 87

charges at the input, ads at the output and charges at the output). It is well known that vendors selling goods and services over the Internet are willing to pay a certain percentage of their sales revenue to other entities referring customers to the vendor's web sites. The concept of display ads in FIG. 13 includes this revenue stream. The system operator may choose all, one, several or none of these revenue systems to be deployed in conjunction with the data security system described earlier herein. Other revenue system may also be utilized. The steps in the revenue system described herein may be reorganized to attain higher consumer and user acceptance and/or to maximize the revenue to the system operator.

[191] Decision step 730 determines whether the system is deployed at the data input phase or not. It is clear that the system operator may utilize the data reconstruction revenue system and hence the decision step 730 is not necessary. If the data input system is employed, step 732 displays the ad to the user. The user may be uploading a complete document to an application server on the Internet or may be using a application service provider on the Internet or an private LAN to secure his or her data. The display ad 732 step enables the user to click on the ad and visit the vendor, thereby potentially generating a referral fee. See referral fee branch 757. Step 734 requires password clearance. Step 736 processes the document or data object with the security system. The user may input the document real time or input it to the application server or may upload the complete document to the server. Alternatively, the ad could be buried in the email or application program run on the user's computer and the user would be shown an ad and given a link to the vendor's Internet site. Selecting the link points the user's browser to the vendor's site.

[192] Step 738 shows display ad 2 to the user thereby potentially generating referral revenue for the system operator. Step 740 notes that the user exits the revenue system. Step 742 determines whether the system charges the user for the security service. If YES, the program processes the

Ex. 1002, Page 88

charge in step 745 (charge systems are known). If NO, the system ends or returns to other programs in step 747.

[193] The NO branch from determination step 730 leads to the receipt of a reconstruction request by the user in step 750. Step 752 determines whether the user will be charged. If YES, the system executes step 745. If NO, the system displays the ad 1 in step 754. Referral generation is noted by branch 757 from step 754. In step 756, the user's password is subject to clearance. In step 758, the user's request is processed, the document or data object is reconstructed (fully or partially as described earlier), and in step 759 the system displays ad 2. In step 762, the user's activity is logged in to the system. Step 764 determines whether the charge to the user is reduced (because he or she viewed the ads) and if not, the system ends in step 747, if YES, the system processes the charge in step 745. Alternatively, the user may be shown display ads and/or charged for services upon storage of extracted data. Step 750 includes this concept.

[194] The claims appended hereto are meant to cover modifications and changes within the scope and spirit of the present invention.

Ex. 1002, Page 89

What is claimed is:

1. A method of securing data having one or more security sensitive words, characters or icons in a computer system with memories designated as a remainder store and an extract store comprising:

extracting said security sensitive words, characters or icons from said data to obtain extracted data and remainder data therefrom;

storing said extracted data and said remainder data in said extract store and said remainder store, respectively; and,

permitting reconstruction of said data via said extracted data and remainder data only in the presence of a predetermined security clearance.

2. A method as claimed in claim 1 including filtering said data while extracting.

3. A method as claimed in claim 2 including defining the filter with a group of security sensitive words, characters or icons and the method includes one of storing said filter or destroying said filter after storing said extracted data.

4. A method as claimed in claim 3 including encrypting one or both of said extracted data and said remainder data.

5. A method as claimed in claim 4 wherein the step of permitting reconstruction includes decrypting one or both of said extracted data and remainder data.

6. A method as claimed in claim 1 including establishing a plurality of security levels each with a respective security clearance, said security sensitive words, characters or icons being correlated with said plurality of security levels with subsets of said security sensitive words, characters or icons and the step of permitting reconstruction including the step of permitting a

Ex. 1002, Page 90

plurality of partial reconstructions of said data in the presence of respective ones of said plurality of security clearance levels.

7. A method of securing data with a parsing algorithm via a computer system having memories designated as a remainder store and an extract store comprising:

parsing said data to obtain extracted data and remainder data therefrom;

storing said extracted data and said remainder data in said extract store and said remainder store, respectively; and,

permitting reconstruction of said data via said extracted data and remainder data only in the presence of a predetermined security clearance.

8. A method as claimed in claim 7 wherein the step of parsing parses the data granularly.

9. A method as claimed in claim 7 wherein the step of parsing uses said parsing algorithm which is based on one or more of a bit count, a word, a word count, a page count, a line count, a paragraph count, an identifiable document characteristic, an identifiable word characteristic, an identifiable icon characteristic, an identifiable data object characteristic, capital letters, italics, and underline.

10. A method as claimed in claim 7 wherein said data is financial data and the step of parsing uses said parsing algorithm based upon one of characters in said data and character count in said data.

11. A method as claimed in claim 7 including establishing a plurality of security levels each with a respective security clearance, increasingly greater reconstruction if said data being correlated with increasingly higher levels of said plurality of security levels, and the step of permitting reconstruction including the step of permitting a plurality of partial reconstructions of said data in the presence of respective ones of said plurality of security clearance levels.

Ex. 1002, Page 91

12. A method as claimed in claim 7 including a further data security step of one from the group of storing said data without encryption, storing said data with encryption and destroying said data; said further data security step being independent of said parsing, storing and permitting reconstruction.

13. A method as claimed in claim 1 including displaying an advertisement prior to one or the other or both of said steps of extracting and permitting reconstruction.

14. A method as claimed in claim 13 including providing a hyperlink to a vendor's web site with said advertisement.

15. A method as claimed in claim 1 including assessing a charge for extracting, storing or permitting reconstruction.

16. A computer readable medium containing programming instructions for securing data having one or more security sensitive words, characters or icons in a computer system with memories designated as a remainder store and an extract store comprising:

extracting said security sensitive words, characters or icons from said data to obtain extracted data and remainder data therefrom;

storing said extracted data and said remainder data in said extract store and said remainder store, respectively; and,

permitting reconstruction of said data via said extracted data and remainder data only in the presence of a predetermined security clearance.

17. A medium with programming instructions as claimed in claim 16 including filtering said data while extracting.

Ex. 1002, Page 92

18. A medium with programming instructions as claimed in claim 16 including defining the filter with a group of security sensitive words, characters or icons and the method includes one of storing said filter or destroying said filter after storing said extracted data.

19. A medium with programming instructions as claimed in claim 18 including encrypting one or both of said extracted data and said remainder data.

20. A medium with programming instructions as claimed in claim 19 wherein the step of permitting reconstruction includes decrypting one or both of said extracted data and remainder data.

21. A medium with programming instructions as claimed in claim 16 including establishing a plurality of security levels each with a respective security clearance, said security sensitive words, characters or icons being correlated with said plurality of security levels with subsets of said security sensitive words, characters or icons and the step of permitting reconstruction including the step of permitting a plurality of partial reconstructions of said data in the presence of respective ones of said plurality of security clearance levels.

22. A computer readable medium containing programming instructions for securing data with a parsing algorithm via a computer system having memories designated as a remainder store and an extract store comprising:

parsing said data to obtain extracted data and remainder data therefrom;

storing said extracted data and said remainder data in said extract store and said remainder store, respectively; and,

permitting reconstruction of said data via said extracted data and remainder data only in the presence of a predetermined security clearance.

Ex. 1002, Page 93

23. A medium with programming instructions as claimed in claim 22 wherein the step of parsing parses the data granularly.

24. A medium with programming instructions as claimed in claim 22 wherein the step of parsing uses said parsing algorithm which is based on one or more of a bit count, a word, a word count, a page count, a line count, a paragraph count, an identifiable document characteristic, an identifiable word characteristic, an identifiable icon characteristic, an identifiable data object characteristic, capital letters, italics, and underline.

25. A medium with programming instructions as claimed in claim 22 wherein said data is financial data and the step of parsing uses said parsing algorithm based upon one of characters in said data and character count in said data.

26. A medium with programming instructions as claimed in claim 22 including establishing a plurality of security levels each with a respective security clearance, increasingly greater reconstruction if said data being correlated with increasingly higher levels of said plurality of security levels, and the step of permitting reconstruction including the step of permitting a plurality of partial reconstructions of said data in the presence of respective ones of said plurality of security clearance levels.

27. A medium as claimed in claim 22 including a further data security step of one from the group of storing said data without encryption, storing said data with encryption and destroying said data; said further data security step being independent of said parsing, storing and permitting reconstruction.

Ex. 1002, Page 94

ABSTRACT

A data security having parsing and dispersion aspects enables the user to parse, disperse and reconstruct the original, plain text data or data object, thereby enabling secure storage of the data. The original data may be maintained in its original state, encrypted or it may be destroyed. For example, financial data maintained by an institute, stored as is customary, be parsed with an algorithm, the parsed segments dispersed off-site (that is, separated and stored in extract and remainder stores or computer memories) and away from the financial institute, and, upon appropriate security clearance, the dispersed data can be reconstructed to duplicate the data. Large distribution of parsed data is contemplated by the system. The original data remains stable, operable and immediately useful in its customary storage location (or alternatively destroyed). The secured dispersed data is a back-up of the original data.

minor\6851-2-cip-3-pa.wpd

Case 6:21-cv-00569-ADA Document 1-2 Filed 05/25/21 Page 95 of 97

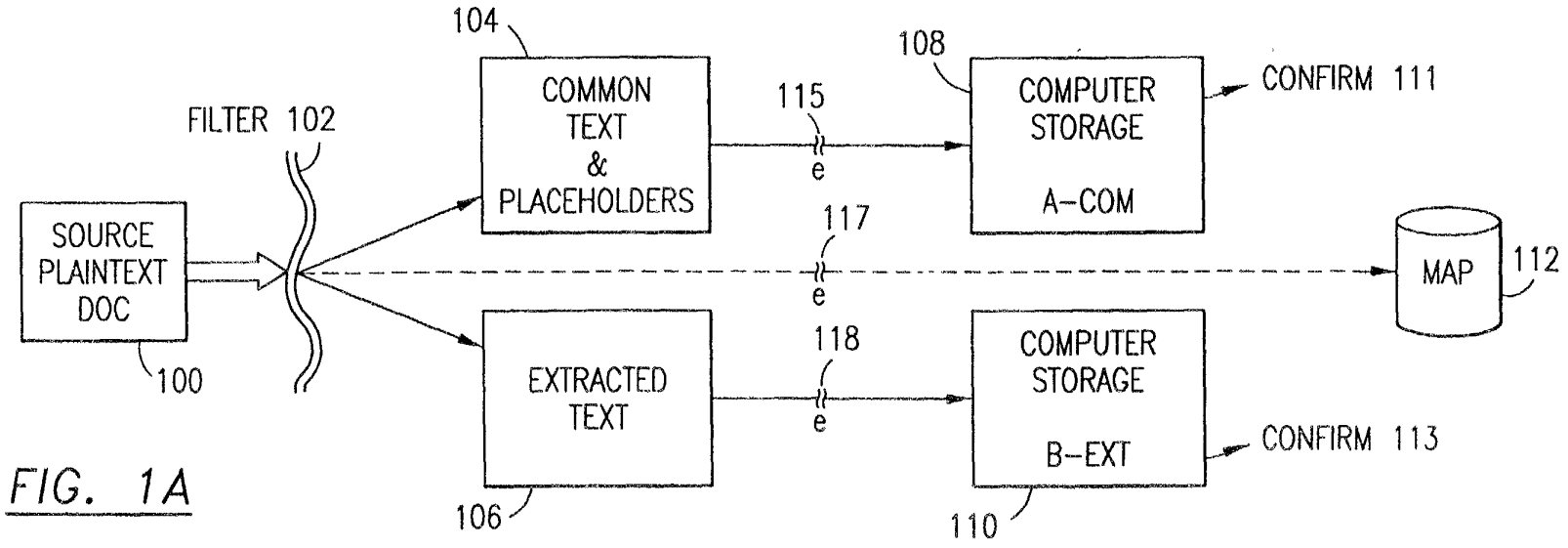

104
COMMON TEXT & PLACEHOLDERS
108
COMPUTER STORAGE A-COM
CONFIRM 111
115
e
FILTER 102
SOURCE PLAINTEXT DOC
100
117
e
MAP
112
EXTRACTED TEXT
118
e
COMPUTER STORAGE B-EXT
CONFIRM 113
106
110

FIG. 1A

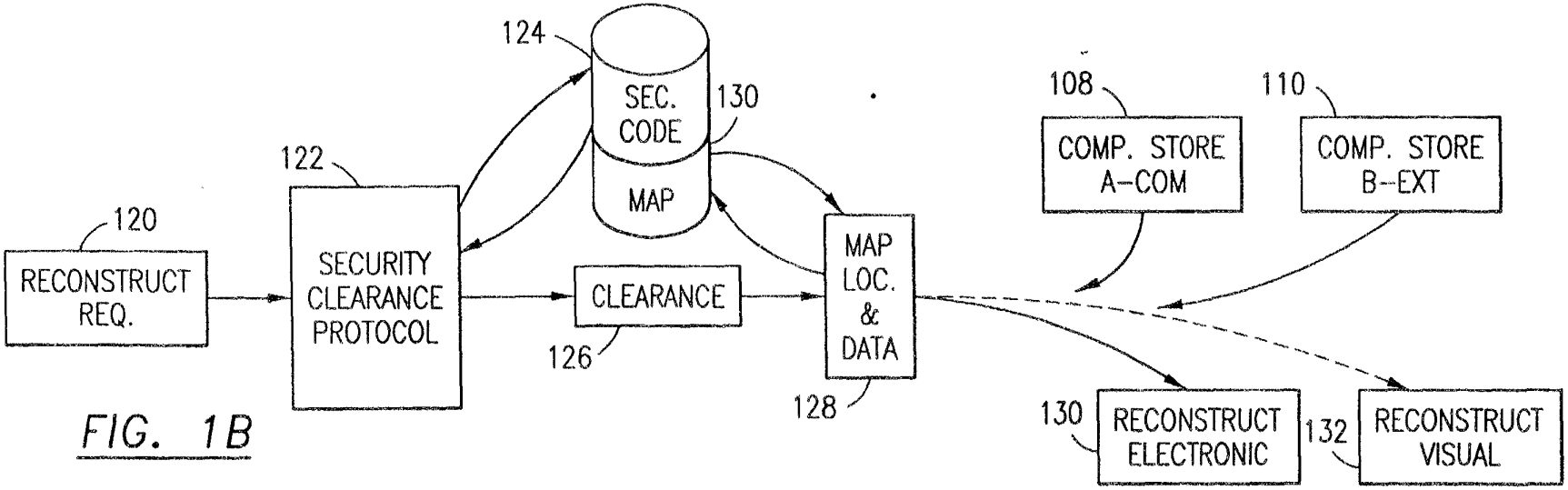

124
SEC. CODE
130
MAP
122
120
RECONSTRUCT REQ.
SECURITY CLEARANCE PROTOCOL
CLEARANCE
126
MAP LOC. & DATA
128
108
COMP. STORE A-COM
110
COMP. STORE B-EXT
130
RECONSTRUCT ELECTRONIC
132
RECONSTRUCT VISUAL

FIG. 1B

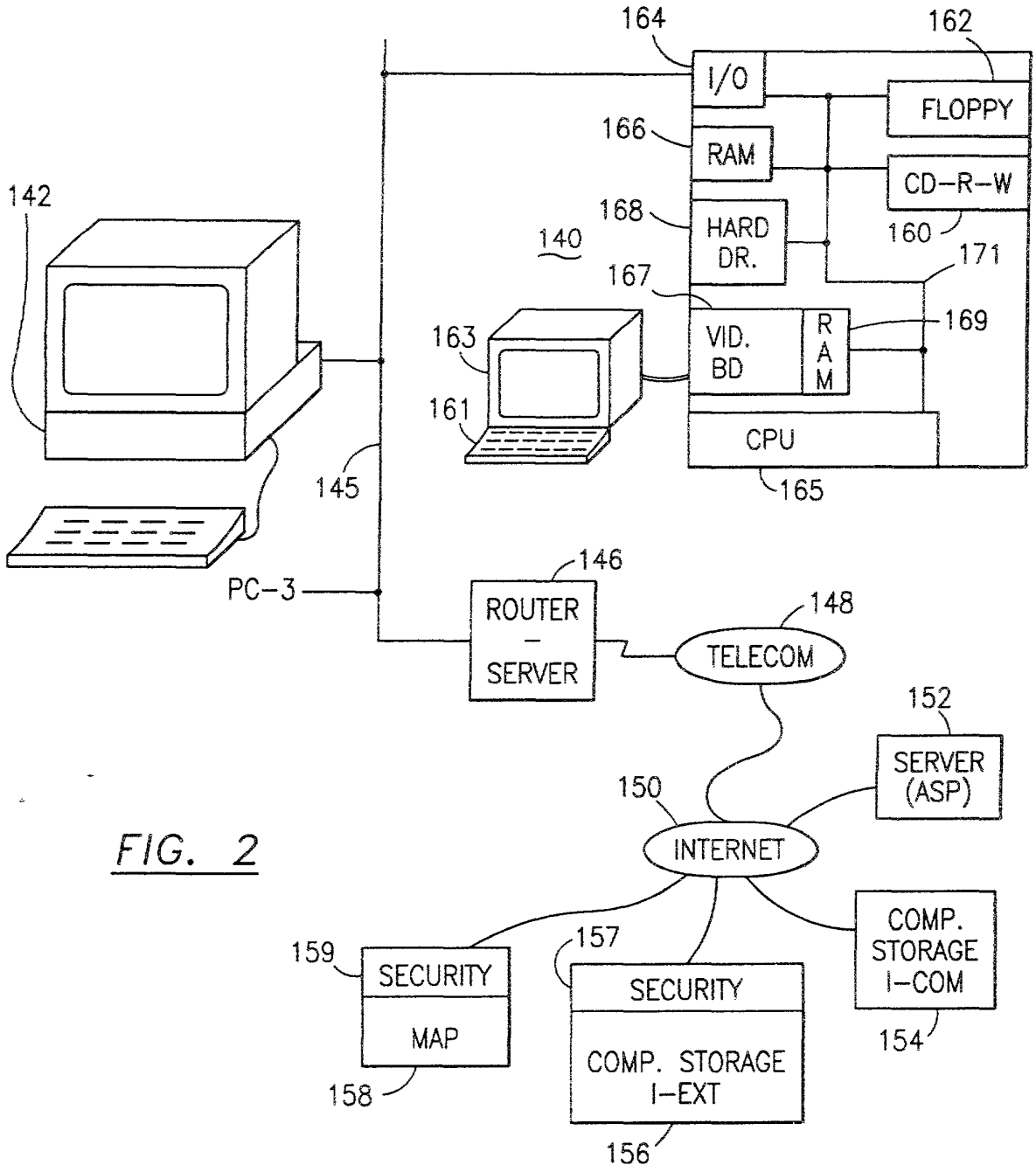

164
162
I/O
FLOPPY
166
RAM
CD-R-W
142
168
HARD DR.
160
140
171
167
163
VID. BD
R A M
169
161
CPU
145
165
PC-3
146
ROUTER – SERVER
148
TELECOM
152
SERVER (ASP)
150
INTERNET
159
SECURITY
MAP
158
157
SECURITY
COMP. STORAGE I-EXT
156
COMP. STORAGE I-COM
154

FIG. 2

20030050-061504T01

FIG. 3

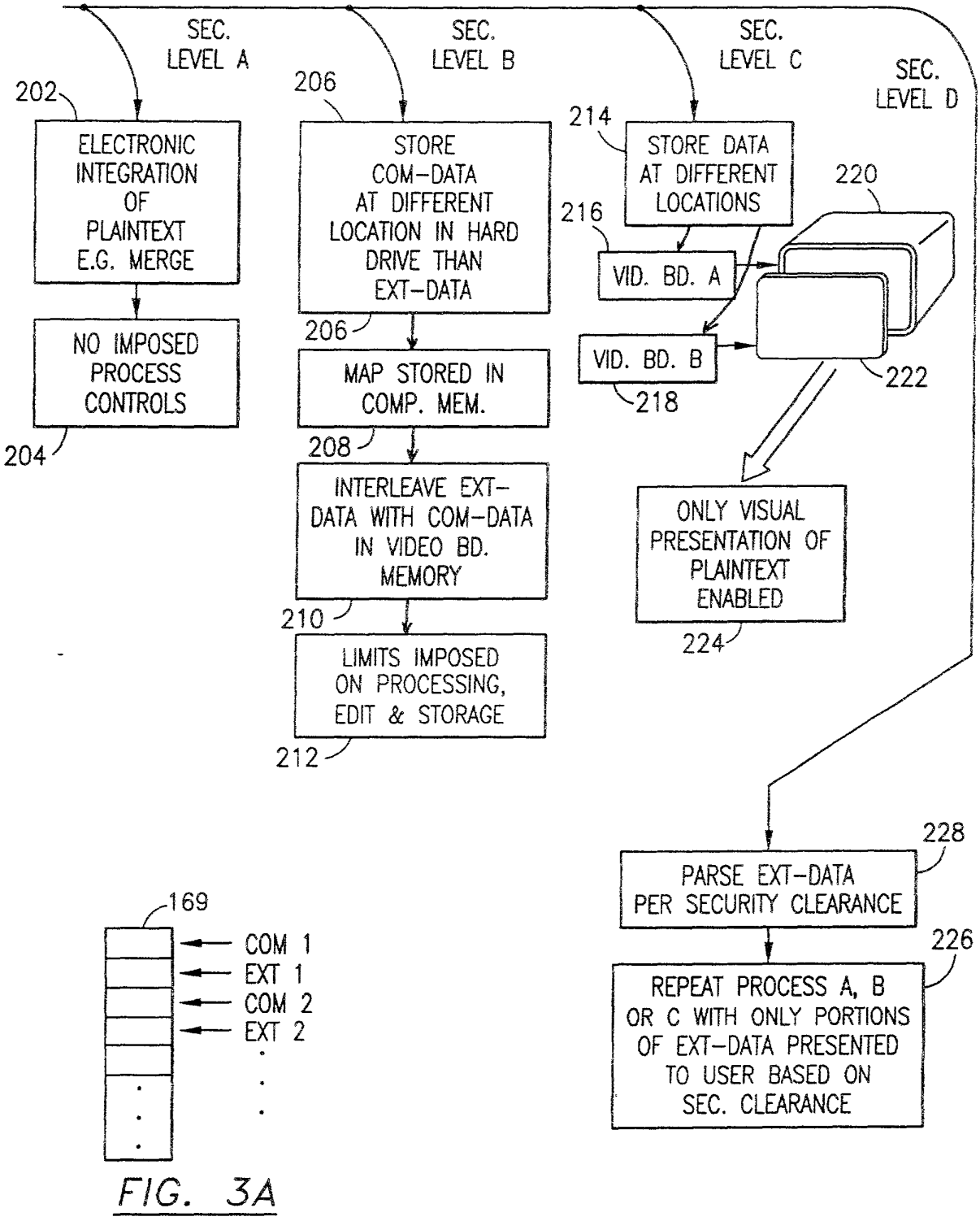
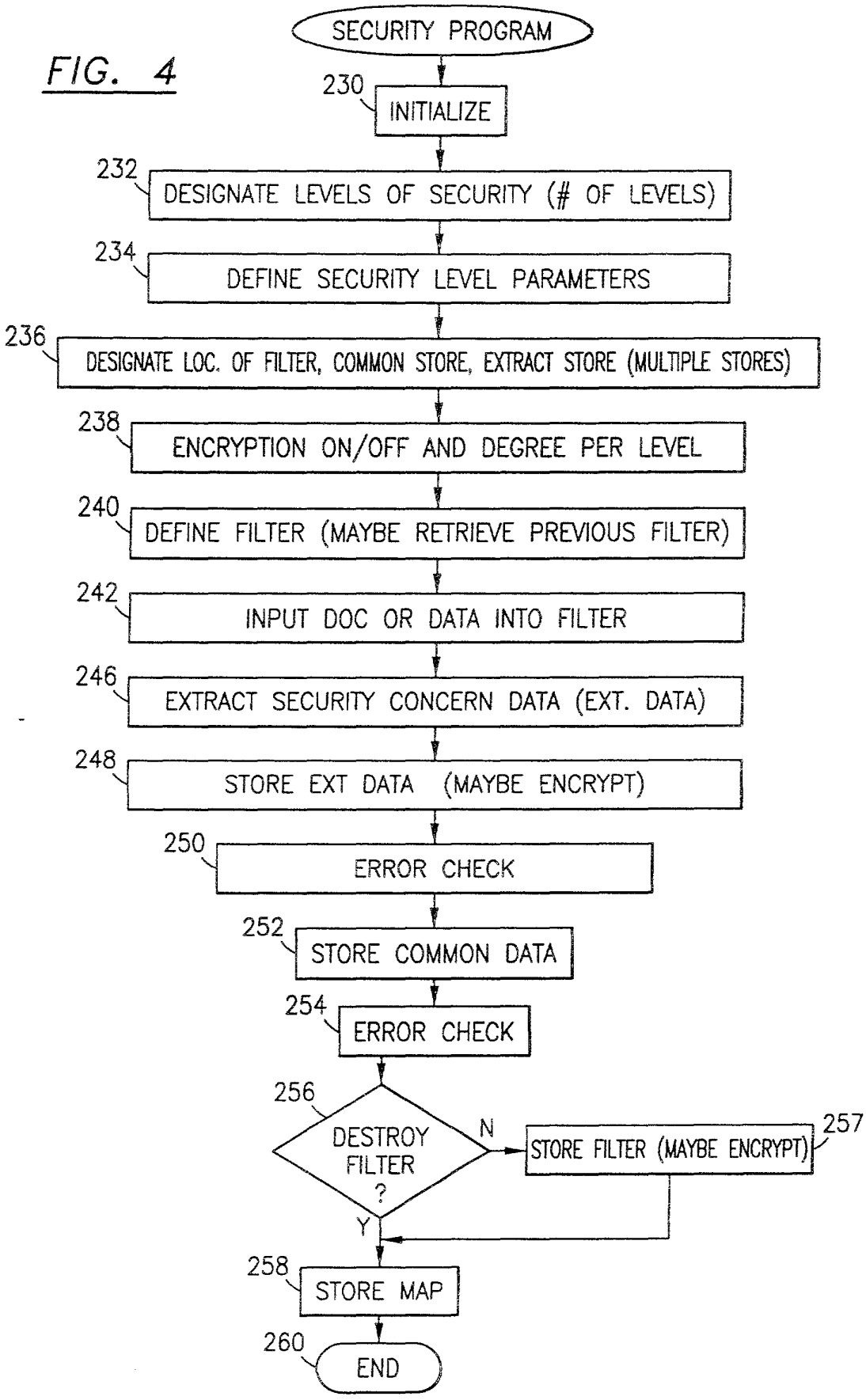


FIG. 3A

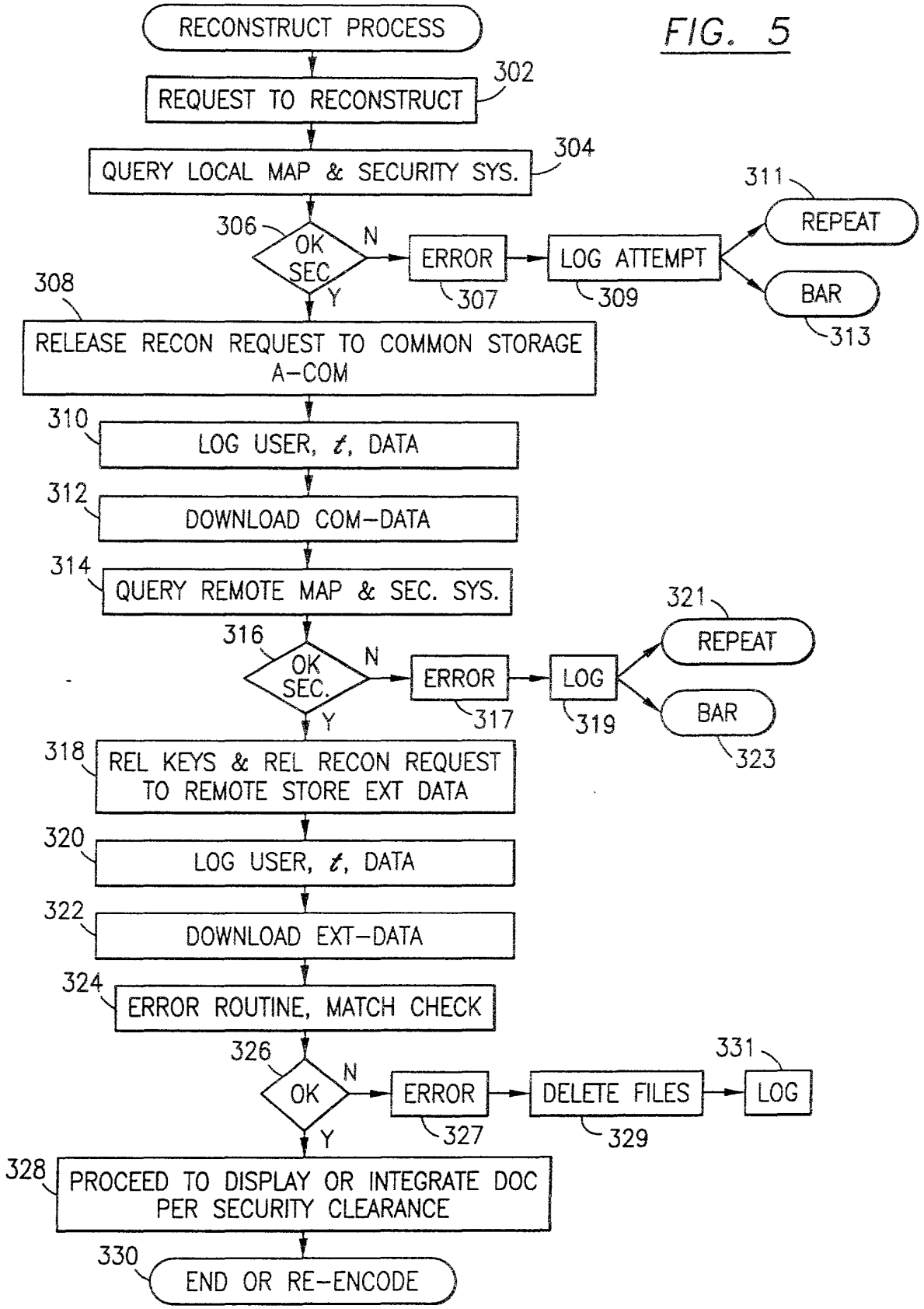
FIG. 4



Ex. 1002, Page 99

EXHIBIT 1002 - PAGE 100 OF 191

FIG. 5



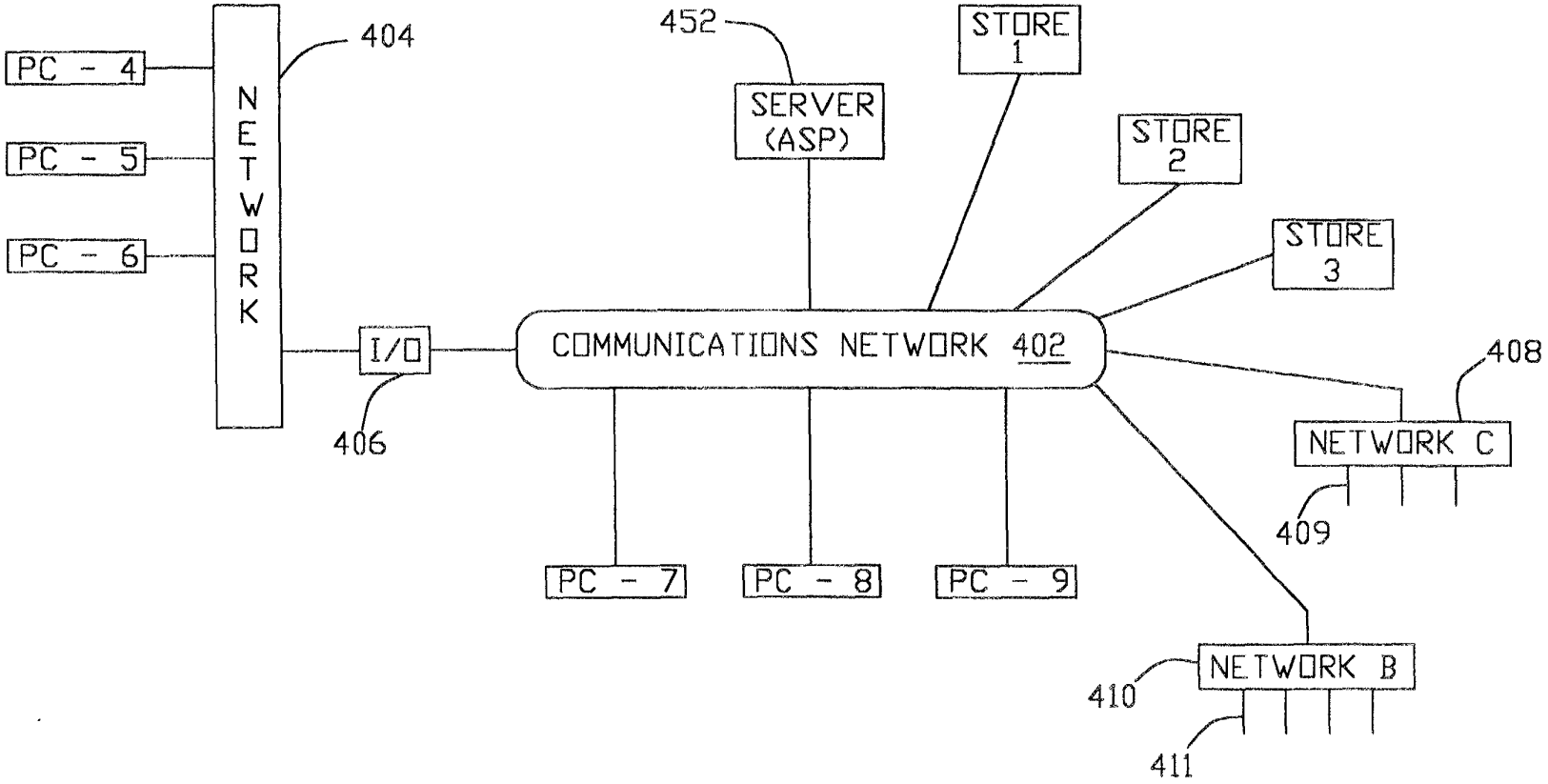
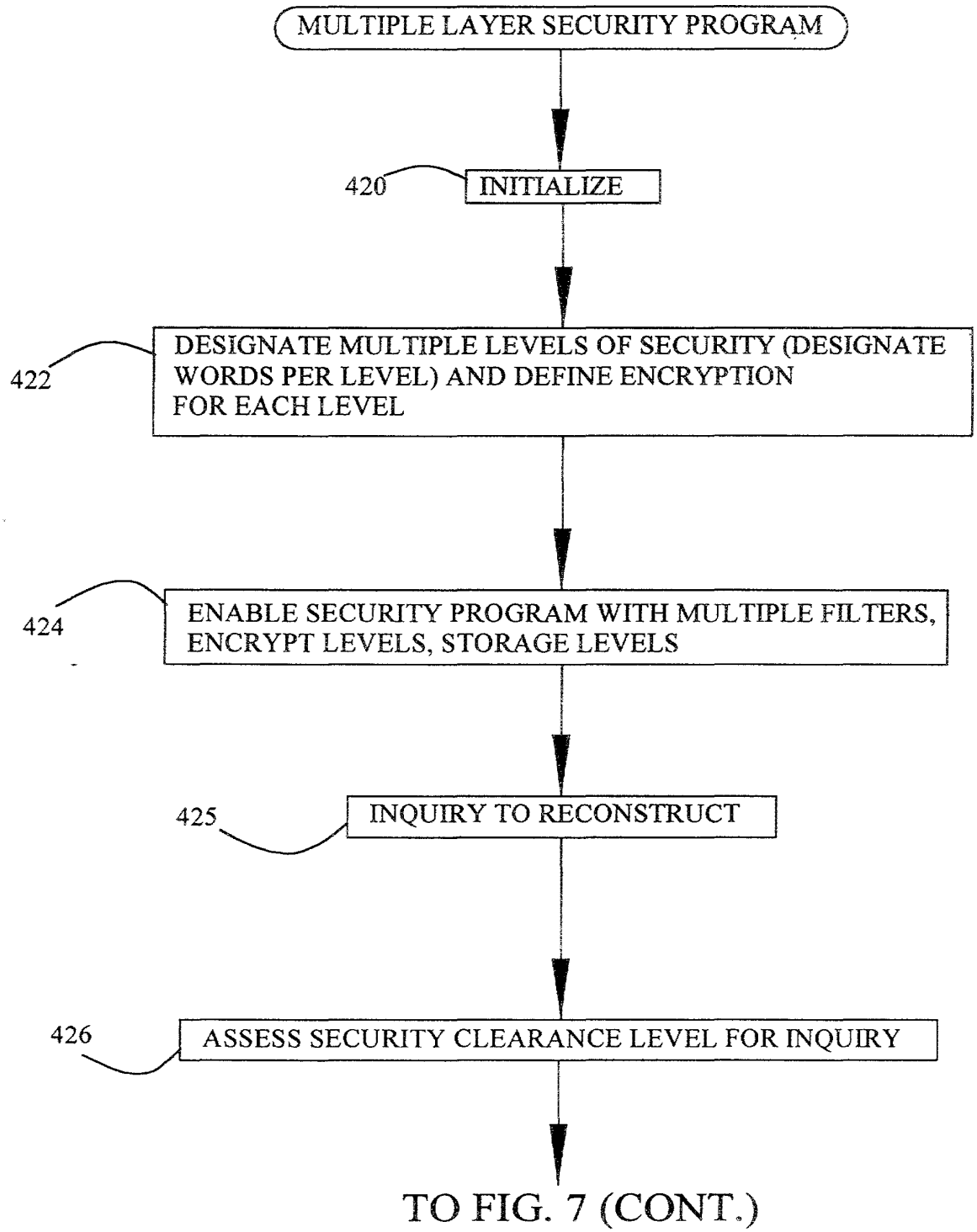

PC - 4
PC - 5
PC - 6
NETWORK
404
I/O
406
452
SERVER (ASP)
STORE 1
STORE 2
STORE 3
COMMUNICATIONS NETWORK 402
408
NETWORK C
409
PC - 7
PC - 8
PC - 9
410
NETWORK B
411

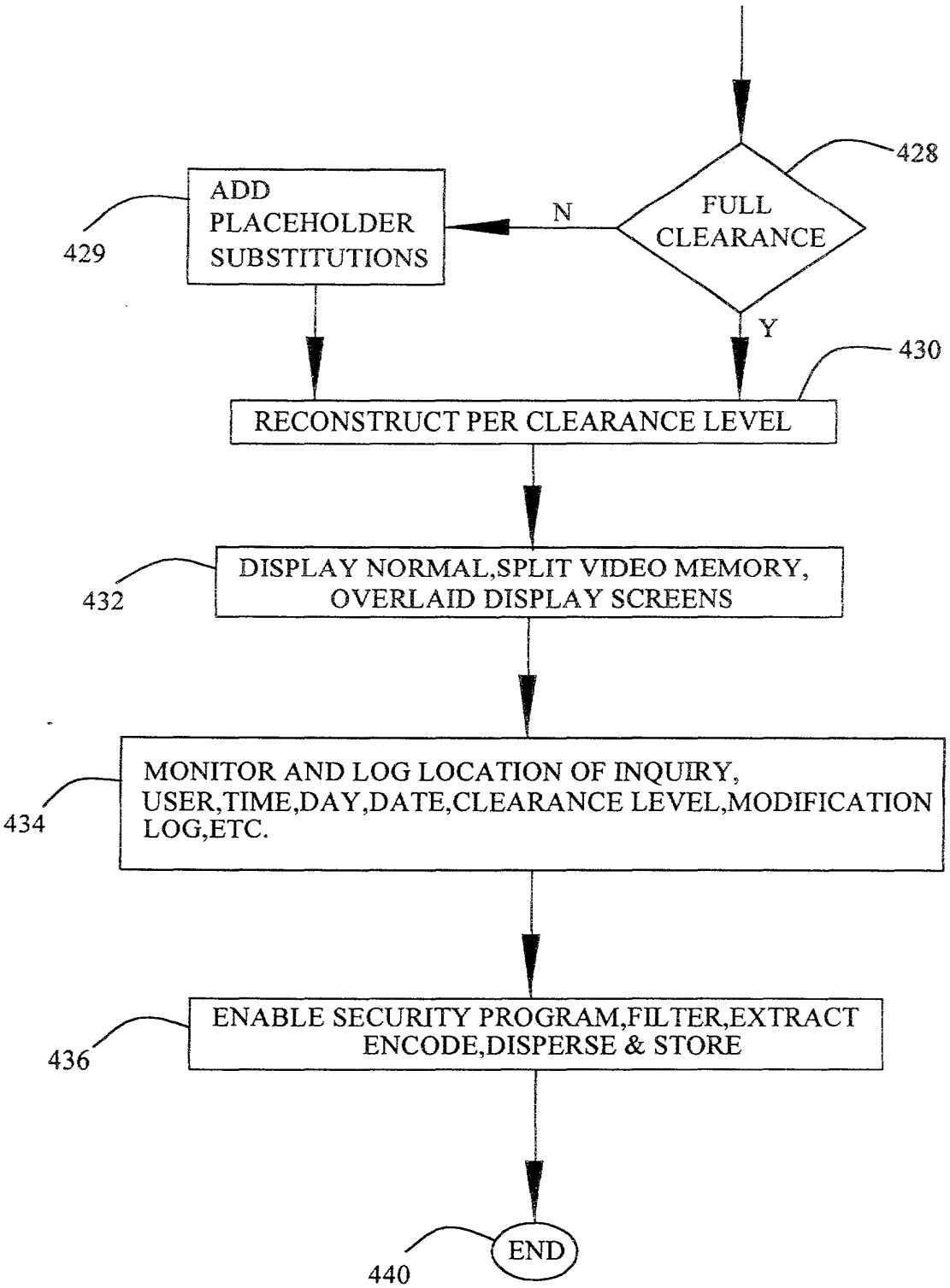
FIG.6

FIG. 7



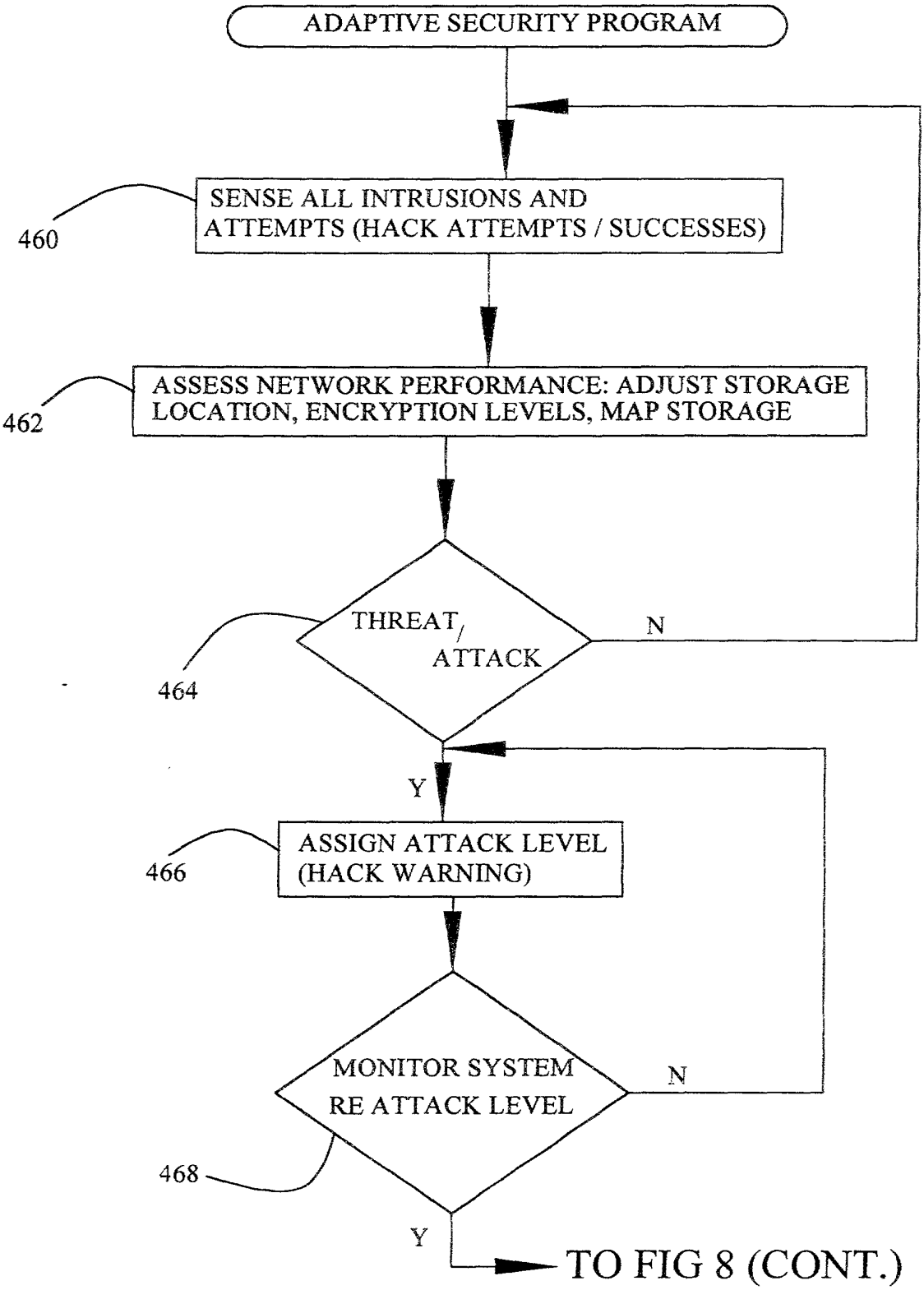
10155.1ge 09/22/2012

FIG. 7 (CONT.)



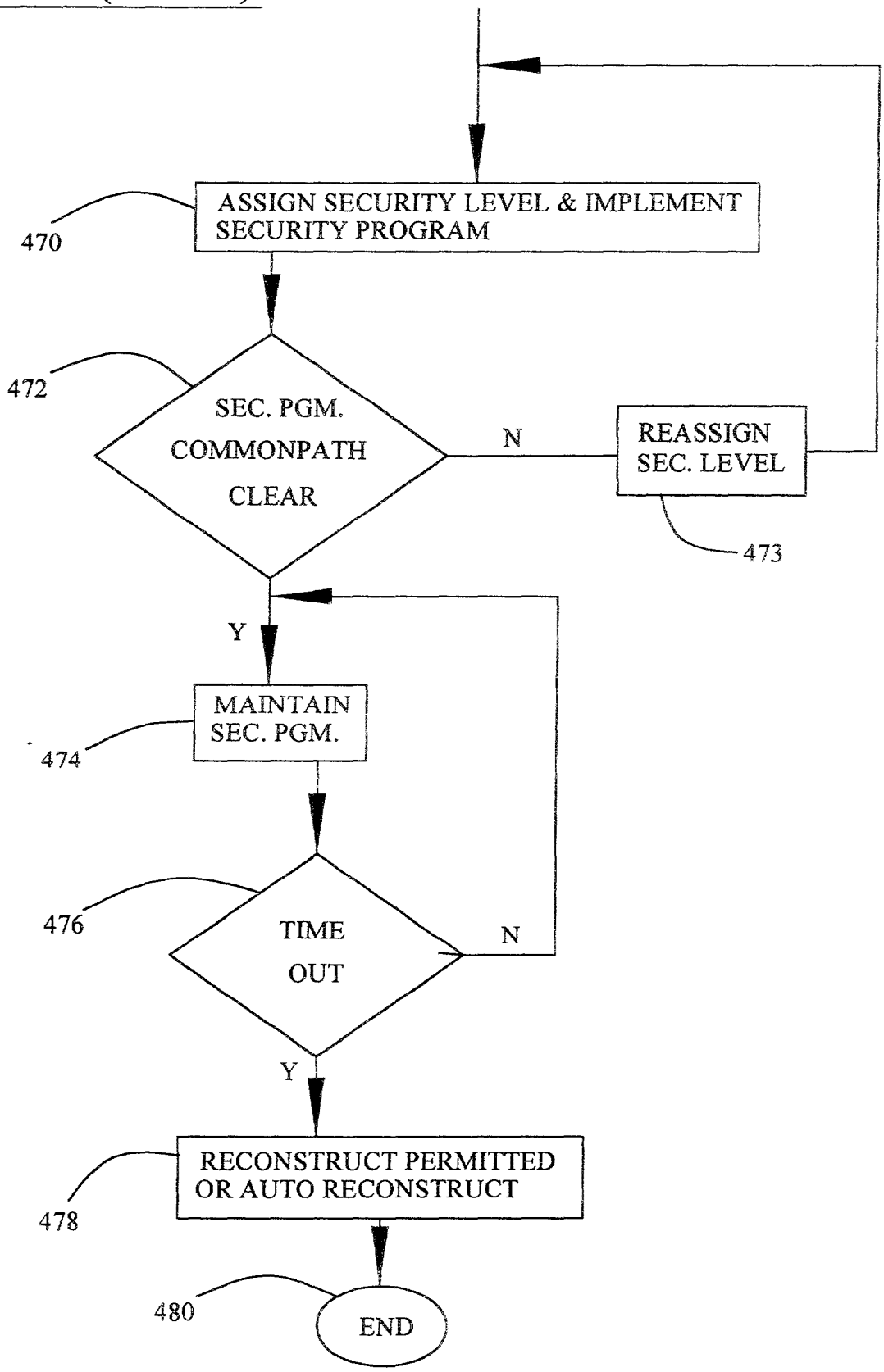
20637560 26 JUL 05 10:05

FIG. 8



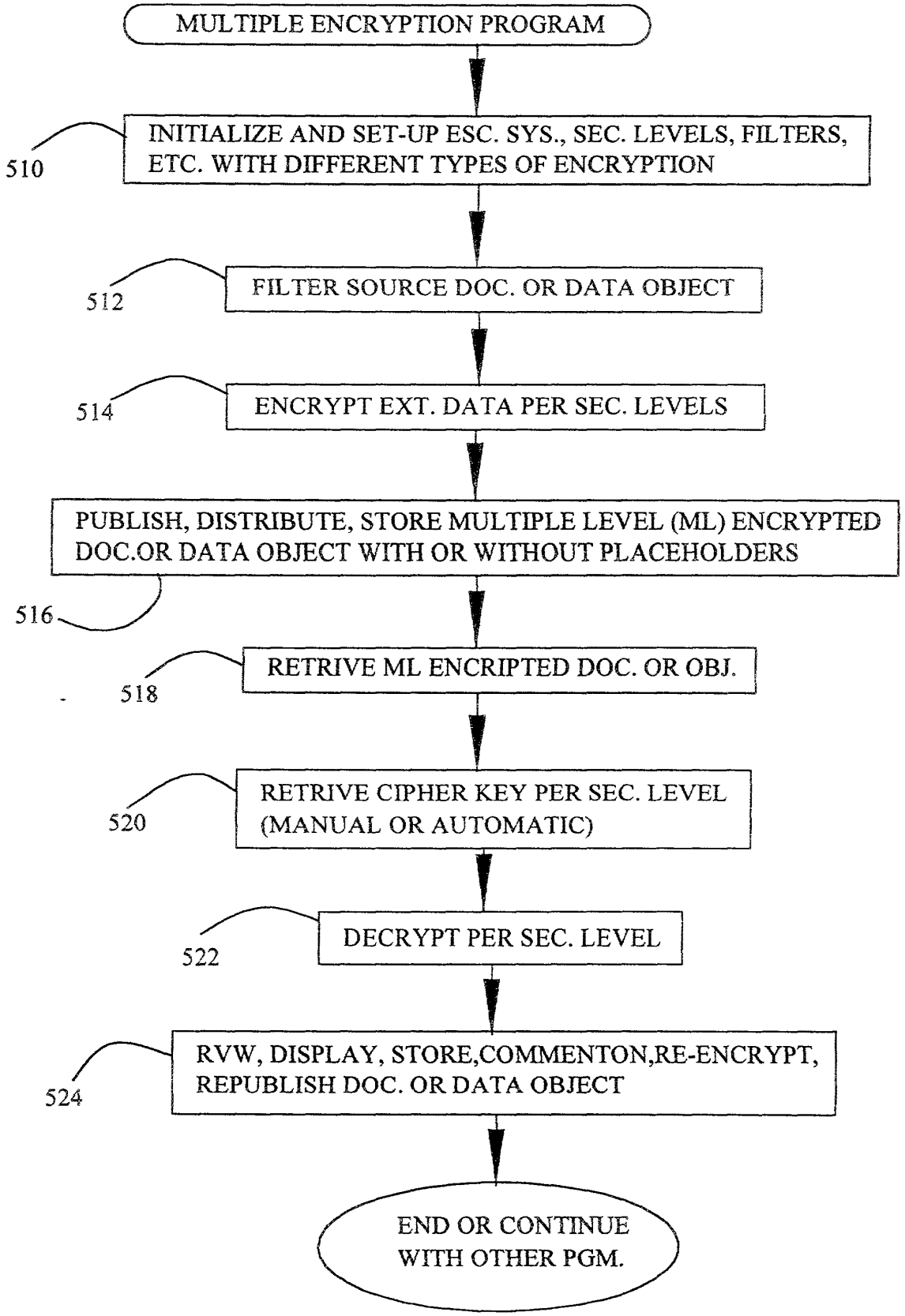
Ex. 1002, Page 104

FIG. 8 (CONT.)



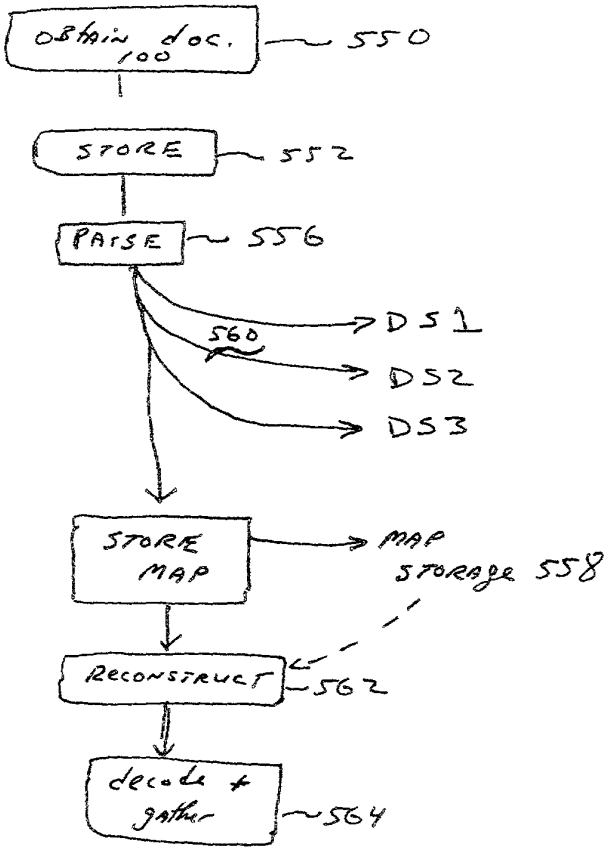
Ex. 1002, Page 105

FIG. 9



20020150276+A1.tif

FIG. 10



10155192 - 055302

20020059751

FIG. 11A

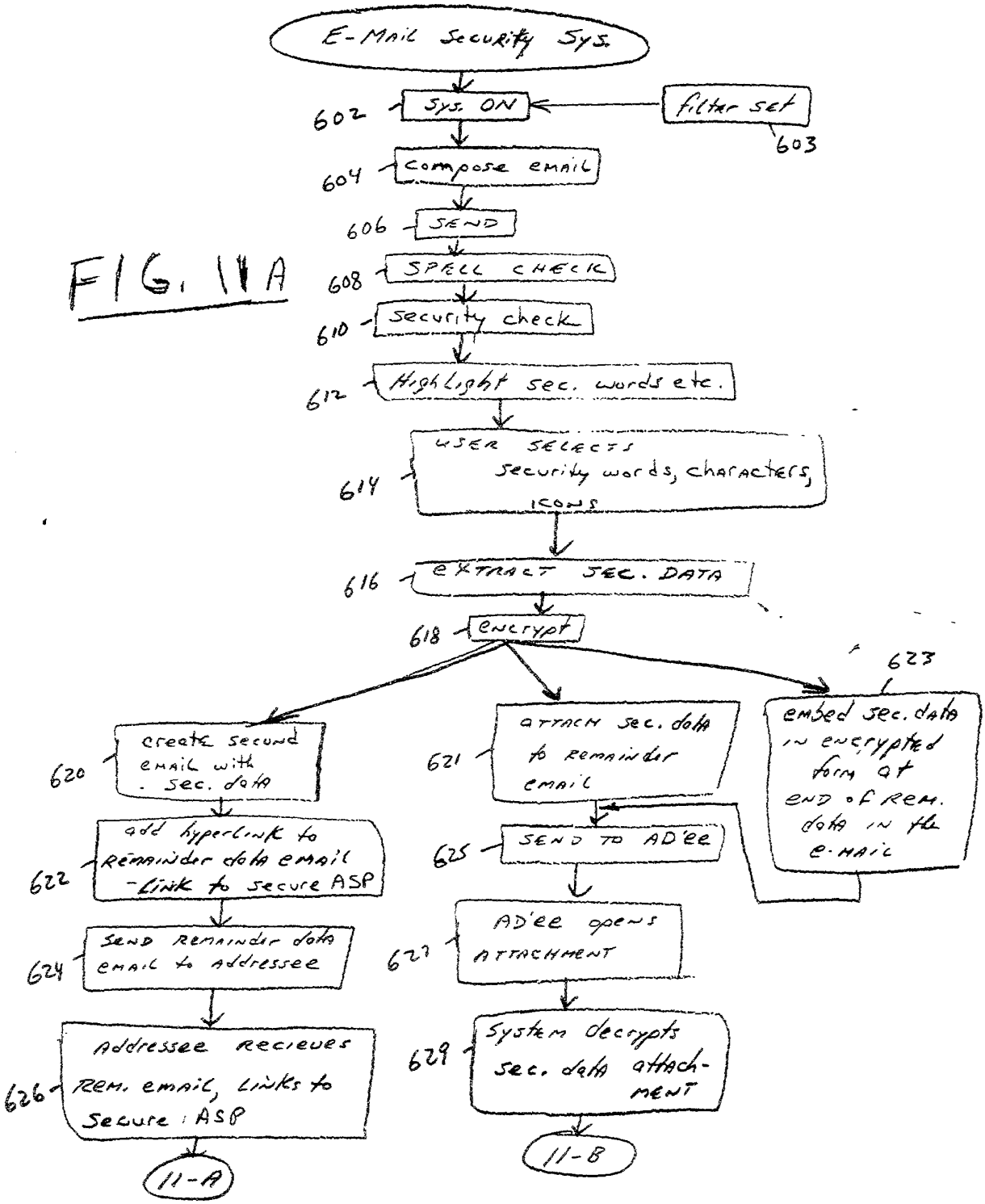
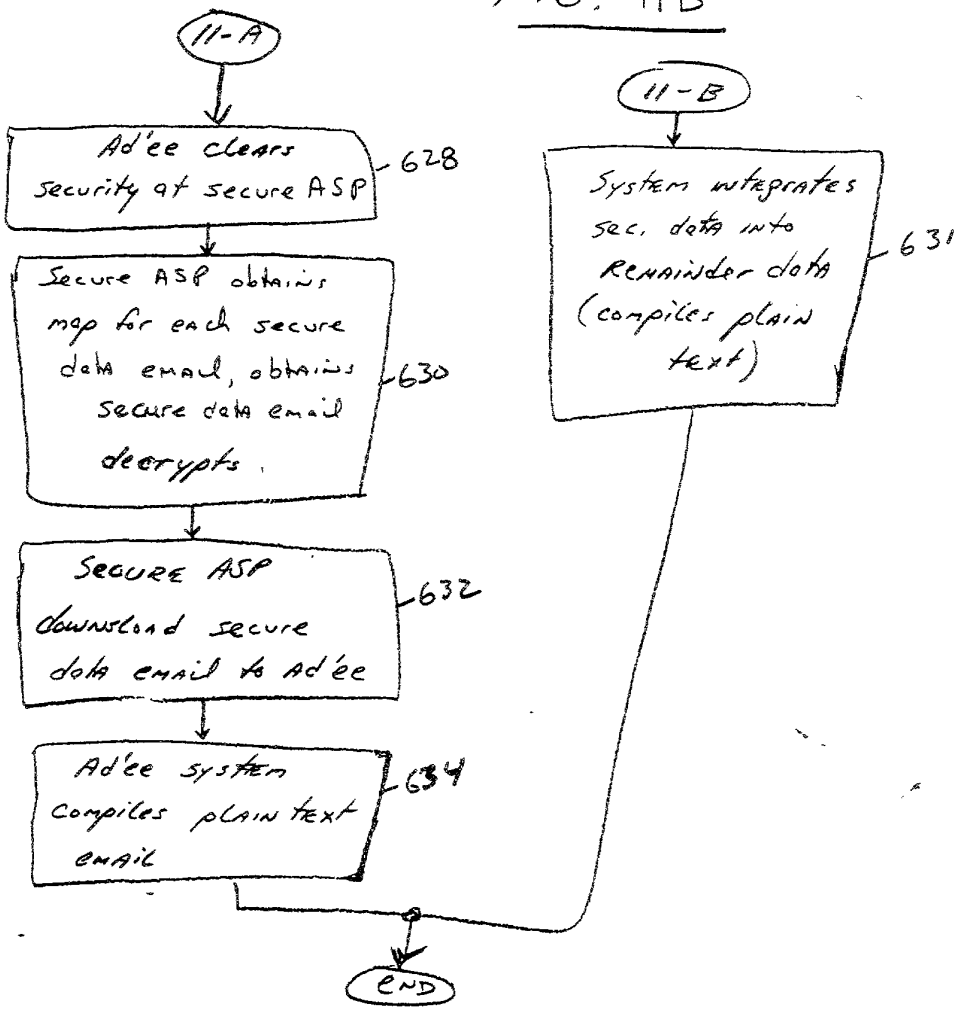
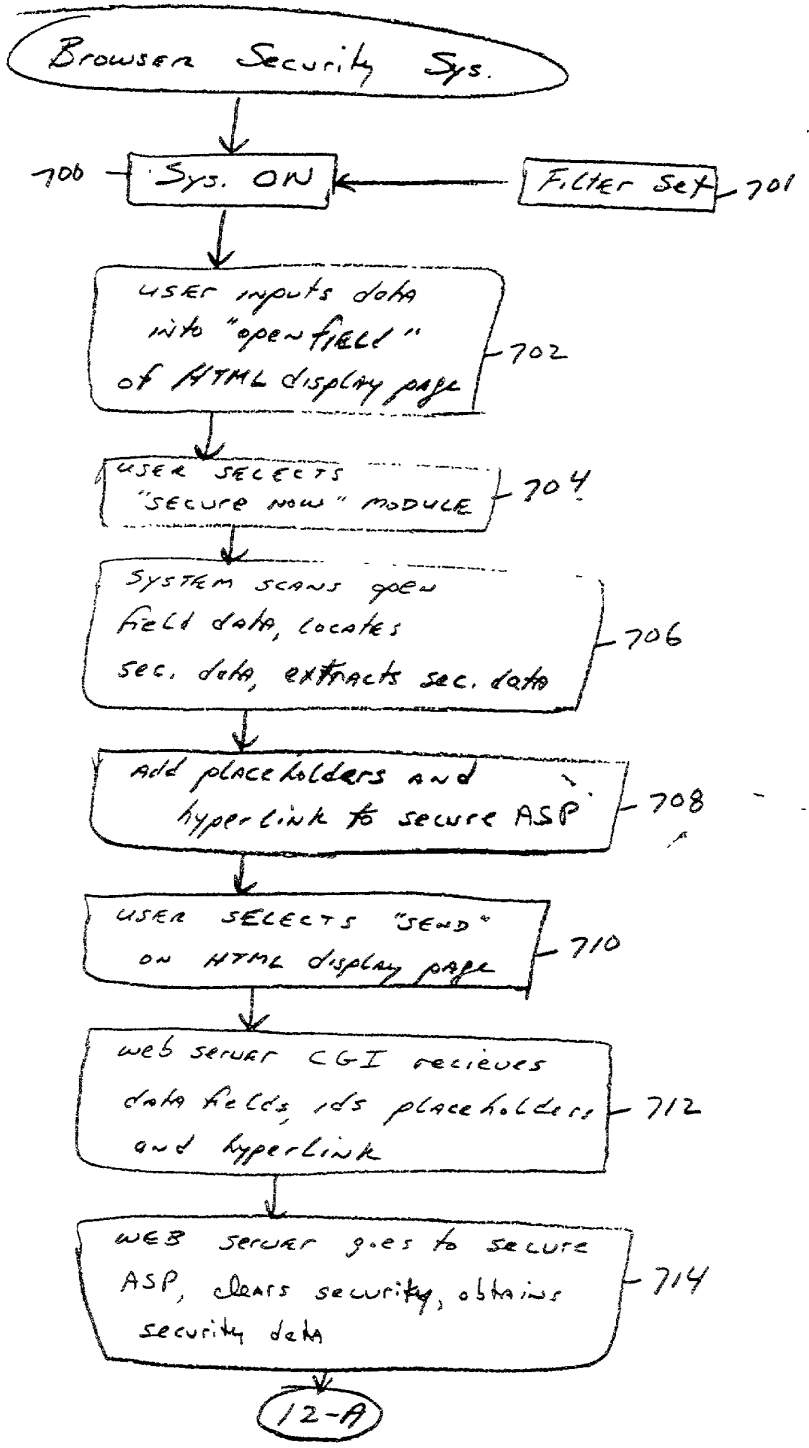


FIG. 11B



09-26-06 05:55:01

FIG. 12A



DOCKET ALARM

FIG. 12B

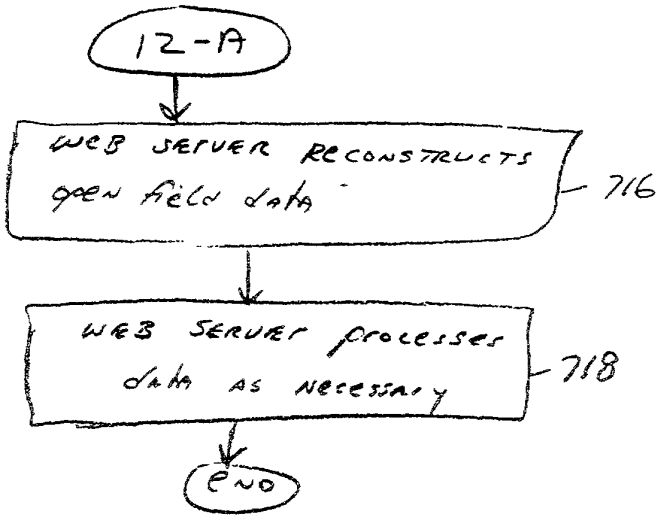
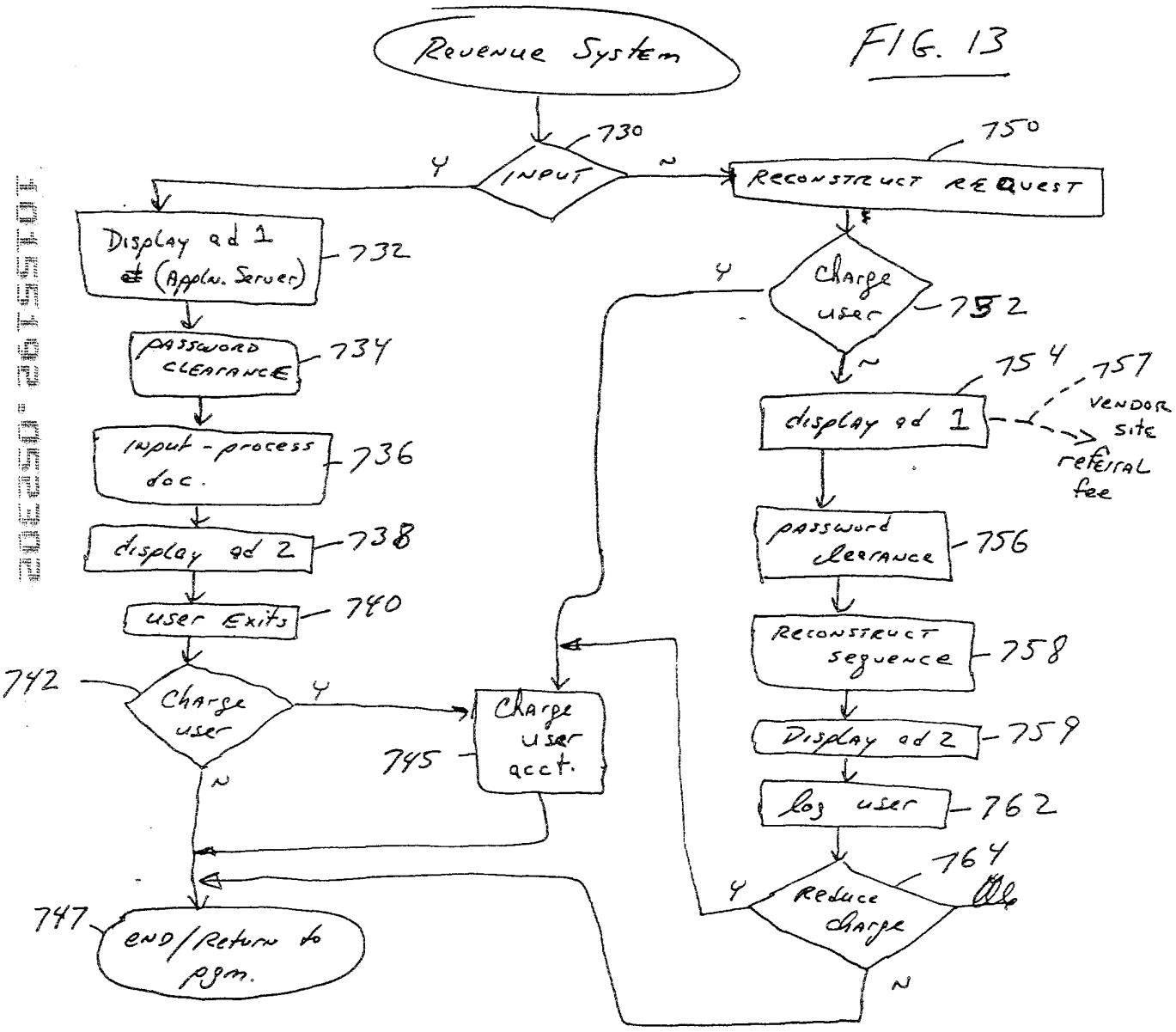


EXHIBIT 1002 - PAGE 111 OF 151

09/25/06, 36154.01

FIG. 13



DECLARATION AND POWER OF ATTORNEY FOR PATENT APPLICATION

As a below named inventor, I hereby declare that:

My residence, post office address and citizenship are as stated below next to my name,

I believe I am the original, first and sole inventor (if only one name is listed below) or an original, first and joint inventor (if plural names are listed below) of the subject matter which is claimed and for which a patent is sought on the invention entitled Data Security System and Method With Parsing and Dispersion Techniques the specification of which

(check one) ☒ is attached hereto.

☐ was filed on _____

as Application Serial No. _____

and was amended on _____ (if applicable).

I hereby state that I have reviewed and understand the contents of the above identified specification, including the claims, as amended by any amendment referred to above.

I acknowledge the duty to disclose information which is material to the examination of this application in accordance with Title 37, Code of Federal Regulations, §1.56(a).

I hereby claim foreign priority benefits under Title 35, United States Code, §119 (a-d) or §365(b) of any foreign application(s) for patent or inventor's certificate, or §365(a) of any PCT international application which designated at least one country other than the United States of America, listed below and have also identified below any foreign application for patent or inventor's certificate, or any PCT international application having a filing date before that of the application on which priority is claimed:

Prior Foreign Application(s)			Priority Claimed	
_____ (Number)	_____ (Country)	_____ (Day/Month/Year Filed)	☐ Yes	☐ No
_____ (Number)	_____ (Country)	_____ (Day/Month/Year Filed)	☐ Yes	☐ No

I hereby claim the benefit under Title 35, United States Code, §120 of any United States application(s) listed below and, insofar as the subject matter of each of the claims of this application is not disclosed in the prior United States application in the manner provided by the first paragraph of Title 35, United States Code, §112, I acknowledge the duty to disclose material information as defined in Title 37, Code of Federal Regulations, §1.56(a) which occurred between the filing date of the prior application and the national or PCT international filing date of this application:

09/916,397 (Application Serial No.)	07/27/01 (Filing Date)	pending (Status) (patented, pending, abandoned)
10/008,209 (Application Serial No.)	12/6/01 (Filing Date)	pending (Status)
10/008,218 (Application Serial No.)	12/6/01 (Filing Date)	pending (Status) (patented, pending, abandoned)

I hereby claim the benefit under Title 35, United States Code, §119(e) of any United States provisional application(s) listed below:

60/260,398, filed June 5, 2001 (Provisional Application No.)
60/287,813, filed May 2, 2001 (Provisional Application No.)
60/267,944, filed February 12, 2001 (Provisional Application No.)

60/247,242, filed November 12, 2000 (Provisional Application No.)
60/247,232, filed November 13, 2000 (Provisional Application No.)

I hereby declare that all statements made herein of my own knowledge are true and that all statements made on information and belief are believed to be true; and further that these statements were made with the knowledge that willful false statements and the like so made are punishable by fine or imprisonment, or both, under 18 U.S.C. 1001, and that such willful false statements may jeopardize the validity of the application or any patent issued thereon.
POWER OF ATTORNEY: As a named inventor, I hereby appoint the following attorney(s) and/or agents to prosecute this application and transact all business in the Patent and Trademark Office connected therewith.
Robert C. Kain, Jr., Reg. No. 30,648; Martin Fleit, Reg. No. 16,900; Jon A. Gibbons, Reg. No. 37,333; Jose Gutman, Reg. No. 35,171; and Steve Bongini, Reg. No. 40,917

Send Correspondence to:	Robert C. Kain, Jr., **Customer Number 004897** Fleit, Kain, Gibbons, Gutman & Bongini, P.L. 750 Southeast Third Avenue, Suite 100 Ft. Lauderdale, Florida 33316-1153
Direct Telephone Calls to:	Robert C. Kain, Jr., (954) 768-9002

Ron M. Redlich
Full name of sole or first inventor

Inventor's signature — Date: 5.20.4

Miami Beach, FL
Residence

U.S.
Citizenship

4201 Collins Avenue, Apt. 2103
Post Office Address

Miami Beach, FL 33140

Martin A. Nemzow
Full name of second joint inventor, if any

Inventor's signature — Date: 05-20-2002

Miami Beach, FL
Residence

U.S.
Citizenship

2915 Flamingo Drive
Post Office Address

Miami Beach, FL 33140

(Supply similar information and signature for third and subsequent joint inventors.)

\\Tigerdata share\RCK\CLIENTS\Redlich\6851-2-cip-3 dec.wpd

DANIEL K. SCHERRER
MY COMMISSION # CC 991952
EXPIRES: January 7, 2005
Bonded Thru Notary Public Underwriters

DANIEL K. SCHERRER
MY COMMISSION # CC 991952
EXPIRES: January 7, 2005
Bonded Thru Notary Public Underwriters

5-28-02

A

J1025 U.S. PTO
05/23/02

PATENTS

Jc971 U.S. PTO
10/155192
05/23/02

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re Application of: Ron Redlich, et al.

Serial No.:

Filed: Herewith

For: Data Security System and Method With Parsing and Dispersion Techniques

TRANSMITTAL AND CERTIFICATE OF EXPRESS MAILING

Box PATENT APPLICATION
Commissioner for Patents
Washington, D.C. 20231

Sir:

I HEREBY CERTIFY that the below listed documents are being deposited with the United States Postal Service "Express Mail Post Office to Address," Mailing Label No. **EL 777613587 US**, under 37 C.F.R. 1.10 on May 23, 2002 and are addressed to Box Patent Application, Commissioner for Patents, Washington, D.C. 20231.

30-25-50 * 05-23-02

X Patent application (93 pgs.; 27 clms.)

X Seventeen (17) sheets drawings - informal

X Declaration and Power of Attorney for Patent Application (2 pgs.)

X Assignment to Digital Doors, Inc. w/assignment cover sheet (5 pages)

X IDS and Form PTO-1449 (5 pgs.) with Twenty three (23) references

X Credit Card Payment Form in the amount of $515.00

X Fee calculation letter (1 pg.) (duplicate)

X Return receipt postcard.

X Please charge any underpayments or credit any overpayments to Deposit Account No. 03-1231. A duplicate copy of this transmittal is enclosed.

Respectfully submitted,

By _____
Ellen Kotler

By _____ William Trueba
Robert C. Kain, Jr. Reg. No. 48002
Reg. No. 30,648
FLEIT, KAIN, GIBBONS, GUTMAN & BONGINI, P.L.
750 S.E. Third Avenue, Suite 100
Ft. Lauderdale, Florida 33316-1153
(954) 768-9002

\\Tiger\data share\RCK\CLIENTS\Redlich\6851-2 CIP-3 EXPMAIL PAT wpd

FLEIT, KAIN, GIBBONS, GUTMAN & BONGINI P.L.
ATTORNEYS AT LAW

Miami • Fort Lauderdale • Boca Raton

750 Southeast Third Avenue
Suite 100
Fort Lauderdale, Florida 33316-1153
Telephone: (954) 768-9002
Toll Free: (800) 846-0900
Facsimile: (954) 768-0158
www.FleitKain.com
Robert C. Kain, Jr. rkain@fleitkain.com
William R. Trueba, Jr. wtrueba@fleitkain.com

May 23, 2002

Commissioner for Patents
Washington, D.C. 20231

Transmitted herewith for filing are papers related to the patent application of:

Inventor(s): Ron Redlich, et al.
For: Data Security System and Method With Parsing and Dispersion Techniques
Our Ref: 6851-2-CIP-3

The filing fee for the enclosed patent application has been calculated as shown below:

			Small Entity		Other than Small Entity	
	No. Filed	No. Extra	Rate	Fee	Rate	Fee
Basic Fee				$370		$740
Total Claims	27 - 20 =	7	x 9	=$ 63	x 18	=$
Ind. Claims	4 - 3 =	1	x 42	=$ 42	x 84	=$
____Multiple claims presented			x140	=$	x 280	=$
			Total	$475	Total	$

A Credit Card Payment Form in the amount of $515.00 is enclosed to cover the filing fee (plus applicable Assignment recording fee). Please charge any additional filing fees required under 37 CFR 1.16 to Deposit Account No. 03-1231. A duplicate copy of this letter is enclosed.

Small Entity under 37 C.F.R. §1.27. YES __X____ NO _____

Respectfully submitted,

Fleit, Kain, Gibbons, Gutman & Bongini, P.L.

By _____ WILLIAM TRUEBA
FOR Robert C. Kain, Jr. REG. NO. 48,002
Reg. No. 30,648

\\Tiger\data share\RCK\CLIENTS\Redlich\6851-2cip-3 fee calc ltr pto credit card wpa.wpd

PATENT APPLICATION SERIAL NO. _____

U.S. DEPARTMENT OF COMMERCE
PATENT AND TRADEMARK OFFICE
FEE RECORD SHEET

05/30/2002 WASFAW1 00000005 10155192

01 FC:201	370.00	OP
02 FC:202	42.00	OP
03 FC:203	63.00	OP

PTO-1556
(5/87)

*U.S. Government Printing Office: 2001 — 481-697/59173

PATENT APPLICATION FEE DETERMINATION RECORD
Effective October 1, 2001

Application or Docket Number: 6851-2 CIP-3

CLAIMS AS FILED - PART I

	(Column 1)	(Column 2)
TOTAL CLAIMS	27	
FOR	NUMBER FILED	NUMBER EXTRA
TOTAL CHARGEABLE CLAIMS	27 minus 20=	* 7
INDEPENDENT CLAIMS	4 minus 3 =	* 1
MULTIPLE DEPENDENT CLAIM PRESENT		☐

* If the difference in column 1 is less than zero, enter "0" in column 2

SMALL ENTITY TYPE ☐		OR	OTHER THAN SMALL ENTITY	
RATE	FEE		RATE	FEE
BASIC FEE	370.00	OR	BASIC FEE	740.00
X$ 9=	63	OR	X$18=	
X42=	42	OR	X84=	
+140=		OR	+280=	
TOTAL	475	OR	TOTAL	

CLAIMS AS AMENDED - PART II

AMENDMENT A	(Column 1) CLAIMS REMAINING AFTER AMENDMENT		(Column 2) HIGHEST NUMBER PREVIOUSLY PAID FOR	(Column 3) PRESENT EXTRA
Total	*	Minus	**	=
Independent	*	Minus	***	=
FIRST PRESENTATION OF MULTIPLE DEPENDENT CLAIM				☐

SMALL ENTITY		OR	OTHER THAN SMALL ENTITY	
RATE	ADDITIONAL FEE		RATE	ADDITIONAL FEE
X$ 9=		OR	X$18=	
X42=		OR	X84=	
+140=		OR	+280=	
TOTAL ADDIT. FEE		OR	TOTAL ADDIT. FEE	

AMENDMENT B	(Column 1) CLAIMS REMAINING AFTER AMENDMENT		(Column 2) HIGHEST NUMBER PREVIOUSLY PAID FOR	(Column 3) PRESENT EXTRA
Total	*	Minus	**	=
Independent	*	Minus	***	=
FIRST PRESENTATION OF MULTIPLE DEPENDENT CLAIM				☐

RATE	ADDITIONAL FEE		RATE	ADDITIONAL FEE
X$ 9=		OR	X$18=	
X42=		OR	X84=	
+140=		OR	+280=	
TOTAL ADDIT. FEE		OR	TOTAL ADDIT. FEE	

Best Available Copy

AMENDMENT C	(Column 1) CLAIMS REMAINING AFTER AMENDMENT		(Column 2) HIGHEST NUMBER PREVIOUSLY PAID FOR	(Column 3) PRESENT EXTRA
Total	*	Minus	**	=
Independent	*	Minus	***	=
FIRST PRESENTATION OF MULTIPLE DEPENDENT CLAIM				☐

RATE	ADDITIONAL FEE		RATE	ADDITIONAL FEE
X$ 9=		OR	X$18=	
X42=		OR	X84=	
+140=		OR	+280=	
TOTAL ADDIT. FEE		OR	TOTAL ADDIT. FEE	

* If the entry in column 1 is less than the entry in column 2, write "0" in column 3.
** If the "Highest Number Previously Paid For" IN THIS SPACE is less than 20, enter "20."
***If the "Highest Number Previously Paid For" IN THIS SPACE is less than 3, enter "3."
The "Highest Number Previously Paid For" (Total or Independent) is the highest number found in the appropriate box in column 1.

FORM PTO-875 (Rev. 8/01) *U.S. Government Printing Office: 2[illegible] – 484-484/59268 Patent and Trademark Office, U.S. DEPARTMENT OF COMMERCE

CLAIMS ONLY

SERIAL NO.	FILING DATE
APPLICANT(S)	

CLAIMS

	AS FILED		AFTER 1st AMENDMENT		AFTER 2nd AMENDMENT	
	IND.	DEP.	IND.	DEP.	IND.	DEP.
1	1					
2						
3						
4						
5						
6						
7	1					
8						
9						
10						
11						
12						
13						
14						
15						
16	1					
17						
18						
19						
20						
21						
22	1					
23						
24						
25						
26						
27	10					
28						
29						
30						
31						
32						
33						
34						
35						
36						
37						
38						
39						
40						
41						
42						
43						
44						
45						
46						
47						
48						
49						
50						
TOTAL IND.	4					
TOTAL DEP.	23					
TOTAL CLAIMS	27					

	*		*		*	
	IND.	DEP.	IND.	DEP.	IND.	DEP.
51						
52						
53						
54						
55						
56						
57						
58						
59						
60						
61						
62						
63						
64						
65						
66						
67						
68						
69						
70						
71						
72						
73						
74						
75						
76						
77						
78						
79						
80						
81						
82						
83						
84						
85						
86						
87						
88						
89						
90						
91						
92						
93						
94						
95						
96						
97						
98						
99						
100						
TOTAL IND.						
TOTAL DEP.						
TOTAL CLAIMS						

* *MAY BE USED FOR ADDITIONAL CLAIMS OR ADMENDMENTS*

FORM PTO-2022 (1-98)

U.S.DEPARTMENT OF COMMERCE
Patent and Trademark Office

*U.S. Government Printing Office: 1998 - 433-214/70303

Best Available Copy

#2

PATENTS



IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re Application of Ron Redlich, et al.

Serial No.

Filed: Herewith

For: Data Security System and Method with Parsing and Dispersion Techniques

INFORMATION DISCLOSURE STATEMENT

Commissioner for Patents
Washington, D.C. 20231

Sir:

As a means of complying with the duty of disclosure set forth in 37 C.F.R. §1.56, applicant, through the undersigned attorney, files this Information Disclosure Statement pursuant to 37 C.F.R. §1.97.

Patent No.	Inventor	Issue Date
6,192,472 B1	Garay et al.	Feb. 20, 2001
6,148,342	Ho	Nov. 14, 2000
6,094,483	Fidrich et al.	Jul. 25, 2000
6,055,544	DeRose et al.	Apr. 25, 2000
5,996,011	Humes	Nov. 30, 1999
5,905,980	Masuichi et al.	May 18, 1999
5,832,212	Cragun et al.	Nov. 3, 1998
5,485,474	Rabin	Jan. 16, 1996

5,892,908	Hughes et al.	Apr. 6, 1999
5,423,032	Byrd et al.	Jun. 6, 1995
6,044,375	Shmueli et al.	Mar. 28, 2000
5,715,399	Bezos	Feb. 3, 1998

-Ingran i100, Content Security Appliance (2 pages)

-Element-Wise XML Encryption, Hiroshi Maruyama and Takeshi Imamura, IBM Research, Tokyo Research Laboratory (4 pages)

-Survival Information Storage Systems by Jay J. Wylie, Michael W. Bigrigg, John D. Strunk, Gregory R. Ganger, Han Kiliccote Pradeep K. Khosla (8 pages)

-ZD Net Interactive week - IBS, SAP: XML to Boost Security, integration.

-Myers, A.C. "Mostly-Static Decentralized Information Flow Control" M.I.T. Doctoral Thesis January, 1999.

-Cisco Systems' Intrusion Detection System (IDS) Host Sensor 2.0 product data sheet and technical overview, 10 pgs., July 3, 2000, www.cisco.com.

-"Protecting Privacy using the Decentralized Label Model" A. Myers, Cornell Univ. 1/1/2000.

-Mimesweeper Product Data sheet, Resoft Int'l LLC., Nov. 12, 2001 re-soft.com/product/mimeswep.

-Complete Content Security for Internet Gateways and Mail Servers, Aladdin, July 2000.

-Aladdin Product announcement for e-Safe Appliance, Aladdin Knowledge Sys., Ltd., Nov. 8, 2001, Aks.com/news/2001/e-safe.

-Mail Attender 4.1x product white paper by Sherpa Software, Oct., 2001.

This Information Disclosure Statement is filed in accordance with 37 C.F.R. §1.97, and this statement shall not be construed as a representation that a search was made, has been made, or that the information cited above is, or is considered to be, material to patentability as defined in §1.56(b), or that no other material information, as defined in 37 C.F.R. §1.56(b), exists. A copy of each listed patent or publication or other item of information in written form, or of at least the portions thereof listed above, is filed with this Information Disclosure Statement. If two or more patents or publications are considered substantially cumulative, a copy of a representative one is included with this statement and others are merely listed above. Translations of the pertinent portions of foreign language patents or publications listed above are supplied if existing translations are readily available to the applicant or the applicant's attorney. A concise explanation of the relevance, as it is presently understood by the individual designated in 37 CFR §1.56(c) most knowledgeable about the content of the information, of each patent, publication, or other information listed that is not in the English language is provided either in the patent specification or in the enclosed attachment.

Respectfully submitted,

FLEIT, KAIN, GIBBONS, GUTMAN & BONGINI, P.L.

By _____ WILLIAM TRUEBA
FOR Robert C. Kain, Jr. REG. No. 48002
Reg. No. 30,648
Suite 100
750 S.E. Third Avenue
Ft. Lauderdale, Florida 33316-1153
(954) 768-9002

minor\6851-2-cip-3-IDS

jc971 U.S. PTO
10/155192
05/23/02

INFORMATION DISCLOSURE CITATION *(Use several sheets if necessary)*	Docket Number (Optional) 6851-2 CIP-3	Application Number
	Applicant(s) Ron Redlich	
	Filing Date	Group Art Unit

U.S. PATENT DOCUMENTS

Examiner Initial	Document Number	Date	Name	Class	Subclass	Filing Date if Appropriate
	6,192,472 B1	2/20/2001	Garay et al.	713	165	
	6,148,342	11/14/2000	Ho	709	225	
	6,094,483	7/25/2000	Fridrich et al.	380	28	
	6,055,544	4/25/2000	DeRose et al.	707	104	
	5,996,011	11/30/1999	Humes	709	225	
	5,905,980	5/18/1999	Masuichi et al.	707	1	
	5,832,212	11/3/1998	Cragun et al.	395	188.01	
	5,485,474	1/16/1996	Rabin	371	37.1	

FOREIGN PATENT DOCUMENTS

	Document Number	Date	Country	Class	Subclass	Translation YES NO

OTHER DOCUMENTS (Including Author, Title, Date, Pertinent Pages, Etc.)

	Ingram i100, Content Security Appliance (2 pages)	
	Element-Wise XML Encryption, Hiroshi Maruyama and Takeshi Imamura, IBM Research, Tokyo Research Laboratory (4 pages)	
	Survival Information Storage Systems by Jay J. Wylie, Michael W. Bigrigg, John D. Strunk, Gregory R. Ganger, Han Kiliccote Pradeep K. Khosla (8 pages)	
	ZD Net Interactive iWeek - IBS, SAP: XML to Boost Security Integration (1 page)	
	Myers, A.C. "Mostly-Static Decentralized Information Flow Control" M.I.T. Doctoral Thesis January, 1999	
	Cisco Systems' Intrusion Detection System (IDS) Host Sensor 2.0 product data sheet and technical overview, 10 pgs., July 3, 2000, www.cisco.com	

EXAMINER DATE CONSIDERED

INFORMATION DISCLOSURE CITATION *(Use several sheets if necessary)*	Docket Number (Optional) 6851-2 CIP-3	Application Number
	Applicant(s) Ron Redlich	
	Filing Date	Group Art Unit

U.S. PATENT DOCUMENTS

Examiner Initial	Document Number	Date	Name	Class	Subclass	Filing Date if Appropriate
	5,892,908	4/6/1999	Hughes et al.	395	200.8	
	5,423,032	6/6/1995	Byrd et al.	395	600	
	6,044,375	3/28/2000	Shmueli et al.	707	101	
	5,715,399	2/3/1998	Bezos	395	227	

FOREIGN PATENT DOCUMENTS

	Document Number	Date	Country	Class	Subclass	Translation YES NO

OTHER DOCUMENTS (Including Author, Title, Date, Pertinent Pages, Etc.)

	"Protecting Privacy using the Decentralized Label Model" A. Myers, Cornell Univ. 1/1/2000	
	Mimesweeper Product Data sheet, Resoft Int'l LLC., Nov. 12, 2001 re-soft.com/product/mimeswep.	
	Complete Content Security for Internet Gateways and Mail Servers, Aladdin, July 2000.	
	Aladdin Product announcement for e-Safe Appliance, Aladdin Knowledge Sys., Ltd., Nov. 8, 2001, Aks.com/news/2001/e-safe.	
	Mail Attender 4.1x product white paper by Sherpa Software, Oct., 2001.	

EXAMINER DATE CONSIDERED

minor\6851-2-cip-3 form pto-1449

Mostly-Static Decentralized Information Flow Control

Andrew C. Myers

January 1999

© Massachusetts Institute of Technology 1999

This research was supported in part by DARPA contract F00014-91-J-4136, monitored by the Office of Naval Research, and by DARPA contracts F30602-96-C-0303 and F30602-98-1-0237, monitored by USAF Rome Laboratory. The author was supported by an Intel Foundation Fellowship.

Massachusetts Institute of Technology
Laboratory for Computer Science
Cambridge, Massachusetts, USA

Mostly-Static Decentralized Information Flow Control

by

Andrew C. Myers

Abstract

The growing use of mobile code in downloaded programs such as applets and servlets has increased interest in robust mechanisms for ensuring privacy and secrecy. Common security mechanisms such as sandboxing and access control are either too restrictive or too weak—they prevent applications from sharing data usefully, or allow private information to leak. For example, security mechanisms in Java prevent many useful applications while still permitting Trojan horse applets to leak private information. This thesis describes the *decentralized label model*, a new model of information flow control that protects private data while allowing applications to share data. Unlike previous approaches to privacy protection based on information flow, this label model is *decentralized*: it allows cooperative computation by mutually distrusting principals, without mediation by highly trusted agents. Cooperative computation is possible because individual principals can *declassify* their own data without infringing on other principals' privacy. The decentralized label model permits programs using it to be checked statically, which is important for the precise detection of information leaks.

This thesis also presents the new language *JFlow*, an extension to the Java programming language that incorporates the decentralized label model and permits static checking of information flows within programs. Variable declarations in JFlow programs are annotated with labels that allow the static checker to check programs for information leaks efficiently, in a manner similar to type checking. Often, these labels can be inferred automatically, so annotating programs is not onerous. Dynamic checks also may be used safely when static checks are insufficiently powerful. A compiler has been implemented for the JFlow language. Because most checking is performed statically at compile time, the impact on performance is usually small.

Keywords: constraint solving, covert channels, integrity, Java, labels, principals, privacy, programming languages, role hierarchy, security, static checking, trojan horse, trusted computing base, type systems, verification

This report is a minor revision of the dissertation of the same title submitted to the Department of Electrical Engineering and Computer Science on January 7, 1999, in partial fulfillment of the requirements for the degree of Doctor of Philosophy in that department. The thesis was supervised by Professor Barbara Liskov.

Ex. 1002, Page 126

Acknowledgments

Many people deserve thanks for helping me make this work, my degree, and my love of research, a reality.

First, I must thank my advisor, Barbara Liskov, for her continuous support and the inspiration she has provided me. If I am a better writer or speaker because of my time at MIT, it is largely due to her. I hope that someday I will approach her clarity of thought and expression, and her ability to focus on the important.

My thesis committee provided many excellent suggestions on both the form and content of this work. Martín Abadi has given thoughtful and extremely useful comments at several phases of this work; I greatly appreciate his support and his meticulous reading of this thesis. John Guttag suggested many useful clarifications. Butler Lampson had several insightful suggestions as I was developing this work and helped me figure out how to make some of the central ideas of this thesis more intuitive.

Many other people also have provided useful feedback on this thesis and on the work as a whole. Chandrasekhar Boyapati, Miguel Castro, and Stephen Garland deserve special thanks for their in-depth readings of my work and helpful suggestions over the last couple of years. Nick Mathewson provided much useful code that I was able to borrow.

It has been a pleasure to be a graduate student in the Programming Methodology Group, and I would like to thank the group members for creating such a lively and thought-provoking environment. Atul Adya, Joseph Bank, Phillip Bogle, Chandra, Miguel, Mark Day, Sanjay Ghemawat, Jason Hunter, Umesh Maheshwari, Quinton Zondervan, and many others made life richer, not only because of their considerable abilities, but also because they were willing to make movie runs, share dinners, suggest good books, play ping-pong, and swap evil puzzles.

Other friends have helped over the years. Ulana Legedza has always been there to help refine talks and papers. In addition to his useful feedback on all of my work, Jim O'Toole was a stimulating housemate in my early years at MIT. Joe Boyt and Paul Isherwood often flew into town and enlivened my graduate life.

Many other people have made me a better researcher and computer scientist. Greg Nelson of SRC made me appreciate the value of a well-written specification. I learned about user interfaces and good design from Rob Myers of Silicon Graphics, and Brendan Eich, Kipp Hickman, Bruce Karsh, and others from SGI made me a better programmer and collaborator. I would also like to thank Franklyn Prendergast, S. Vuk-Pavlovic, and Zeljko Bajzer of the Mayo Clinic, who introduced me to research and helped me realize that I wanted to become a scientist.

My parents and the rest of my family have been incredibly supportive over the years of my quest for a doctoral degree. My parents always have pushed me to achieve to the limits of my ability and have done what they could to ensure that I had the best education possible. I owe them much gratitude.

Finally, my greatest asset for the last three years has been my wife, Kavita Bala. She is always emotionally supportive, but she is also my best reviewer and the person I first bounce ideas off of. Even while she worked to complete her own doctoral degree, she helped me unstintingly in every conceivable way, right from the start in a Barcelona park where she helped nurture an offbeat idea into my thesis topic.

Contents

1 Introduction **11**

1.1 Example . . . 12
1.2 Existing security techniques . . . 14
1.3 Decentralized information flow control . . . 15
1.3.1 Decentralized label model . . . 15
1.3.2 Static information flow analysis . . . 16
1.4 Trusted computing base . . . 18
1.5 Applications . . . 19
1.6 Limitations . . . 19
1.7 Outline . . . 20

2 The Label Model **21**

2.1 Basic model . . . 22
2.1.1 Principals . . . 22
2.1.2 Labels . . . 22
2.1.3 Relabeling by restriction . . . 23
2.1.4 Computation and label join . . . 24
2.1.5 Relabeling by declassification . . . 25
2.1.6 Channels . . . 26
2.2 Examples . . . 27
2.2.1 Tax preparer example . . . 27
2.2.2 Hospital example . . . 28
2.3 Extending and interpreting labels . . . 30
2.3.1 Limitations of the subset relabeling rule . . . 30
2.3.2 Interpreting labels . . . 32
2.3.3 Formalizing the principal hierarchy . . . 33
2.3.4 Label interpretation function . . . 33
2.3.5 Flow set constraints . . . 34
2.3.6 Label functions . . . 35
2.4 Checking relabeling statically . . . 37
2.4.1 Annotations . . . 38
2.4.2 Static correctness condition . . . 39
2.4.3 A sound and complete relabeling rule . . . 40
2.4.4 Static checking . . . 44
2.5 Output channels . . . 50
2.6 Generalizing labels and principals . . . 51
2.6.1 Integrity policies . . . 51

Ex. 1002, Page 128

2.6.2 Combining integrity and privacy . . . 54
2.6.3 Generalizing principals and the acts-for relation . . . 55
2.7 Summary . . . 59

3 The JFlow Language **60**
3.1 Static vs. dynamic checking . . . 62
3.2 Language support for information flow checking . . . 63
3.2.1 Labeled types . . . 63
3.2.2 Implicit flows . . . 64
3.2.3 Termination channels . . . 65
3.2.4 Run-time labels . . . 66
3.2.5 Reasoning about principals . . . 68
3.2.6 Declassification . . . 69
3.2.7 Run-time principals . . . 70
3.3 Interactions with features of Java . . . 70
3.3.1 Method declarations . . . 71
3.3.2 Default labels . . . 72
3.3.3 Method constraints . . . 73
3.3.4 Exceptions . . . 75
3.3.5 Parameterized classes . . . 75
3.3.6 Arrays . . . 78
3.3.7 Run-time type discrimination . . . 79
3.3.8 Authority declarations . . . 79
3.3.9 Inheritance and constructors . . . 80
3.4 Examples . . . 82
3.4.1 Example: passwordFile . . . 82
3.4.2 Example: Protected . . . 84
3.5 Limitations . . . 84
3.6 Grammar extensions . . . 85
3.6.1 Label expressions . . . 85
3.6.2 Labeled types . . . 86
3.6.3 Class declarations . . . 86
3.6.4 Method declarations . . . 87
3.6.5 New statements . . . 88
3.6.6 New expressions . . . 88

4 Statically Checking JFlow **90**
4.1 Correctness . . . 90
4.2 Static checking framework . . . 91
4.2.1 Type checking vs. label checking . . . 92
4.2.2 Environments . . . 92
4.2.3 Exceptions . . . 93
4.2.4 Additional notation conventions . . . 96
4.2.5 Environment bindings . . . 97
4.2.6 Representing principals . . . 98
4.2.7 Representing labels and components . . . 98
4.2.8 Representing types . . . 100
4.2.9 Invariant vs. covariant types . . . 100

Ex. 1002, Page 129

4.3 Basic rules . . . 102
4.3.1 Reasoning about principals . . . 102
4.3.2 Reasoning about labels . . . 102
4.3.3 Class scope and environments . . . 103
4.3.4 Reasoning about subtypes . . . 106
4.4 Checking Java statements and expressions . . . 108
4.4.1 Simple rules . . . 108
4.4.2 Arithmetic . . . 109
4.4.3 Local variables . . . 110
4.4.4 Variable access . . . 110
4.4.5 Variable assignment . . . 112
4.4.6 Compound statements . . . 113
4.4.7 Goto-like statements . . . 114
4.4.8 Exceptions . . . 115
4.4.9 Dynamic type discrimination . . . 116
4.5 Checking new statements and expressions . . . 117
4.5.1 Testing the principal hierarchy . . . 117
4.5.2 Declassification . . . 118
4.5.3 Run-time label tests . . . 119
4.6 Method and constructor calls . . . 120
4.6.1 Generic checking . . . 120
4.6.2 Specific rules for checking calls . . . 122
4.7 Checking classes and methods . . . 123
4.7.1 Checking classes . . . 123
4.7.2 Class authority . . . 125
4.7.3 Method signature compatibility . . . 125
4.7.4 Method declarations . . . 127

5 Constraint Solving and Translation **133**
5.1 Constraint solving . . . 133
5.1.1 Integrating static checking and constraint solving . . . 134
5.1.2 Constraint equations . . . 134
5.1.3 Solving constraints . . . 136
5.1.4 Determining the meet of two components . . . 137
5.1.5 Handling dynamic constraints . . . 138
5.1.6 Recursion in dynamic components . . . 140
5.1.7 Ordering the relaxation steps . . . 141
5.1.8 Empirical comparisons . . . 144
5.2 Translation . . . 146
5.2.1 Principal values and the actsFor statement . . . 146
5.2.2 Label values and the switch label statement . . . 148

6 Related Work **149**
6.1 Access control . . . 149
6.2 Limitations of discretionary access control . . . 150
6.3 Information flow control . . . 151
6.4 Static enforcement of security policies . . . 155
6.5 Modeling principals and roles . . . 156

Ex. 1002, Page 130

6.6 Cryptography 157
6.7 Covert channels 157

7 Conclusions **159**
7.1 Decentralized label model 159
7.2 Static analysis of information flow 160
7.3 Future work 161

Ex. 1002, Page 131

Chapter 1

Introduction

Computer security is becoming increasingly important, as the result of several ongoing trends. Computers everywhere are becoming inextricably connected to the Internet. Increasingly, computation and even data storage are distributed to geographically remote and untrusted sites, and both programs and data are becoming highly mobile. Sensitive personal, corporate, and government data is being placed online and is routinely accessed over networks. The number of users and other interacting entities also continues to increase rapidly, and trust relationships among these entities are growing increasingly complex. In short, there is more to protect and it is more difficult to protect it.

It is difficult even to characterize what protection is needed. Abstractly, the goal of computer security is to ensure that all computations obey some set of policies, but there are two central goals of computer security: private or secret data should not be leaked to parties that might misuse it, and valuable data should not be damaged or destroyed by other parties. These complementary goals will be referred to here as *privacy* and *integrity*. This thesis focuses on the protection of privacy, though integrity is also considered briefly. Protecting privacy and secrecy of data has long been known to be a very difficult problem, and existing security techniques do not provide satisfactory solutions to this problem.

Systems that support the downloading of distrusted code are particularly in need of better protection for privacy. For example, Java [GJS96] supports downloading of code from remote sites, creating the possibility that the downloaded code will transfer private data to those sites. Suppose a user computes his taxes using a downloaded applet. The user cannot ensure that the applet will not transfer his tax information back to the applet provider. Java attempts to prevent improper transfers by using a compartmental security model called the *sandbox model* [FM96, MF96], but this approach largely prevents applications from sharing data, while still permitting privacy violations like the one just described. A key problem is that information must be shared with downloaded code, while preventing that code from leaking the information.

There is no generally accepted definition of what it means to protect privacy. A distinction sometimes has been drawn between privacy and other security goals such as *secrecy* or *confidentiality*. Sometimes privacy is identified with the weaker goal of *anonymity*: protecting the identity of various parties, as in a medical protocol, rather than their data, as in [Swe96]. However, in this work the terms privacy and secrecy

Ex. 1002, Page 132

are considered to be synonymous; they both refer to the ability to control information leakage of any kind. The use of the term privacy emphasizes that in a decentralized environment, no generally accepted notion of the sensitivity of data exists. Users generally consider their own data to be private, and are naturally less concerned with the privacy of the data of other users. However, the privacy requirements of all users are treated as equally important.

In general, security enforcement mechanisms may be internal or external to the computing system. Internal mechanisms attempt to prevent security violations by making them impossible; external mechanisms, such as the threat of legal action, attempt to convince users not to initiate computation that would violate security. Current security mechanisms, both internal and external, are becoming less viable as the computing system becomes large, decentralized, anonymous, and international.

With the widespread downloading of code, dealing with untrusted programs becomes a greater issue for security than in the past. Conventionally, the focus is placed on protecting the operating system from buggy or malicious programs, and on protecting users from each other. On most computer systems, the programs that might be used to violate user privacy are programs already installed on the system, and purchased from some vendor. Since the source of the program is known, some form of external redress is available if the program is found to violate privacy. When programs such as Java applets are dynamically downloaded and executed, the ability to identify and exact redress from the supplier of privacy-violating code is reduced. Therefore, the goal of this work is to develop better internal mechanisms, preventing *programs* from violating security policies rather than convincing *users* not to.

In another sense, the goal of this work is to reduce the cost of ensuring security—a cost that is passed on to users. If a user downloads a free application, the user accepts either the risk that a program will violate security, or the considerable cost of ensuring that a program does not violate security. This observation applies to commercial software as well; a company providing an application must ensure that it does not violate user security, or else be liable in cases where it violates security, at least in the sense that the reputation of the company may suffer. With both kinds of software, the cost is passed on to the users of that application. Better internal mechanisms that can be applied either by end-users or by software developers should reduce this cost.

1.1 Example

Figure 1.1 depicts an example with security requirements that cannot be satisfied using existing techniques. This scenario contains mutually distrusting principals that must cooperate to perform useful work. In the example, the user Bob is preparing his tax form using both a spreadsheet program and a piece of software called "WebTax". Bob would like to be able to prepare his final tax form using WebTax, but he does not trust WebTax to protect his privacy. The computation is being performed using two programs: a spreadsheet that he trusts and grants his full authority to, and the WebTax program, which he does not trust. Bob would like to transmit his tax data from the spreadsheet to WebTax and receive a final tax form as a result, while

Ex. 1002, Page 133

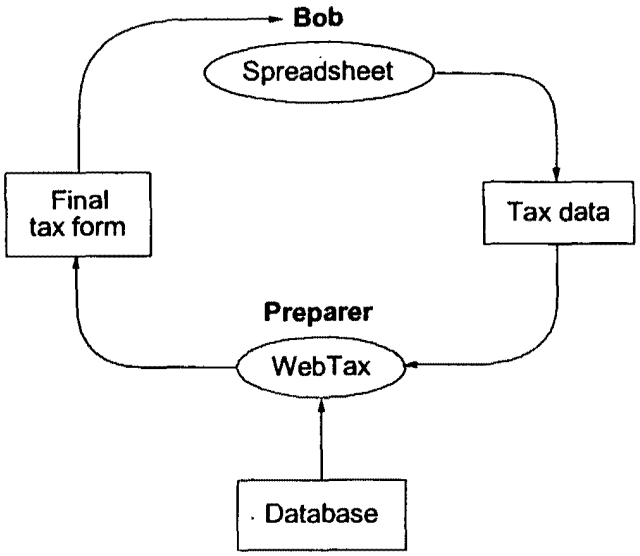


Figure 1.1: A simple example

being protected against WebTax leaking his tax information.

In this example, there is another principal named Preparer that has privacy interests. The principal Preparer represents a firm that distributes the WebTax software. The WebTax application computes the final tax form using a proprietary database, shown at the bottom, that is owned by Preparer. This database might, for example, contain algorithms for minimizing tax payments. Since this principal is the source of the WebTax software, it trusts the program not to distribute the proprietary database through malicious action, though the program might leak information because it contains bugs.

In principle, it may be difficult to prevent some information about the database contents from leaking back to Bob, particularly if Bob is able to make a large number of requests and then carefully analyze the resulting tax forms. This information leak is not a practical problem if Preparer can charge Bob a per-form fee that exceeds the value of the information Bob obtains through each form.

To make this scenario work, the Preparer principal needs two pieces of functionality. First, it needs protection against accidental or malicious release of information from the database by paths other than through the final tax form. Second, it needs the ability to *sign off* on the final tax form, confirming that the information leaked in the final tax form is sufficiently small or scrambled by computation that the tax form may be released to Bob.

It is worth noting that Bob and Preparer do need to trust that the execution platform has not been subverted. For example, if WebTax is running on a computer that Bob completely controls, then Bob will be able to steal the proprietary database. Clearly, Preparer cannot have any real expectation of privacy or secrecy if its private data is manipulated in unencrypted form by an execution platform that it does not trust!

In this thesis, it is assumed that the execution platform is trusted, even though the programs running on

that platform may not be. The issue of trust in the execution platform is discussed further in Section 1.4. Even with this assumption, this scenario cannot be implemented satisfactorily or even modeled using existing security techniques. With current techniques, Bob must carefully inspect the Webtax code and verify that it does not leak his data; in general, this task is difficult. The techniques described in this thesis allow the security goals of both Bob and Preparer to be met without this inspection; Bob and Preparer then can cooperate in performing useful computation. In another sense, this work shows how both Bob and Preparer can inspect the Webtax program efficiently and simply to determine whether it violates their security requirements.

1.2 Existing security techniques

Let us now briefly consider the application of existing security techniques to this problem; for a more in-depth discussion, see Chapter 6. When most people think of computer security, they think of well-established security techniques such as access control. Typical access control mechanisms (which support *discretionary access control*) do not protect privacy well when programs are not trusted: access control prevents unauthorized information release but does not control information propagation once the information has been accessed. For example, if a program A is allowed to read user B's data, B cannot control how A distributes the information it has read.

A less well-known approach to protecting privacy is *information flow control*. In information flow techniques (such as *mandatory access control*), every piece of data has an attached *sensitivity label*. The labels are typically from a small ordered set such as {unclassified, classified, secret, top secret}. The labels remain attached to data as it propagates through the system, preventing it from being released improperly even if it is released to an untrusted program. Data may be *relabeled* to further restrict its use (such as a relabeling from secret to top secret). However, relabeling data from top secret to secret (or allowing top secret data to affect secret data) would be *declassification* or *downgrading*, which could lead to an information leak.

Intuitively, information flow control protects privacy much more directly than access control does, but practical problems with information flow control have prevented its widespread adoption. Sensitivity labels are usually maintained dynamically, causing substantial loss of performance. Dynamic labels impose even greater run-time and storage overheads than access control mechanisms do, because for every primitive operation, the label of the result must be computed. Another limitation is that sensitivity labels are implicitly *centralized*: they express the privacy concerns of a single principal (typically, the government). If one considers providing privacy in a more decentralized setting, such as the community of Web users, it is clear that no universal notion of secret sensitivity can be established.

All practical information flow control systems provide the ability to *declassify* or *downgrade* data because strict information flow control is too restrictive for writing real applications. Declassification in these systems lies outside the model: it performed by a *trusted subject*, which is code possessing the authority of a

Ex. 1002, Page 135

highly trusted principal. However, the notion of a highly trusted principal does not extend to a decentralized system. Traditional information flow models do not support workable declassification for a decentralized environment.

Another important issue for information flow systems is the *precision* of the detection of information flow. Information is assumed to flow from one program value to another if there is any dependency between the values. Any unidentified dependency would create a potential information leak. However, it is also important to avoid false dependencies, since a false dependency results in data being overly restrictively labeled, and thus not usable in situations where it ought to be. To provide a precise determination of data dependencies, particularly dependencies arising from *implicit flows*, static analysis is required [DD77]. Dynamic enforcement of information flow control, as in mandatory access control systems [DOD85], can determine data dependencies conservatively—even dependencies arising from implicit flows—but results in false dependencies and overly restrictive labels.

1.3 Decentralized information flow control

The central goal of this work is to make information flow control a viable technique for providing privacy in a complex, decentralized world with mutually distrusting principals. This work has involved two major components, each of which is independently useful.

1.3.1 Decentralized label model

The first component is the development of a new model for labeling data that supports situations involving mutual distrust. This model allows users to control the flow of their information without imposing the rigid constraints of a traditional multilevel security system. It provides security guarantees to users and to groups rather than to a monolithic organization—in essence, it provides every principal with its own multilevel security.

The decentralized information flow model differs from previous work on information flow control: it introduces a notion of ownership of data, and allows users to explicitly declassify data that they own. When data is derived from several sources, all the sources own the data and must agree to release it. Previous work on information flow allowed declassification only by a *trusted agent* or *trusted subject* with essentially arbitrary powers of declassification; the notion of a universally trusted agent is clearly inapplicable to a decentralized environment. Declassification in this model provides a safe escape hatch from the rigid restrictions of strict information flow checking. Deciding when declassification is appropriate is outside the scope of this model; work in *inference controls* and *statistical databases* has developed some applicable methods [Den82].

The decentralized label model has a number of important properties that are discussed further in Chapter 2:

Ex. 1002, Page 136

- It allows individual principals to attach flow policies to pieces of data. The flow policies of all principals are reflected in the *label* of the data, and the system guarantees that all the policies are obeyed simultaneously. Therefore, the model works even when the principals do not trust each other.
- The model allows a principal to declassify data by modifying the flow policies in the attached label. Arbitrary declassification is not possible because flow policies of other principals are still maintained. Declassification permits the programmer to remove restrictions when appropriate; for example, the programmer might determine that the amount of his information being leaked is acceptable using techniques from information theory [Mil87].
- The model is compatible with static checking of information flow.
- It allows a richer set of safe relabelings than in previous label models [Den76, MMN90] by fully exploiting information about relationships between different principals.
- It has a formal semantics that allows a precise characterization of what relabelings are legal.
- The rule for static checking is shown to be both sound and complete with respect to the formal semantics: the rule allows only safe relabelings, and it allows all safe relabelings.
- In this model, labels form a lattice-like structure that helps make static checking of programs effective.
- The model can be applied in dual form to yield decentralized integrity policies.

1.3.2 Static information flow analysis

The second component of this work is a collection of new techniques for static analysis of information flow in programs. These techniques have been incorporated in the new language *JFlow*, an extension of the Java language [GJS96] that allows information in the program to be annotated with decentralized labels. These annotations can then be checked statically, allowing more precise, fine-grained determination of information flows within programs than in previous languages allowing static checking of information flow. Like other recent approaches [PO95, VSI96, ML97, SV98, HR98, Mye99], JFlow treats static checking of flow annotations (*label checking*) as an extended form of type checking. Programs written in JFlow can be checked statically by the JFlow compiler, which detects any information leaks through covert storage channels. JFlow is intended to support the writing of secure servers and applets that manipulate sensitive data.

An important philosophical difference between JFlow and other work on statically checking information flow is the focus on a usable programming model, avoiding the unnecessary restrictiveness of earlier systems for static flow analysis. JFlow provides a more practical programming model than earlier work does. The goal of this work is to add enough power to the static checking framework to allow reasonable programs to be written in a natural manner.

Ex. 1002, Page 137

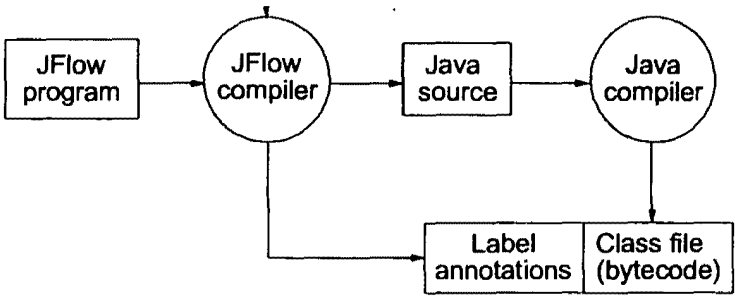


Figure 1.2: JFlow compiler

Adding this power has required several new contributions. Because JFlow extends a complex programming language, it supports many language features that have not been integrated previously with static flow checking, including mutable objects (which are more complex than function values), subclassing, dynamic type tests, access control, and exceptions.

JFlow also provides powerful new features that make information flow checking less restrictive and more convenient than in previous models:

- *Label polymorphism* allows the writing of code that is generic with respect to the security class of the data it manipulates.
- Run-time label checking and first-class label values create a dynamic escape in cases where static checking is too restrictive. Run-time checks are statically checked to ensure that information is not leaked by the success or failure of the run-time check itself.
- Automatic label inference makes it unnecessary to write many of the annotations that would be required otherwise.
- A statically-checked declassification operator allows safe declassification as described by the decentralized label model.

The JFlow compiler is structured as a source-to-source translator; its output is a standard Java program that can be compiled by any Java compiler. The operation of the compiler is depicted in Figure 1.2. The input to the compiler is the text of a JFlow program and the compiled bytecode for any external program modules used by the program. This model of compilation is exactly that of Java. Using this information, the compiler checks JFlow programs and translates them into an equivalent Java program, which is converted to executable form by a standard Java compiler. In addition, the JFlow compiler generates an auxiliary file containing information about label annotations found within the program. This auxiliary file is used in conjunction with the compiled bytecode file whenever this program is used as an external module for the purpose of compiling other code that depends on it, as shown by the dashed arrow.

Ex. 1002, Page 138

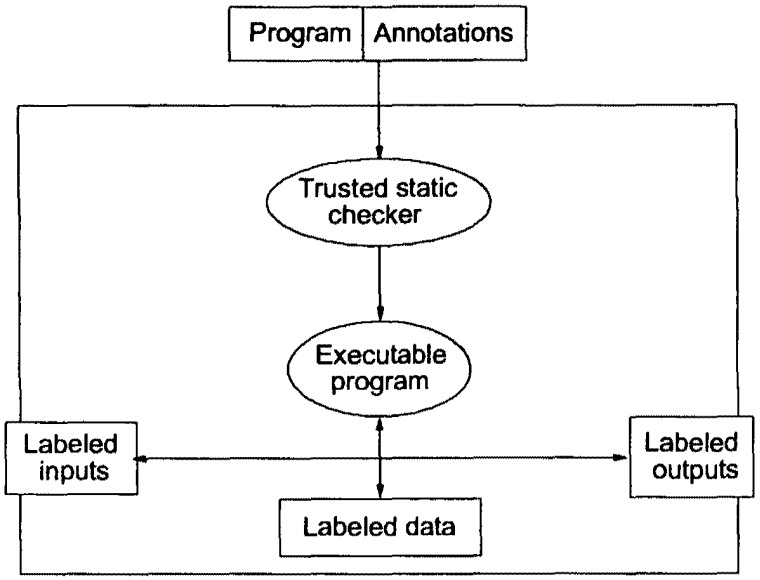


Figure 1.3: Trusted execution platform

For the most part, translation involves removal of the static annotations in the JFlow program (after checking them, of course). For this reason, there is little code space, data space, or run time overhead, because most checking is performed statically.

1.4 Trusted computing base

An important aspect of any security mechanism is the identification of the *trusted computing base* (TCB): the set of hardware and software that must function correctly in order for security to be maintained. In this work, the trusted computing base includes many of the usual trusted components: hardware that has not been subverted, a trustworthy underlying operating system, and a reliable authentication mechanism.

With conventional security mechanisms, all programs are part of the trusted computing base with respect to the protection of privacy, since there is no internal mechanism ensuring that programs respect privacy. For privacy to be protected, it is necessary that programs not transfer information in ways that violate it. In this work, the model is that a static checker rejects programs containing information flows that violate privacy. The static checker may be a compiler that statically checks the information flows in a program and then digitally signs the program, or else a verifier that checks the work of such a compiler.

Together, these trusted components make a trusted execution platform. Figure 1.3 depicts a trusted execution platform, into which code may enter only if it has been checked statically to ensure that it may be trusted to obey the label model. Data in the system is labeled, as are inputs to and outputs from the system.

When this trusted computational environment is constructed from trusted nodes connected by a network, the communication links between the nodes also must be trusted, which can be accomplished through phys-

Ex. 1002, Page 139

ical security or by encrypting and digitally signing communication between nodes. Unrelated third parties are assumed to be unable to violate privacy and integrity by snooping on or subverting channels directly; the question addressed here is how to prevent the intended receiver of an information transfer from violating privacy.

1.5 Applications

The goal of this new information flow control system is to support secure distributed computation, including the following useful applications:

- A node could share information with a downloaded program, yet prevent the mobile code from leaking the information; additionally, the program could be protected from leaking its private information to other programs running on the same node. This kind of security for mobile code would be useful both for clients, which download applet code from servers, and for servers, which upload servlet code and data from clients for remote evaluation.

- Secure servers and other heavily-used applications can be written in programming languages extended with information flow annotations, adding confidence that sensitive information is not revealed to clients of the service through programming errors.

- Trusted parties can provide *secure computation servers* that allow mutually distrusting parties to carry out computations securely and privately, even though neither trusts that the programs of the other will respect its security. This architecture is a solution to the problem that arises when neither party trusts the execution platform of the other, and might be used in the tax preparation example. A trustworthy platform for computation becomes a service with economic value for which the provider might charge.

The annotations used in the JFlow programming language could be used to extend many conventional programming languages, intermediate code (such as Java Virtual Machine bytecode [LY96]), or machine code, where the labeling system defined here makes a good basis for easily checkable security proofs as in proof-carrying code [Nec97]. A good approach to producing proof annotations is for the compiler to generate them as a by-product of static checking; this approach has been shown to work for checkable type-safe machine code [MWCG98], and ought to be applicable to information flow labels as well.

1.6 Limitations

The static analysis techniques developed in Chapters 3 through 5 are intended to control covert and legitimate storage channels. These techniques do does not deal with timing channels, which are harder to control. Because the static analysis is applied to the program being executed, it cannot identify covert channels that do not exist at the level of abstraction presented by the programming language. These covert channels are

Ex. 1002, Page 140

mostly timing channels that are ruled out in a single-threaded system. However, in a multi-threaded system, information may be communicated by covert channels such as cache miss timing. Covert channels of this sort cannot be identified by analysis of a program in source code form, because the source code is at too high a level of abstraction.

1.7 Outline

The remainder of this thesis is structured as follows. Chapter 2 describes the decentralized label model and demonstrates its formal properties. Chapter 3 presents the JFlow programming language, which extends the Java language with support for information flow control. Chapter 4 shows how information flow in the JFlow language can be checked statically through a process similar to type checking, though certain aspects of static checking and source-to-source translation are deferred until Chapter 5. Other security techniques and related work on privacy protection are discussed in Chapter 6. Chapter 7 concludes and offers some thoughts on extensions to this work.

Ex. 1002, Page 141

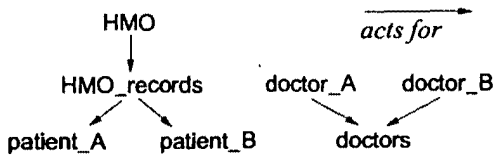


Figure 2.3: The hospital principal hierarchy

except Bob himself. With this label applied to it, tax data cannot be sent to an untrusted network location, represented as an output channel with label {}, because it is not the case that {Bob: Bob} $\sqsubseteq$ {}. Bob can give this data to the WebTax program with reasonable confidence that it cannot be leaked, because WebTax will be unable to remove the {Bob: Bob} policy from the tax data or any data derived from it.

The WebTax program uses Bob's tax data and its private database to compute the tax form. Any intermediate results computed from these data sources will have the label {Bob: Bob; Preparer: Preparer}. Because the reader sets of this label disagree, the label prevents both Bob and Preparer (and everyone else) from reading the intermediate results. This joint label is generated by the rule for join:

$$\{\text{Bob} : \text{Bob}\} \sqcup \{\text{Preparer} : \text{Preparer}\} = \{\text{Bob} : \text{Bob} ; \text{Preparer} : \text{Preparer}\}$$

Preparer is protected by this label against accidental disclosure of its private database through programming errors in the WebTax application.

Before being released to Bob, the final tax form has the same label as the intermediate results, and is not readable by Bob, appropriately. In order to make the tax form readable, the WebTax application *declassifies* the label by removing the {Preparer: Preparer} policy. The application can do this because the Preparer principal has granted the application its authority. This grant of authority is reasonable because Preparer supplied the application and presumably trusts that it will not use the power maliciously.

The authority to act as Preparer need not be possessed by the entire WebTax application, but only by the part that performs the final release of the tax form. By limiting this authority to a small portion of the application, the risk of accidental release of the database is reduced. However, it is important that this part of the application not be exposed as a generally accessible external interface, because this exposure might allow Bob and other parties to misuse the interface to declassify data owned by Preparer.

2.2.2 Hospital example

In this example, there are three parties with privacy concerns: a patient obtaining medical services, a doctor providing the services, and a health maintenance organization (HMO) that serves as an intermediary. There are principals in the system for patients, *e.g.*, patient_A, and doctors, *e.g.*, doctor_B; additionally, all doctors can act for a principal doctors that represents the group of doctors within the HMO. Two HMO principals also exist: HMO, representing maximum authority within the HMO, and HMO_records, representing authority over the record-keeping functions of the HMO; HMO can act for HMO_records, and HMO_records

Ex. 1002, Page 142

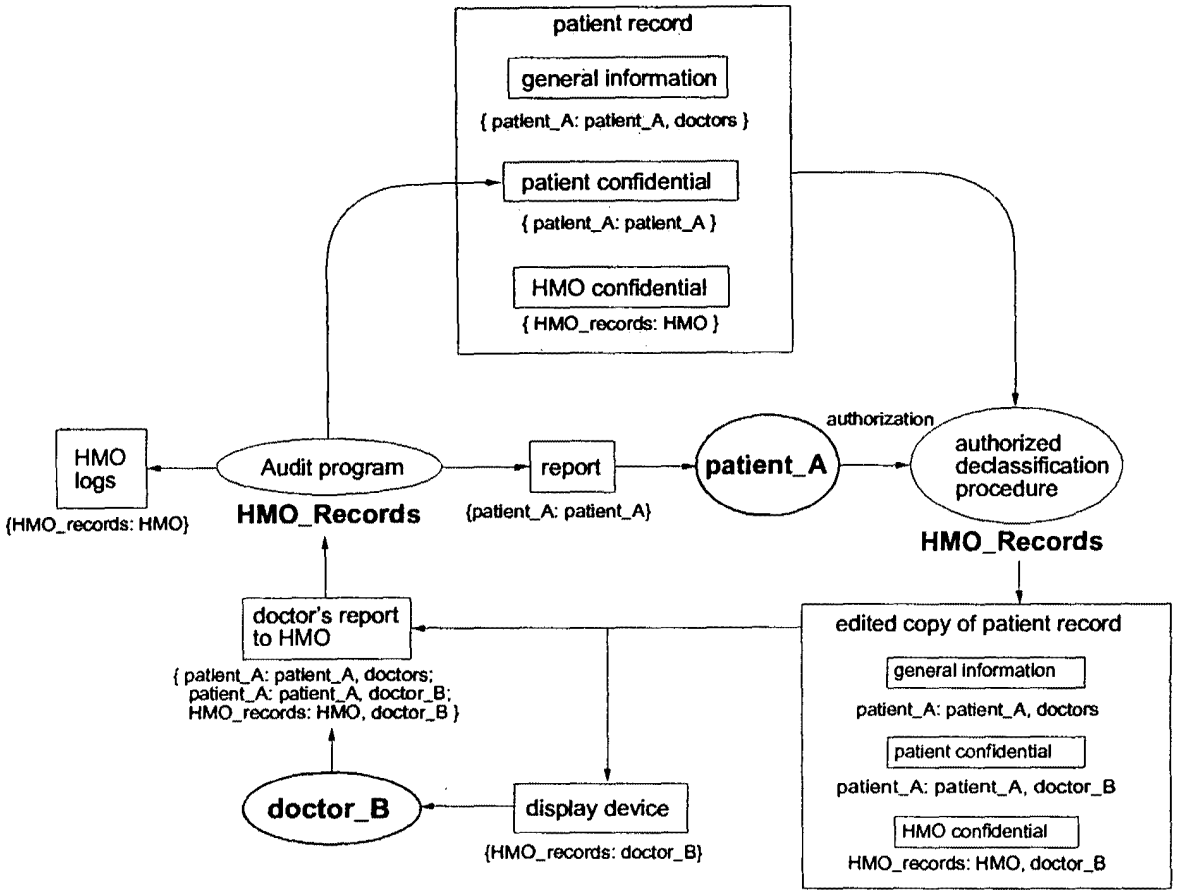


Figure 2.4: The hospital example

can act for patients: each patient must trust the HMO to keep track of its records. The resulting principal hierarchy is shown in Figure 2.3.

Figure 2.4 shows the hospital example, which shows how information flows as the patient receives medical services. The HMO maintains the patient's medical history, which has three parts: general information, which is controlled by the patient but is readable by any doctor, private information (such as the medical history of the patient), which is normally not readable by doctors, and confidential information that the HMO does not release to patients.

The first step in a patient/doctor interaction is for the doctor to obtain a copy of the patient's record. The record is declassified so that the doctor can read it; this can happen only with the authorization of the patient. The patient, represented in the diagram by the dark oval labeled patient_A, makes an authenticated request to an existing program running with the authority of HMO_records; this program uses the patient's authority to provide the doctor with an edited version of the patient's private information and of the HMO's confidential information.

The doctor is represented by the dark oval labeled doctor_B. To read the information, the doctor requires

an output channel to a display device with the single reader, doctor_B. This display device is certified by HMO_records as a secure device that only doctor_B is reading from. In principle, all of the information in the patient records should be safe to write to this display device, though the subset relabeling rule will not permit it. Thus, this example motivates the development of a better relabeling rule, which is developed in the following sections. Writing the information to the display device is safe because HMO_records can act for all of the owners of the data in the patient records (patient_A and HMO_records), so its certification should be good enough. In addition, various parts of the patient record are released to doctors or doctor_B, and the actual reader, doctor_B, can act for both these principals. Note that the patient information cannot be written to a channel that has any readers other than doctor_B, and that there is no way the doctor can declassify the patient information.

Eventually, the doctor sends a report to the HMO of services rendered. In addition to the comments of the doctor, the report contain information from all three components of the patient's record, so it acquires a joint label reflecting all these sources. Note that the general patient information does not explicitly permit doctor_B as a reader. Using the subset relabeling rule, the first policy owned by patient_A in the resulting joint label prevents the doctor from reading his own report. This example of unnecessary restrictiveness also arises from the subset relabeling rule and is fixed by the more flexible relabeling rule developed later.

The audit program runs with the authority of the HMO_records principal and thus can store the information with the appropriate labels both in the log and in the patient record database. It can also send a report to the patient; as in the tax preparer example, the designer of the audit program must use mechanisms outside the scope of information flow control to determine either that no HMO-confidential information is leaked or that the leak is acceptably small.

2.3 Extending and interpreting labels

The hospital example presented in the previous section shows that the basic model is not powerful enough, and a more permissive relabeling rule is needed that takes the principal hierarchy into account. This section formalizes the notions of labels and principal hierarchies and then defines a condition for judging whether a relabeling rule is correct.

2.3.1 Limitations of the subset relabeling rule

One way to think about whether a relabeling rule is safe is by considering *incremental relabelings* that can make a label more restrictive, or leave it equally restrictive. The relabeling rules discussed in this thesis can be understood in terms of the incremental relabelings they allow. For example, the subset relabeling rule allows the following two kinds of *incremental relabelings*, which make a label more restrictive (or possibly have no effect).

- **Removing a reader**. Removing a reader from a policy will restrict the propagation of the labeled data further, if it has any effect at all.

Ex. 1002, Page 144

Chapter 6

Related Work

Most of this thesis has been concerned with the problem of protecting the secrecy of data. This problem has been recognized for at least 25 years, and also has been referred to as *confinement* [Lam73] of data, or *confidentiality*. In this thesis, it has been referred to as protecting *privacy*, since the goal is to protect data owned by mutually distrusting principals, rather than the secret data of a single entity such as the government. A great deal of work has gone into addressing the problem of secrecy, and it is not feasible to enumerate all of it. This chapter summarizes previous work done on various kinds of security techniques that relate to this work, particularly focusing on information flow control.

6.1 Access control

Most systems protect privacy and integrity through *discretionary access control*, or what is usually called simply *access control*. The idea of access control is that before a potentially dangerous action may be taken by a computer program, a run-time test is made to ensure that the program has been granted the necessary authority for the action. Many access control mechanisms have been designed, such as *capabilities* [DV66, WCC+74], *access control lists* [Lam71], and various hybrid schemes (*e.g.*, [RSC92]). Actions that do not conform to stated policies are not permitted, whether they are reads, writes, or higher-level operations. Unix file permissions are an example of a simple, well-known access-control mechanism.

Since JFlow provides a simple mechanism for controlling the privileges of a program, in the form of static authority, it is interesting to compare it to existing Java access control models, based on *stack inspection* [WBF97, WF98]. Current versions of the Java run-time environment provided by Netscape, Microsoft, and Sun implement variants of this model [Net97, Mic97, GS98]. In Java, privileges are needed to perform various unsafe operations, such as accesses to the local filesystem. In the stack inspection approach, these privileges are known as *targets*. Each class can be authorized to claim one or more privileges, but by default, the class code does not possess these privileges. Explicit operations are provided for enabling and disabling privileges. When a privilege is needed in order to perform an unsafe operation, the stack leading up from the point of invocation is inspected at run time. Every class whose code is on the stack, up to the point where the needed privilege was enabled, must be authorized to claim that privilege. This model allows a class to

Ex. 1002, Page 145

grant a privilege, but only if it has itself enabled the privilege explicitly. The privilege can be granted only to the code of another trusted class that could have claimed the privilege for itself. Thus, privilege is enabled *explicitly*, but granted *implicitly*, by the act of calling another method while the privilege has been enabled.

This set of design choices differs in several respects from those in JFlow. In JFlow, principals may be used to represent targets as well as users. The authority clause of a class gives a class the power to act for the named principals, but individual methods do not possess the corresponding privilege unless they explicitly declare it. Thus, the models are similar in that privileges are not available unless explicitly declared. In JFlow, authority is granted to a called method *explicitly*: it is passed as an argument of type principal that is present in a caller clause of the called method. Unlike in Java, the called method need not have the potential authority of that principal (*i.e.*, target). The stated reason for preventing this in the Java models is that it defeats *luring attacks* in which the authority granted is misused by the called method. Luring attacks are a greater concern in the Java model, since authority is granted implicitly. In JFlow, it is clear what authority is granted to the called method (although it may be a run-time parameter). JFlow also allows authority to be bound into an object in a parametric fashion; a class can require that its constructors be called from a site possessing the authority of its principal parameters; this authority is bound into the object. An obvious difference between the models is the manner in which they are enforced. The JFlow authority mechanisms are largely statically checked (though there is support for dynamic checking), whereas the Java model is checked entirely dynamically, with consequent run-time overhead. Static checking is possible in JFlow because authority transfers are completely explicit. Since the Java model of access control is largely a subset of that in JFlow, it seems likely that it could be enforced at load time by an extended Java Virtual Machine if class files were extended with explicit annotations about granted authority.

6.2 Limitations of discretionary access control

Discretionary access control does not support privacy well, because although it prevents information release, it does not control information propagation. For example, consider the tax preparer example of Section 1.1, reproduced here in Figure 6.1. In this example, Bob is preparing his tax form using a piece of software called "WebTax". Bob would like to be able to prepare his final tax form using WebTax, but without trusting WebTax to protect his privacy. Bob can impose an access check that determines whether Preparer can see his tax data. However, once the access is allowed, Bob cannot control how Preparer distributes the information it has read. He is forced to trust that the WebTax program will respect his privacy correctly. Thus, discretionary access protects the privacy of data against others, but it is vulnerable to Trojan horse programs.

Everything that has just been said about privacy applies to integrity as well. If program *A* allows program *B* to modify *A*'s data, then *A* has controlled who may write the data, but cannot control how *B* obtains the data to write there. With only discretionary access control, *A* must trust not only *B* but every program that might have affected the data *B* is providing. Discretionary access control is a point-of-sale mechanism that cannot

Ex. 1002, Page 146

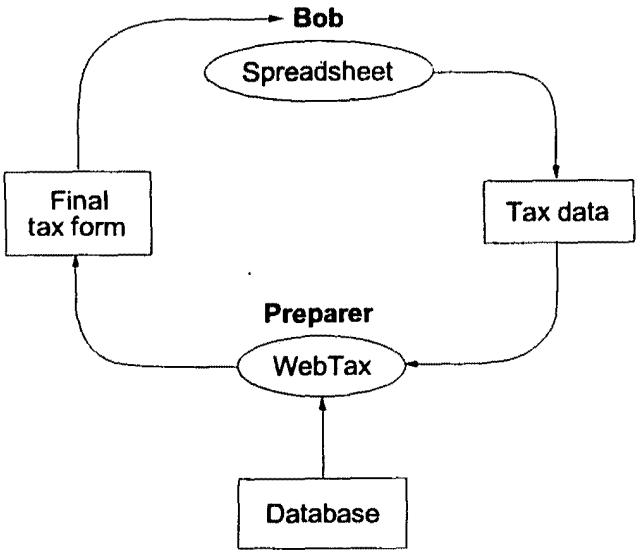


Figure 6.1: Tax preparer example

control either the propagation of information after its release or the propagation of information leading to an update.

6.3 Information flow control

In the case of both privacy and integrity, what is wanted is a way to extend access restrictions transitively, arbitrarily far from the point where data is released or updated. This transitive extension is not possible in a conventional discretionary access control system, because the decision about whether to transfer information from program *A* to program *B* is made based upon the authority and privileges possessed by *A* and *B*; restrictions that the data's ultimate source or destination might like to apply cannot be enforced reliably because information about these restrictions in general has been lost. This insight leads to *information flow control* and *mandatory access control* models, which apply sensitivity labels to data. These labels propagate with the data and are used to mediate information transfers within and between programs. Restrictions on the use of data propagate with the data and apply to any data derived from it. Privacy restrictions prevent data from being seen by untrusted users; integrity restrictions prevent untrusted data from affecting storage locations. A good overview of information flow control is presented by Denning [Den82].

The original model of information flow for secrecy comes from the early work of Bell and LaPadula [BL75]. In this work, objects in the system are assigned to security classes from a small ordered set (e.g., unclassified, classified, secret). Information can flow between the partitions only by moving upward in security class. A subject, or process, in the system is assigned a security class, and the data it manipulates is assigned the same security class. It can read data from a subject of the same or lower security class. The Bell-LaPadula model supports privacy through information flow control; it also controls writes through

access control. Non-destructive writes are permitted to an object of a higher security class, but destructive writes are permitted only to objects of the same security class. This rule prevents low-level subjects from overwriting high-level data, even though this overwriting would not cause an information leak.

The most common information flow enforcement mechanism is dynamic. Fenton's Data Mark Machine (DMM), an early abstract model for information flow enforcement [Fen73, Fen74], is a good example of the dynamic approach to fine-grained information flow. As a program computes, sensitivity labels (security classes) are associated with all data values. The sensitivity label of a computed value must be at least as restrictive as the sensitivity labels of the values it was computed from. In the DMM model, the program-counter label pc is maintained at run time. One weakness of the DMM model is its inability to deal with implicit flows precisely. After an if statement, pc does not revert to its former value, unlike in JFlow. Data computed after a conditional becomes excessively restrictively labeled. The DMM model is made workable because the pc is unaffected by a function call, but at the cost that exceptions are not supported. JFlow allows the program-counter label to revert if the method can terminate only normally, but also allows fine-grained tracking of information communicated through exceptions.

The DoD Orange book requires a dynamic mechanism for enforcing *mandatory access control* (MAC) for secure systems of class B1 and higher [DOD85]. In this approach, a fixed label is associated with the currently running process. As in the Bell-LaPadula model, a process may read only from objects with a label that is of the same or lower level than its own. However, it may write to an object with an equal or higher-security label. The Orange Book specifies that in systems with mandatory access control, information can leak only by leaving the system through *channels*. There are two kinds of channels: single-level channels, which have a single fixed label against which all data is dynamically tested before transmission; and multi-level channels, which allow arbitrarily labeled data to be transmitted, but also dynamically transmit the label of the data along with it.

The JFlow language provides both static and dynamic enforcement of information flow, with an emphasis on making static enforcement as expressive as possible. However, the dynamic enforcement features of mandatory access control can be simulated in JFlow by using run-time labels and run-time principals. Channels in the decentralized label model are single-level channels; however, multi-level channels can be simulated by transmitting values of the type Protected, which encapsulates a value with its label. JFlow also provides fine-grained tracking of information labels. With mandatory access control, a process is irrevocably tainted by the label of data it has observed, and therefore passes the label on to all data it touches afterward, making that data unnecessarily restrictive. This approach is necessary with purely dynamic enforcement in order to prevent implicit flows. The fine-grained static analysis in JFlow allows implicit flows to be prevented while avoiding many unnecessary restrictions.

There has been considerable work on developing richer and more expressive models for labeling data. Denning extended and clarified the Bell-LaPadula label model with the notion of a *lattice* of security classes [Den75, Den76]. As in the model defined in this thesis, information may be relabeled upward in the lattice, and information derived from multiple sources acquires a label (security class) that is the join of

Ex. 1002, Page 148

the labels of the sources. The decentralized label model does not quite fit into Denning's lattice structure, although it retains the essential properties. One obvious difference is that the decentralized label model supports a limited form of declassification. The label system looks different to each principal; every principal shares a common set of safe relabelings, but has access to its own *declassification relabelings*. Relabeling in the decentralized label model defines an ordering relation ($\sqsubseteq$), as in Denning's model, but it is not a partial order, since two labels may be equivalent without being equal. However, it does support the lattice operations of join ($\sqcup$) and meet ($\sqcap$) on equivalence classes of labels, and these operations distribute over each other.

Denning's lattice framework was instantiated by Feiertag et al. [FLR77] in *multilevel security policies*. A multilevel security policy is a pair (A, C), where A is a *hierarchical security class*, and C is a set of *categories*. Hierarchical security classes form a totally ordered set like that of the Bell-LaPadula model; categories are arbitrary symbols. One multilevel security policy (A_1, C_1) can be relabeled to another, (A_2, C_2), as long as $A_1 \sqsubseteq A_2$ and $C_1 \subseteq C_2$. Categories operate in the reverse direction one might expect: it is acceptable to increase the set of categories but not to decrease them. They provide a notion of the owners of the data rather than of potential readers of the data.

Multilevel security policies are a common underlying model used with mandatory access control systems. However, they can be modeled straightforwardly within the decentralized label model by introducing principals to represent each of the hierarchical security classes and each of the possible categories. The principals representing security classes have the corresponding acts-for relations: the principal representing top secret can act for the principal representing secret, and so on. A multilevel policy $(A, \{c_1 \ldots c_n\})$ is translated to a decentralized label $\{A : \; ; c_1 : \; ; \ldots ; c_n : \}$; the complete relabeling rule then enforces exactly the relabeling rule for multilevel policies. Users are given security classifications by introducing acts-for relations between their principals and the appropriate A and c_i principals; the output channel to a user p can be labeled root $: p$ (where root is a highly trusted principal) and the relabeling rule will enforce the appropriate restriction. One weakness of this translation is that it allows the user p to declassify all the data he can read; this flaw can be fixed using the approach of Section 2.6.3.

Biba showed that information flow control can be used to enforce *integrity* as well as secrecy, and that integrity is a dual of secrecy [Bib77]; this insight has been employed in several subsequent systems, and also applies to the decentralized integrity policies described in Section 2.6.1. IX [MR92] is a good example of a real-world information flow control system that implements MAC and supports both secrecy and integrity policies simultaneously.

More recent work on label models has not been as widely adopted. One popular theme has been models for commercial applications that capture conflicts of interest and allow non-transitive flow policies [CW87, BN89, TW89, Fol91]. The Chinese Wall policy of Brewer and Nash [BN89] has been the subject of some study. The idea behind this policy is that information labels should be able to enforce separation of duties. For example, a bank might maintain a separation between its accounts and investments departments. An employee who is supposed to handle the investments of the bank should not have access

Ex. 1002, Page 149

to information about customer accounts, and vice versa. However, Sandhu has argued that the Chinese Wall policy can be implemented according to a standard lattice-based labeling policy by properly distinguishing users and programs [San92]. In the decentralized label model, this separation of duties can be enforced through restrictions on the principal hierarchy rather than through labels. The group principals accounts and investments are introduced, and employee principals are prohibited from belonging to both groups. This structure is arguably more intuitive, since the separation of duties is built into the principals themselves, rather than into the labels of individual pieces of data. More recent work on modeling separation of duties has taken a similar approach of mapping user and duties into a role hierarchy [GGF98].

The decentralized label model has several similarities to the ORAC model of McCollum et al. [MMN90]: both models provide some approximation of the "originator-controlled release" labeling used by the U.S. DoD/Intelligence community. The ORAC model was developed because of the observation that conventional MAC and DAC policies do not adequately support this kind of security policy. Both ORAC and the decentralized label model have the key concept of *ownership* of policies. Both models also support the joining of labels as computation occurs, though the ORAC model lacks some important lattice properties since it attempts to merge policies with common owners. In the ORAC model, as in some mandatory access control models, both process labels and object labels can float upward in the label lattice arbitrarily, a phenomenon called *label creep* that leads to excessively restrictive labels. The absence of lattice properties and the dynamic binding of labels to objects and processes makes any static analysis of the ORAC model rather difficult. Interestingly, ORAC does allow owners to be replaced in label components (based on ACL checks that are analogous to acts-for checks), but it does not support extension of the reader set. The ORAC model also does not support any form of declassification.

All practical information flow control systems provide the ability to *declassify* or *downgrade* data because strict information flow control is too restrictive for writing real applications. More complex mechanisms such as *inference controls* [Den82, SS98] often are used to decide when declassification is appropriate. Declassification in these systems lies outside the label model, so declassification is performed by a *trusted subject*: code with the authority of a highly trusted principal. A recent variant of this approach by Ferrari et. al [FSBJ97] introduces a form of dynamically-checked declassification through special *waivers* to strict flow checking. Some of the need for declassification in their framework would be avoided with fine-grained static analysis. Because waivers are applied dynamically and mention specific data objects, they seem likely to have administrative and run-time overheads. One key advantage of the new label structure is that it is *decentralized*: unlike in the trusted subject approach, other principals in the system need not trust the declassification decision of a principal p, since p cannot weaken the policies of principals that it does not act for.

Previous information flow techniques do not deal well with situations of mutual distrust. These techniques were originally designed to protect the privacy and integrity of data owned by a single principal—typically, the government. If one considers privacy and integrity in a more decentralized setting, such as the community of Web users, it is clear that no universal notion of secret sensitivity can be established.

Ex. 1002, Page 150

No label including a hierarchical security class can be acceptable in a decentralized environment. Even schemes containing a generalized lattice of labels do not solve the problem of mutual distrust. Consider the tax preparation example in a lattice-based MAC system. Unless Bob can act for Preparer or vice-versa, the final tax form in this example will be labeled so that neither Bob nor Preparer are able to read it—a result that is safe but not very useful.

JFlow provides a programming model that integrates information flow control and a simple model of access control. Stoughton [Sto81] developed a purely dynamic model integrating both access control and information flow control, defined formally using denotational semantics. This model does not seem to have been implemented. In the model, objects have both a *current access level* and a *potential access level*. The potential access level is used to enforce information flow constraints as in mandatory access control systems. The current access level is used to enforce discretionary access control; it can be relaxed by an appropriately trusted principal, but only to the point where it is as restrictive as the potential access level. To relax it further would violate information flow control. Thus, this model does not support declassification. Because this model is purely dynamic, it also does not treat implicit flows securely. The model of access control is particularly simple; it mediates accesses at the level of reads and writes to objects, and does not provide the ability to control higher-level operations.

6.4 Static enforcement of security policies

JFlow is unusual not only in integrating information flow control and access control, but also in providing both static and dynamic enforcement of these mechanisms. Most prior security work has focused on dynamic enforcement, but there has been some earlier work on static enforcement of access control.

Jones and Liskov defined a system for statically enforcing discretionary access control through a scheme of restricted types, in which some methods were marked as inaccessible [JL78]. Their rules define a form of subtyping, with security guaranteed by the inability to cast downward in the type hierarchy dynamically. However, the lack of any capability for dynamically enforcing access control checks makes this scheme impractical.

The CACL model of access control [RSC92] has a model of mixed static and dynamic enforcement of access control that is more practical. As in the Jones and Liskov model, references to objects may have a type in which certain methods are inaccessible. However, when objects cross protection domains, new copies of the references are constructed for which method accessibility is recomputed lazily. In JFlow, methods can be called only if all of their caller constraints are satisfied. When objects are passed between different trust domains, method accessibility changes automatically based on static reasoning about authority; no rewriting is needed.

Static analysis was applied to information flow control early on by Denning and Denning [DD77], but has not been adopted widely since because of its limitations. Static checking allows the fine-grained tracking of sensitivity and integrity labels through program computations, without the run-time overhead of dynamic

Ex. 1002, Page 151

security classes. Because this approach inspects entire programs, it has a significant advantage over simple dynamic checking: a program can be checked to determine that no possible execution results in a security policy violation. However, dynamic checking is needed for some programming examples, and previous static checking techniques did not integrate dynamic checking, making them impractical. Earlier static checking techniques did not handle exceptions, either.

Another approach to checking programs for information flows statically has been automatic or semi-automatic theorem proving. Researchers at MITRE [Mil76, Mil81] and SRI [Fei80] developed techniques for information flow checking using formal specifications. Feiertag [Fei80] developed a tool for automatically checking these specifications using a Boyer-Moore theorem prover.

Recently, there has been more interest in provably-secure programming languages, treating information flow checks in the domain of type checking, which does not require a theorem prover. Palsberg and Ørbæk have developed a simple type system for checking integrity [PO95]. Volpano, Smith and Irvine have taken a similar approach to static analysis of secrecy, encoding Denning's rules in a functional type system and showing them to be sound using standard programming language techniques [VSI96, Vol97]. Also, Abadi [Aba97] has examined the problem of achieving secrecy in security protocols, also using typing rules, and has shown that encryption can be treated as a form of safe declassification through a primitive encryption operator.

Heintze and Riecke [HR98] have shown that information-flow-like labels can be applied to a simple language with reference types (the SLam calculus). They show how to statically check an integrated model that provides access control, information flow control, and integrity. Their model is similar to Stoughton's earlier, dynamic model; labels include two components: one that enforces conventional access control, and another that enforces information flow control. Their model inherits some limitations of Stoughton's model.

The models of Smith, Volpano, and Irvine and of Heintze and Riecke have the limitation that they are entirely static: unlike JFlow, they have no run-time access control, no declassification, and no run-time flow checking. These models also do not provide label polymorphism or support for objects. Addition of these features is important for supporting a realistic programming model, though it does make the programming language more difficult to treat with the conventional tools of programming language theory. Heintze and Riecke do prove some useful soundness theorems for their model. This step would be desirable for JFlow, but the various language extensions make formal proofs of correctness difficult at this point.

6.5 Modeling principals and roles

The notion of a principal hierarchy, used in the decentralized label model, is similar to several other models for modeling roles. The acts-for relation is similar to the *speaks-for* relation that is introduced by Lampson et al. [LABW91] for describing authentication in a distributed system. In that model, a notion of *compound principals* is introduced; a compound principal is an expression such as Bob as manager, where Bob is an ordinary principal, and manager is a role. The decentralized label model does not provide this much

structure; however, a compound principal can be modeled as a third principal for which Bob acts, and which acts for manager.

Some work on *role-based access control* has also introduced notions of a *role hierarchy* based on various kinds of dominance relations among principals and roles [FK92, SCFY96]. This structure is used to model the assignment of users to groups and to roles, similarly to the decentralized label model. Roles have also been used as security classes in an information flow model [San96]. However, because this model does not distinguish between roles and information flow labels, information can flow only upward in the role hierarchy.

6.6 Cryptography

In the minds of many people, computer security is associated with encryption. It is reasonable to ask how cryptographic techniques are related to this work. Encryption can be used to achieve some important security goals that are subsidiary to protecting privacy and integrity, and much recent computer security research has focused on this use. One such goal is *authentication*: the reliable identification of *who* is requesting that an action be performed [Lam71, LABW91, ABLP93]. Many computer systems use password checking to authenticate their users. However, in a distributed system, some form of encryption is generally needed to perform authentication securely. Reliable authentication is a prerequisite for protecting privacy and integrity. For example, any access control mechanism requires an underlying authentication mechanism so that one can be sure that a process does possess the granted authority that claims to.

Another important feature of a secure system is reliable information channels that cannot be subverted by unrelated third parties. Encryption protects privacy by preventing these channels from having their information extracted; digital signatures protect integrity by preventing new material from being inserted onto the channel by a third party to fool the receiver.

The encryption technology for reliable authentication and secure channels has been researched heavily and also is widely available, in systems like Kerberos [SNS88] and ssh [Ylo96]. Encryption provides a rather elemental protection for privacy and integrity. The work presented herein makes the assumption that these technologies are available as a standard component, and builds on them.

6.7 Covert channels

This work has ignored covert channels arising from time measurement and thread communication. These channels have long been recognized as very difficult to control [Lam73]. A scheme for statically analyzing thread communication has been proposed [Rei79, AR80]; essentially, a second pc is added with different propagation rules. A local pc handles information flow within a thread; the global pc restricts operations that communicate with other threads. Stoughton's model [Sto81] also uses this local/global approach. The same technique can be used to control timing channels. This approach could be applied to JFlow and even

Ex. 1002, Page 153

checked statically, similarly to static side-effect and region analysis [JG91], which aims to infer all possible side-effects caused by a piece of code. However, it is not clear how well this scheme works in practice; it seems likely to restrict timing and communication quite severely, particularly if applied directly to a programming model in which objects are shared between threads. In such a programming model, all object modifications are potentially asynchronous communications with other threads, and will be highly restricted if limited by a pc that is shared across all threads. Smith and Volpano have developed rules recently for checking information flow in a multithreaded functional language [SV98]. As might be expected, the rules they define prevent the run time of a program from depending in any way on non-public data, which is arguably impractical.

Chapter 7

Conclusions

Protecting privacy and secrecy of data has long been known to be a very difficult problem. The increasing use of untrusted programs in decentralized environments with mutual distrust makes a solution to this problem both more important and more difficult to solve. Existing security techniques do not provide satisfactory solutions to this problem.

The goal of this work is to make information flow control a viable technique for providing privacy in a complex, decentralized world with mutually distrusting principals. Information flow control is an attractive approach to protecting the privacy (and integrity) of data because it allows security requirements to be extended transitively towards or away from the principals whose security is being protected. However, it has not been a widely accepted technique because of the excessive restrictiveness it imposes and the computational overhead.

To address these limitations of conventional information flow techniques, this work focuses on two areas. First, a new model of decentralized information flow labels provides the ability to express privacy policies for multiple, mutually distrusting principals, and to enforce all of their security requirements simultaneously. Second, the new language JFlow permits static checking of decentralized information flow annotations. JFlow seems to be the most practical programming language yet that allows this checking.

7.1 Decentralized label model

The decentralized label model described in Chapter 2 makes information flow more practical by removing some of the unnecessary restrictiveness of earlier models. It provides considerable flexibility by allowing individual principals to attach flow policies to individual values manipulated by a program. It also incorporates a notion of principal hierarchy that allows these policies to be expressed in terms of and on behalf of more complex authority entities such as groups and roles.

Practical information flow systems require some ability to declassify or downgrade data. Since the policies in decentralized labels have a notion of ownership, the owner can be allowed to declassify policies that it owns. This declassification is safe because it does not affect the secrecy guarantees to other principals who have an interest in the secrecy of the data. The owner may use reasoning processes such as information

theory techniques or inference controls to determine that the information leaked through declassification is acceptably small, but other principals in the system do not need to trust these reasoning processes. This support for decentralized declassification makes the label model ideal for a system containing mutually distrusting principals.

An important feature of the decentralized label model is the formal semantics that are defined for the model, and the relabeling rule that was shown to be both sound and complete with respect to this formal semantics. The relabeling rule precisely captures all the legal relabelings that are allowed when knowledge about the principal hierarchy is available statically, and has the necessary lattice properties to support static checking and automatic label inference. Because the complete relabeling rule is as permissive as possible without being unsafe, it is easier to model common security paradigms, allowing control of information flow in a system with group or role principals. Examples in Chapter 2 showed that the expressive power of the complete relabeling rule was helpful in modeling reasonable application scenarios without resorting to declassification.

Extensions to the basic model discussed in Chapter 2 also show that integrity [Bib77] constraints have a natural lattice structure, and decentralized integrity policies can also be expressed conveniently in the same framework, with rules precisely dual to those of decentralized privacy policies. In addition, labels that combine integrity and privacy constraints can be expressed, with straightforward rules. Finally, extensions to the principal hierarchy model allow more expressive modeling of group and role principals.

7.2 Static analysis of information flow

Information flow control is usually enforced dynamically, causing substantial loss of performance and also difficulty in handling implicit information flows. Static program checking appears to be the only enforcement technique that can control information flows with reasonable efficiency and precision, although it cannot identify certain covert channels. However, previous static analysis techniques have not been shown to be practical.

Chapters 3–5 describe the new language JFlow, which extends the Java language to permit simple static checking of flow annotations. The goal of this work is to add enough power to the static checking framework to allow reasonable programs to be written in a natural manner. JFlow addresses many of the limitations of previous work in this area. It supports many language features that previously have not been integrated with static flow checking, including mutable objects (which subsume function values), subclassing, dynamic type tests, dynamic access control, and exceptions.

Avoiding unnecessary restrictiveness while supporting a complex language has required the addition of sophisticated language mechanisms: implicit and explicit polymorphism, so that code can be written in a generic fashion; dependent types, to allow dynamic label checking when static label checking would be too restrictive; static reasoning about access control; statically-checked declassification. Making the programming language convenient has also involved automatic label inference, as described in Chapter 5.

Ex. 1002, Page 156

This list of mechanisms suggests that one reason why static flow checking has not been accepted widely as a security technique, despite having been invented over two decades ago, is that programming language techniques and type theory were not then sophisticated enough to support a sound, practical programming model. By adapting these techniques, JFlow makes a useful step towards usable static flow checking.

7.3 Future work

There are several directions for extending this work. One obviously important direction is to continue to make it a more practical system for writing applications. JFlow addresses many of the limitations of earlier information flow systems that have prevented their use for the development of reasonable applications; however, more experience is needed to better understand the practical applications of this approach.

One direction for exploration is the development of secure run-time libraries written in JFlow that support JFlow applications. Features of JFlow such as polymorphism and hybrid static/dynamic checking should make it possible to write such libraries in a generic and reusable fashion. One interesting possibility is the development of a secure user interface library that provides event distribution and rendering capabilities available in user interface toolkits. This library should include user interface widgets that support information flow control directly; for example, a type-in that reliably notifies the user of what security policy is applied to data entered into it.

It should also be possible to augment the Java Virtual Machine [LY96] with annotations similar to those used in JFlow source code. The bytecode verifier would check both types and labels at the time that code is downloaded into the system. Other recent work [LY96, Nec97, MWCG98] has shown that type checking performed at compile time can be transformed into machine-code or bytecode annotations. The code can then be transmitted along with the annotations, and the two checked by their receiver to ensure that the machine code obeys the constraints established at compile time. This approach also should be applicable to information flow annotations that are expressible as a kind of type system.

The JFlow language contains relatively complex features such as objects, inheritance and dependent types, and these features have made it difficult thus far to use theoretical programming-language techniques to show that the static checking rules of Chapter 4 are sound. However, this demonstration is important for widespread acceptance of a language for secure computation.

This work has assumed an entirely trusted execution environment. The model described here does not work well in large, networked systems in which different principals may have different levels of trust in the various hosts in the network. One simple technique for dealing with distrusted nodes is to transmit opaque receipts or tokens for the data. Another approach is for a third party to provide a trusted host to get around the impasse of mutually distrusted hosts. It would be interesting to investigate a distributed computational environment in which secure computation is made transparent through the automatic application of these techniques.

This work shows how to control several kinds of information flow channels better, including channels

Ex. 1002, Page 157

through storage, implicit flows, and run-time security checks. However, covert channels that arise from timing channels and from the timing of asynchronous communication between threads are not treated in this thesis, by ruling out timing and multi-threaded code. Supporting multi-threaded applications would make this work more widely applicable. Although there has been work on analyzing these channels through static analysis [SV98, HR98], the current techniques are restrictive. One central difficulty is the need to distinguish between locally and globally visible operations within a multi-threaded program. Current multi-threaded programming environments have tended to minimize this distinction, but without it, static analysis will not be a reasonably precise tool for controlling information flow. An altered programming model may be possible in which enough information is available about inter-thread communication to permit precise analysis.

This thesis has provided new models and techniques for protecting privacy. Providing better protection of privacy is a challenging and important problem for future computing environments. These environments are likely to be large and distributed, and to contain distrusted users, programs, and hosts. This problem has not received as much attention recently as it merits, and I hope that the contributions of this thesis will serve as a fresh impetus to its further consideration.

Ex. 1002, Page 158

Bibliography

[Aba97] Martín Abadi. Secrecy by typing in security protocols. In *Proc. Theoretical Aspects of Computer Software: Third International Conference*, September 1997.

[ABLP93] Martín Abadi, Michael Burrows, Butler W. Lampson, and Gordon D. Plotkin. A calculus for access control in distributed systems. *TOPLAS*, 15(4):706–734, 1993.

[AC96] Martín Abadi and Luca Cardelli. *A Theory of Objects*. Monographs in Computer Science. Springer-Verlag, New York, 1996.

[ACPP91] Martín Abadi, Luca Cardelli, Benjamin C. Pierce, and Gordon D. Plotkin. Dynamic typing in a statically typed language. *ACM Transactions on Programming Languages and Systems (TOPLAS)*, 13(2):237–268, April 1991. Also appeared as SRC Research Report 47.

[AR80] Gregory R. Andrews and Richard P. Reitman. An axiomatic approach to information flow in programs. *ACM Transactions on Programming Languages and Systems*, 2(1):56–76, 1980.

[Bib77] K. J. Biba. Integrity considerations for secure computer systems. Technical Report ESD-TR-76-372, USAF Electronic Systems Division, Bedford, MA, April 1977.

[BL75] D. E. Bell and L. J. LaPadula. Secure computer system: Unified exposition and Multics interpretation. Technical Report ESD-TR-75-306, MITRE Corp. MTR-2997, Bedford, MA, 1975. Available as NTIS AD-A023 588.

[BN89] D. F. Brewer and J. Nash. The Chinese Wall security policy. In *Proc. IEEE Symposium on Security and Privacy*, pages 206–258, May 1989.

[Car91] Luca Cardelli. Typeful programming. In E. J. Neuhold and M. Paul, editors, *Formal Description of Programming Concepts*. Springer-Verlag, 1991. An earlier version appeared as DEC Systems Research Center Research Report #45, February 1989.

[CLR90] Thomas A. Cormen, Charles E. Leiserson, and Ronald L. Rivest. *Introduction to Algorithms*. MIT Press, 1990.

[CW87] David Clark and David R. Wilson. A comparison of commercial and military computer security policies. In *Proc. IEEE Symposium on Security and Privacy*, pages 184–194, 1987.

[DD77] Dorothy E. Denning and Peter J. Denning. Certification of Programs for Secure Information Flow. *Comm. of the ACM*, 20(7):504–513, July 1977.

[DE97] Sophia Drossopoulou and Susan Eisenbach. Java is type safe – probably. In *Proceedings of Object-Oriented Programming, 11th European Conference (ECOOP 1997)*, pages 389–418, Jyväskylä, Finland, June 1997. Lecture Notes in Computer Science, Vol. 1241, Springer, 1997.

Ex. 1002, Page 159

[Den75] Dorothy E. Denning. *Secure Information Flow in Computer Systems*. PhD thesis, Purdue University, W. Lafayette, Indiana, USA, May 1975.

[Den76] Dorothy E. Denning. A lattice model of secure information flow. *Comm. of the ACM*, 19(5):236–243, 1976.

[Den82] Dorothy E. Denning. *Cryptography and Data Security*. Addison-Wesley, Reading, Massachusetts, 1982.

[DG84] William F. Dowling and Jean H. Gallier. Linear-time algorithms for testing the satisfiability of propositional Horn formulæ. *Journal of Logic Programming*, 1(3):267–284, October 1984.

[DGLM95] Mark Day, Robert Gruber, Barbara Liskov, and Andrew C. Myers. Subtypes vs. Where Clauses: Constraining Parametric Polymorphism. In *Proc. OOPSLA '95*, pages 156–168, Austin TX, October 1995. ACM SIGPLAN Notices 30(10).

[DOD85] Department of Defense. *Department of Defense Trusted Computer System Evaluation Criteria*, DOD 5200.28-STD (The Orange Book) edition, December 1985.

[DV66] J. B. Dennis and E. C. VanHorn. Programming semantics for multiprogrammed computations. *Comm. of the ACM*, 9(3):143–155, March 1966.

[Fei80] Richard J. Feiertag. A technique for proving specifications are multilevel secure. Technical Report CSL-109, SRI International Computer Science Lab, Menlo Park, California, January 1980.

[Fen73] J. S. Fenton. *Information Protection Systems*. PhD thesis, University of Cambridge, Cambridge, England, 1973.

[Fen74] J. S. Fenton. Memoryless subsystems. *Computing J.*, 17(2):143–147, May 1974.

[FK92] David Ferraiolo and Richard Kuhn. Role-based access controls. In *15th National Computer Security Conference*, 1992.

[FLR77] R. J. Feiertag, K. N. Levitt, and L. Robinson. Proving multilevel security of a system design. *Proc. 6th ACM Symp. on Operating System Principles (SOSP), ACM Operating Systems Review*, 11(5):57–66, November 1977.

[FM96] J. Steven Fritzinger and Marianne Mueller. Java security. Technical report, Sun Microsystems, Inc., 1996.

[Fol91] Simon N. Foley. A taxonomy for information flow policies and models. In *Proc. IEEE Symposium on Security and Privacy*, pages 98–108, 1991.

[FSBJ97] Elena Ferrari, Pierangela Samarati, Elisa Bertino, and Sushil Jajodia. Providing flexibility in information flow control for object-oriented systems. In *Proc. IEEE Symposium on Security and Privacy*, pages 130–140, Oakland, CA, USA, May 1997.

[GGF98] Virgil D. Gligor, Serban I. Gavrila, and David Ferraiolo. On the formal definition of separation-of-duty policies and their composition. In *Proc. IEEE Symposium on Security and Privacy*, pages 172–183, Oakland, California, USA, May 1998.

[GJS96] James Gosling, Bill Joy, and Guy Steele. *The Java Language Specification*. Addison-Wesley, August 1996. ISBN 0-201-63451-1.

[GM84] J. A. Goguen and J. Meseguer. Unwinding and inference control. In *Proc. IEEE Symposium on Security and Privacy*, pages 11–20, April 1984.

[GS98] Li Gong and Roland Schemers. Implementing protection domains in the Java Development Kit 1.2. In *The Internet Society Symposium on Network and Distributed System Security*, San Diego, California, USA, March 1998. Internet Society.

[HDT87] Susan Horwitz, Alan Demers, and Tim Teitelbaum. An efficient general iterative algorithm for dataflow analysis. *Acta Informatica*, 24:679–694, 1987.

[HR98] Nevin Heintze and Jon G. Riecke. The SLam calculus: Programming with secrecy and integrity. In *Proc. 25th ACM Symp. on Principles of Programming Languages (POPL)*, San Diego, California, January 1998.

[JD96] Daniel Jackson and Craig A. Damon. Elements of style: Analyzing a software design feature with a counterexample detector. *IEEE Transactions on Software Engineering*, 22(7):484–495, July 1996.

[JG91] Pierre Jouvelot and David K. Gifford. Algebraic reconstruction of types and effects. In *ACM Symposium on Principles of Programming Languages*, pages 303–310, January 1991.

[JL75] A. K. Jones and R. J. Lipton. The enforcement of security policies for computation. In *Proc. 5th ACM Symp. on Operating System Principles (SOSP), ACM Operating Systems Review*, pages 197–206, November 1975.

[JL78] Anita K. Jones and Barbara Liskov. A language extension for expressing constraints on data access. *Comm. of the ACM*, 21(5):358–367, May 1978.

[Kil73] G. Kildall. A unified approach to global program optimization. In *Proc. ACM Symp. on Principles of Programming Languages (POPL)*, 1973.

[KU76] John B. Kam and Jeffrey D. Ullman. Global data flow analysis and iterative algorithms. *Journal of the ACM*, 23(1):158–171, 1976.

[KW94] Atsushi Kanamori and Daniel Weise. Worklist management strategies for dataflow analysis. Technical Report MSR–TR–94–12, Microsoft Research, May 1994.

[LAB+84] Barbara Liskov, Russell Atkinson, Toby Bloom, J. Eliot Moss, J. Craig Schaffert, Robert Scheifler, and Alan Snyder. *CLU Reference Manual.* Springer-Verlag, 1984. Also published as Lecture Notes in Computer Science 114, G. Goos and J. Hartmanis, Eds., Springer-Verlag, 1981.

[LABW91] Butler Lampson, Martín Abadi, Michael Burrows, and Edward Wobber. Authentication in distributed systems: Theory and practice. In *Proc. 13th ACM Symp. on Operating System Principles (SOSP)*, pages 165–182, October 1991. *Operating System Review*, 253(5).

[Lam71] Butler W. Lampson. Protection. In *Proc. Fifth Princeton Symposium on Information Sciences and Systems*, pages 437–443, Princeton University, March 1971. Reprinted in Operating Systems Review, 8(1), January 1974, pp. 18–24.

[Lam73] Butler W. Lampson. A note on the confinement problem. *Comm. of the ACM*, 16(10):613–615, October 1973.

[LCD+94] Barbara Liskov, Dorothy Curtis, Mark Day, Sanjay Ghemawat, Robert Gruber, Paul Johnson, and Andrew C. Myers. *Theta Reference Manual.* Programming Methodology Group Memo 88, MIT Laboratory for Computer Science, Cambridge, MA, February 1994. Available at http://www.pmg.lcs.mit.edu/papers/thetaref/.

[LMM98] Barbara Liskov, Nicholas Mathewson, and Andrew C. Myers. PolyJ: Parameterized types for Java. Software release. Located at http://www.pmg.lcs.mit.edu/polyj, July 1998.

[LY96] T. Lindholm and F. Yellin. *The Java Virtual Machine.* Addison-Wesley, Englewood Cliffs, NJ, May 1996.

[MBL97] Andrew C. Myers, Joseph A. Bank, and Barbara Liskov. Parameterized types for Java. In *Proc. 24th ACM Symp. on Principles of Programming Languages (POPL)*, pages 132–145, Paris, France, January 1997.

[McL88] John McLean. Reasoning about security models. In *Proc. IEEE Symposium on Security and Privacy*, pages 123–131, Oakland, CA, 1988. IEEE.

[McL90] John McLean. Security models and information flow. In *Proc. IEEE Symposium on Security and Privacy*, pages 180–187, 1990.

[MF96] Gary McGraw and Edward Felten. *Java Security: Hostile Applets, Holes and Antidotes.* John Wiley and Sons, New York, 1996.

[Mic97] Microsoft Corporation. *Trust-Based Security for Java*, April 1997.

[Mil76] Jonathan K. Millen. Security kernel validation in practice. *Comm. of the ACM*, 19(5):243–250, May 1976.

[Mil81] Jonathan K. Millen. Information flow analysis of formal specifications. In *Proc. IEEE Symposium on Security and Privacy*, pages 3–8, April 1981.

[Mil87] Jonathan K. Millen. Covert channel capacity. In *Proc. IEEE Symposium on Security and Privacy*, Oakland, CA, 1987.

[ML97] Andrew C. Myers and Barbara Liskov. A decentralized model for information flow control. In *Proc. 17th ACM Symp. on Operating System Principles (SOSP)*, pages 129–142, Saint-Malo, France, 1997.

[ML98] Andrew C. Myers and Barbara Liskov. Complete, safe information flow with decentralized labels. In *Proc. IEEE Symposium on Security and Privacy*, Oakland, CA, USA, May 1998.

[MMN90] Catherine J. McCollum, Judith R. Messing, and LouAnna Notargiacomo. Beyond the pale of MAC and DAC—defining new forms of access control. In *Proc. IEEE Symposium on Security and Privacy*, pages 190–200, 1990.

[MR92] M. D. McIlroy and J. A. Reeds. Multilevel security in the UNIX tradition. *Software—Practice and Experience*, 22(8):673–694, August 1992.

[MTH90] R. Milner, M. Tofte, and R. Harper. *The Definition of Standard ML.* MIT Press, Cambridge, MA, 1990.

Ex. 1002, Page 162

[MWCG98] Greg Morrisett, David Walker, Karl Crary, and Neal Glew. From System F to typed assembly language. In *Proc. 25th ACM Symp. on Principles of Programming Languages (POPL)*, San Diego, California, January 1998.

[Mye99] Andrew C. Myers. JFlow: Practical mostly-static information flow control. In *Proc. 26th ACM Symp. on Principles of Programming Languages (POPL)*, San Antonio, TX, USA, January 1999.

[Nec97] George C. Necula. Proof-carrying code. In *Proc. 24th ACM Symp. on Principles of Programming Languages (POPL)*, pages 106–119, January 1997.

[Nel91] Greg Nelson, editor. *Systems Programming with Modula-3*. Prentice-Hall, 1991.

[Net97] Netscape Communications Corporation. *Introduction to the Capabilities Classes*, 1997.

[Nv98] Tobias Nipkow and David von Oheimb. $Java_{light}$ is type-safe—definitely. In *Proceedings of the 25th ACM Symposium on Principles of Programming Languages*, pages 161–170. ACM, New York, January 1998.

[OW97] Martin Odersky and Philip Wadler. Pizza into Java: Translating theory into practice. In *Proc. 24th ACM Symp. on Principles of Programming Languages (POPL)*, pages 146–159, Paris, France, January 1997.

[PO95] Jens Palsberg and Peter Ørbæk. Trust in the λ-calculus. In *Proc. 2nd International Symposium on Static Analysis*, number 983 in Lecture Notes in Computer Science, pages 314–329. Springer, September 1995.

[Rei79] Richard P. Reitman. A mechanism for information control in parallel programs. In *Proc. 7th ACM Symp. on Operating System Principles (SOSP), ACM Operating Systems Review*, pages 55–62, December 1979.

[RM96] Jakob Rehof and Torben Æ. Mogensen. Tractable constraints in finite semilattices. In *Proc. 3rd International Symposium on Static Analysis*, number 1145 in Lecture Notes in Computer Science, pages 285–300. Springer-Verlag, September 1996.

[RSC92] Joel Richardson, Peter Schwarz, and Luis-Felipe Cabrera. CACL: Efficient fine-grained protection for objects. In *Proceedings of the 1992 ACM Conference on Object-Oriented Programming Systems, Languages, and Applications*, pages 154–165, Vancouver, BC, Canada, October 1992.

[Sal74] J. H. Saltzer. Protection and the control of information sharing in Multics. *Comm. of the ACM*, 17(7):388–402, July 1974.

[San92] Ravi S. Sandhu. A lattice interpretation of the Chinese Wall policy. In *Proc. of the 15th NIST-NCSC National Computer Security Conference*, pages 221–235, Baltimore, Maryland, USA, October 1992.

[San96] Ravi S. Sandhu. Role hierarchies and constraints for lattice-based access controls. In *Proc. Fourth European Symposium on Research in Computer Security*, Rome, Italy, September 25-27 1996.

[SCFY96] Ravi S. Sandhu, E.J. Coyne, H.L. Feinstein, and C.E. Youman. Role-based access control models. *IEEE Computer*, 29:38–47, February 1996.

[SNS88] J. G. Steiner, C. Neuman, and J. I. Schiller. Kerberos: An authentication service for open network systems. Technical report, Project Athena, MIT, Cambridge, MA, March 1988.

[SS98] Pierangela Samarati and Latanya Sweeney. Generalizing data to provide anonymity when disclosing information. In *ACM Principles of Database Systems*, Seattle, Washington, USA, June 1998.

[Sto81] Allen Stoughton. Access flow: A protection model which integrates access control and information flow. In *IEEE Symposium on Security and Privacy*, pages 9–18. IEEE Computer Society Press, 1981.

[Sto87] B. Stoustrup. *The C++ Programming Language*. Addison-Wesley, 1987.

[SV98] Geoffrey Smith and Dennis Volpano. Secure information flow in a multi-threaded imperative language. In *Proc. 25th ACM Symp. on Principles of Programming Languages (POPL)*, San Diego, California, January 1998.

[Swe96] Latanya Sweeney. Replacing personally-identifying information in medical records, the Scrub System. *Proceedings, Journal of the American Medical Informatics Association*, pages 333–337, 1996.

[Sym97] Don Syme. Proving Java type soundness. Technical Report 427, University of Cambridge Computer Laboratory, June 1997.

[TW89] Phil Terry and Simon Wiseman. A 'new' security model. In *Proc. IEEE Symposium on Security and Privacy*, pages 215–228, 1989.

[Vol97] Dennis Volpano. Provably-secure programming languages for remote evaluation. *ACM SIGPLAN Notices*, 32(1):117–119, January 1997.

[VSI96] Dennis Volpano, Geoffrey Smith, and Cynthia Irvine. A sound type system for secure flow analysis. *Journal of Computer Security*, 4(3):167–187, 1996.

[WBF97] Dan S. Wallach, Dirk Balfanz, and Edward W. Felten. Extensible security architectures for Java. In *Proc. 16th ACM Symp. on Operating System Principles (SOSP)*, pages 116–128, Saint-Malo, France, October 1997.

[WCC+74] W. A. Wulf, E. Cohen, W. Corwin, A. Jones, R. Levin, C. Pierson, and F. Pollack. HYDRA: The kernel of a multiprocessor system. *Comm. of the ACM*, 17(6):337–345, June 1974.

[WF98] Dan S. Wallach and Edward W. Felten. Understanding Java stack inspection. In *Proc. IEEE Symposium on Security and Privacy*, Oakland, California, USA, May 1998.

[Ylo96] Tatu Ylonen. Ssh – secure login connections over the Internet. In *The Sixth USENIX Security Symposium Proceedings*, pages 37–42, San Jose, California, 1996.

Index

access control, 14, 28, 58
  discretionary, 68, 149, 151
  integrating with, 155, 156
  mandatory, 151
acts-for relation, 30, 33
  extending, 55
actsFor statement, 68, 117, 146
annotations, 38
arrays, 66, 78, 112
  access rule, 110
  assignment rule, 112
assignment rules, 112
authority, 25, 68, 73, 98, 124, 125, 146

bank example, 70
basic blocks, 64
boolean satisfaction problem, 136
break statement, 114

capabilities, 68, 80, 149
channels, 26, 29, 50
Chinese Wall policy, 153
classes, 71, 80, 98, 103, 120, 123
code labels, 54
compatibility with Java, 85
completeness, 39
  proof, 42
constraints, 73, 120, 125, 133
  dynamic, 138
constructors, 120, 129
continue statement, 114
correctness condition
  dynamic, 36
  static, 39
covariant parameters, 77, 100, 124
covert channels, 60, 62, 65, 84, 157
cryptography, 157

dataflow analysis, 144
declassification, 13, 21, 25, 54, 69
declassify statement and expression, 118, 146
dependent types, 77
devices, 55
discretionary access control, 14, 28, 149, 151
distribution properties, 45, 49
dynamic cast, 116
dynamic checking, 62
dynamic constraints, 138
dynamic labels, 66, 148
dynamic principals, 70, 146

empty statement, 108
encryption, 157
environments, 92, 97, 103, 123
example
  bad relabeling, 39
  bank, 70
  complex number, 81
  hospital, 28
  password file, 82
  protected, 84
  tax preparer, 12, 27
  vector, 71, 76
exceptions, 93
  unchecked, 75, 85, 94

fields, *see* instance variables
final variables, 77, 105
finalizers, 84
flows, 32

greatest lower bound, 25

hashCode, 84
Horn clauses, 136
hospital example, 28

if statement, 113
implicit flows, 62, 64
incremental relabeling, 30, 41, 52
inference rules, 92
input channels, 26

Ex. 1002, Page 165

instance variables, 64, 112, 130
  access rule, 110
  assignment rule, 112
instanceof, 116
integrity, 51–153
  combining with privacy, 54
  declassification for, 54
  policies, 51
interpretation functions, 35
interpreting labels, 32

JFlow, 60
join (⊔), 25, 46
judgements, 93

labeled statement, 114
labeled types, 63
labels, 22, 98, 102
  code, 54
  components, 98, 103
  creep, 24
  defaults, 72
  dynamic checking, 62
  generalizing, 51
  inference, 61
  interpretation function, 33, 35
  label expressions, 99, 146
  paths, 94
  polymorphism, 61, 75
  program-counter, 64, 69
  restriction, 30
  run-time, 66, 119, 148
  semantics, 32
  static checking, 62, 92
  variables, 23, 103
lattices, 49, 137
least upper bound, 25
literal, 108

mandatory access control, 151
meet (⊓), 25, 47
meet operator, 136, 137
member functions, *see* methods
member variables, *see* instance variables
methods, 71, 123, 127
  arguments, 127
  bodies, 128
  calls, 120
  constraints, 73, 128
  signatures, 106, 124, 125
monotonicity, 36, 39
multilevel security, 62, 153
mutual distrust, 13, 154

notation, 95, 96, 98

originator-controlled release, 154
output channels, 26, 29, 50
owners, 22

parameterized types, 75
parameters, 100, 103
  covariant, 77
password file example, 82
path labels, 94
policies
  integrity, 51
  privacy, 22
  redundant, 46
PolyJ, 77
polymorphism, 75
  labels, 61
principals, 22, 98, 102, 146
  hierarchy, 22, 68, 74, 98, 117
  owners, 22
  readers, 22
  run-time, 70, 146
  writers, 51
principle of least privilege, 32, 56
privacy, 22–26
  combining with integrity, 54
process authority, 25
program annotations, 38
program-counter label, 64, 69
proof
  of relabeling completeness, 42
  of relabeling soundness, 42
protected example, 84

readers, 22
  constraint, 34
redundant policies, 46
relabeling
  complete rule, 40
  examples, 24, 39
  incremental, 30, 41, 52
  proof of completeness, 42
  proof of soundness, 42

Ex. 1002, Page 166

relation $\sqsubseteq$, 41
subset rule, 23
relations
$\succeq$, 22, 55
$\sqsubseteq$, 23, 41, 53
restriction, 30
revocation, 68
role-based access control, 156
rules
actsFor statement, 117
arithmetic, 109
break, 114
constructors, 129
continue, 114
declassify statement and expression, 118
dynamic cast, 116
empty statement, 108
exceptions, 115
if, 113
instanceof, 116
literal, 108
local variables, 110
method arguments, 127
method bodies, 128
method declarations, 127
sequence, 113
signature compatibility, 125
single-path, 108
subtyping, 106
switch label, 119
variable access, 110
variable assignment, 112
while, 113
run-time labels, 66, 119, 148
run-time principals, 70, 146

separation of duties, 153
signatures of methods, 106, 124, 125
single-path rule, 108
soundness, 24
proof, 42
stack inspection, 149
static checking, 62
static correctness condition, 39
static security enforcement, 155
strongly connected components, 143
subset relabeling rule, 23, 30, 39
subtyping, 64, 106
switch label statement, 119, 146, 148

tax preparer example, 12, 27
termination channels, 65
theorem proving, 156
threads, 84
throw statement, 115
timing channels, 65, 84, 85
topological sort, 143
transactions, 69
translating JFlow code, 146
trusted subject, 15, 154
try/catch/finally statement, 115
types
dependent, 77
discrimination, 78
interpretation, 100
labeled, 63
parameters, 78
static checking, 92
subtype rules, 106
subtyping, 64

uids, 102

variables, 23, 112
access rules, 110
assignment rule, 112
extend function, 110
final, 105
static, 84
Vector, 76
visibility, 82, 92

waivers, 154
while statement, 113

Ex. 1002, Page 167

OIPE JC54
JAN 27 2003
PATENT & TRADEMARK OFFICE

2143 #3 BJ
01-31-03

PATENTS

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re Application of Ron Redlich

RECEIVED
JAN 28 2003
Technology Center 2100

Serial No.: 10/155,192

Filed: May 23, 2002

For: DATA SECURITY SYSTEM AND METHOD WITH PARSING AND DISPERSION TECHNIQUES

SUPPLEMENTAL INFORMATION DISCLOSURE STATEMENT TRANSMITTAL LETTER

Commissioner for Patents
Washington, D.C. 20231

Sir:

Please find enclosed for filing:

X Supplemental Information Disclosure Statement (2 pages)

X Form PTO-1449 (1 page)

X Copies of listed information items accompanying this Statement (5 reference)

This Transmittal Letter is submitted in duplicate.

Respectfully submitted,

Fleit, Kain, Gibbons, Gutman & Bongini, P.L.

By _____
Robert C. Kain, Jr.
Reg. No. 30,648
750 S.E. Third Avenue, Suite 100
Ft. Lauderdale, Florida 33316-1153
(954) 768-9002

CERTIFICATE UNDER 37 CFR 1.8(a)
I HEREBY CERTIFY THAT THIS CORRESPONDENCE IS BEING DEPOSITED WITH THE U.S. POSTAL SERVICE AS FIRST CLASS MAIL IN AN ENVELOPE ADDRESSED TO ASSISTANT COMMISSIONER FOR PATENTS, WASHINGTON, D.C. 20231 BOX _____ ON 1-22-03
Reg. No. 30648
Date 1-22-03

Docket No.: 6851-2-cip3

\\Tiger\rck\CLIENTS\Redlich\6851-2-cip3 TRANS.Suppl.IDS.wpd

OIPE JC54
JAN 27 2003
PATENT & TRADEMARK OFFICE

PATENTS

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re Application of Ron Redlich

RECEIVED
JAN 28 2003
Technology Center 2100

Serial No.: 10/155,192

Filed: May 23, 2002

For: DATA SECURITY SYSTEM AND METHOD WITH PARSING AND DISPERSION TECHNIQUES

SUPPLEMENTAL INFORMATION DISCLOSURE STATEMENT

Commissioner for Patents
Washington, D.C. 20231

Sir:

As a means of complying with the duty of disclosure set forth in 37 C.F.R. §1.56, applicant, through the undersigned attorney, files this Information Disclosure Statement pursuant to 37 C.F.R. §1.97. Each of the following references were cited within the past three (3) months during the prosecution of foreign patent applications, that is, during the international phase of a PCT applciation.

Patent No.	Inventor	Issue Date
5,933,498	Schneck et al.	Aug. 3, 1999
6,078,907	Lamm	Jun. 20, 2000
5,581,682	Anderson et al.	Dec. 3, 1996
US2002/0073311	Brown et al.	Jun. 13, 2002
WO 00/75779-A2	Lambert	Dec. 14, 2000

This Information Disclosure Statement is filed in accordance with 37 C.F.R. §1.97, and this statement shall not be construed as a representation that a search was made, has been made, or that the

information cited above is, or is considered to be, material to patentability as defined in §1.56(b), or that no other material information, as defined in 37 C.F.R. §1.56(b), exists. A copy of each listed patent or publication or other item of information in written form, or of at least the portions thereof listed above, is filed with this Information Disclosure Statement. If two or more patents or publications are considered substantially cumulative, a copy of a representative one is included with this statement and others are merely listed above. Translations of the pertinent portions of foreign language patents or publications listed above are supplied if existing translations are readily available to the applicant or the applicant's attorney. A concise explanation of the relevance, as it is presently understood by the individual designated in 37 CFR §1.56(c) most knowledgeable about the content of the information, of each patent, publication, or other information listed that is not in the English language is provided either in the patent specification or in the enclosed attachment.

Respectfully submitted,

FLEIT, KAIN, GIBBONS, GUTMAN & BONGINI, P.L.

By _____
Robert C. Kain, Jr.
Reg. No. 30,648
Suite 100
750 S.E. Third Avenue
Ft. Lauderdale, Florida 33316-1153
(954) 768-9002

CERTIFICATE UNDER 37 CFR 1.8(a)
I HEREBY CERTIFY THAT THIS CORRESPONDENCE IS BEING DEPOSITED WITH THE U.S. POSTAL SERVICE AS FIRST CLASS MAIL IN AN ENVELOPE ADDRESSED TO ASSISTANT COMMISSIONER FOR PATENTS WASHINGTON, D.C. 20231 BOX _____ ON 1-22-03

Reg. No. 30648
Date 1-22-03

\\Tiger\rck\CLIENTS\Redlich\6851-2-cip3-supple-IDS.wpd

OIPE JC
JAN 27 2003
PATENT & TRADEMARK OFFICE

INFORMATION DISCLOSURE CITATION *(Use several sheets if necessary)*	Docket Number (Optional) 6851-2 CIP3	Application Number 10/155,192
	Applicant(s) Ron Redlich	
	Filing Date May 23, 2002	Group Art Unit

U.S. PATENT DOCUMENTS

Examiner Initial	Document Number	Date	Name	Class	Subclass	Filing Date if Appropriate
	5,933,498	8/3/99	Schneck et al.	380	4	
	6,078,907	6/20/00	Lamm	705	40	
	5,581,682	12/3/96	Anderson et al.	395	792	
	US2002/0073311	6/13/02	Brown et al.	713	165	

RECEIVED
JAN 28 2003
Technology Center 2100

FOREIGN PATENT DOCUMENTS

	Document Number	Date	Country	Class	Subclass	Translation YES NO
	WO 00/75779-A2	12/14/00	Lambert			

OTHER DOCUMENTS (Including Author, Title, Date, Pertinent Pages, Etc.)

EXAMINER DATE CONSIDERED

\\Tiger\rck\CLIENTS\Redlich\6851-2-cip3 Form 1449 for Suppl.IDS.wpd

(12) INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
14 December 2000 (14.12.2000)

PCT

(10) International Publication Number
WO 00/75779 A2

(51) International Patent Classification[7]: G06F 9/46

(21) International Application Number: PCT/US00/15224

(22) International Filing Date: 2 June 2000 (02.06.2000)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:
60/137,568 4 June 1999 (04.06.1999) US

(71) Applicant *(for all designated States except US)*: IWITNESS, INC. [US/US]; Suite 2N, 2995 Wilderness Place, Boulder, CO 80301 (US).

(72) Inventors; and

(75) Inventors/Applicants *(for US only)*: LAMBERT, Francis, T. [US/US]; 1901 Spruce Street, Boulder, CO 80302 (US). ARCHER, Emory, Scott [US/US]; 4799 White Rock Circle, #E, Boulder, CO 80301 (US).

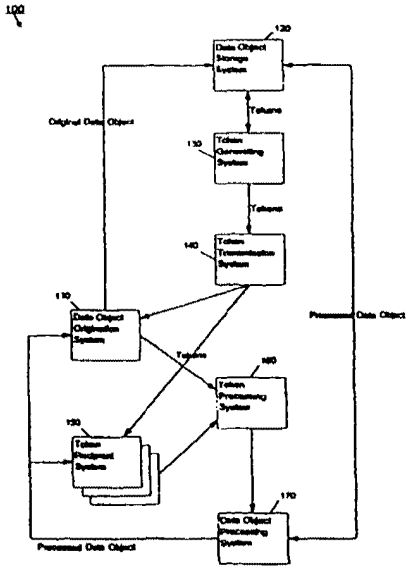
(74) Agents: SABETT, Randy, V. et al.; Cooley Godward LLP, One Freedom Square-Reston Town Center, 11951 Freedom Drive, Reston, VA 20190-5601 (US).

(81) Designated States *(national)*: AE, AL, AM, AT, AU, AZ, BA, BB, BG, BR, BY, CA, CH, CN, CR, CU, CZ, DE, DK, DM, EE, ES, FI, GB, GD, GE, GH, GM, HR, HU, ID, IL, IN, IS, JP, KE, KG, KP, KR, KZ, LC, LK, LR, LS, LT, LU, LV, MA, MD, MG, MK, MN, MW, MX, NO, NZ, PL, PT, RO, RU, SD, SE, SG, SI, SK, SL, TJ, TM, TR, TT, TZ, UA, UG, US, UZ, VN, YU, ZA, ZW.

(84) Designated States *(regional)*: ARIPO patent (GH, GM, KE, LS, MW, MZ, SD, SL, SZ, TZ, UG, ZW), Eurasian patent (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European patent (AT, BE, CH, CY, DE, DK, ES, FI, FR, GB, GR, IE,

[Continued on next page]

(54) Title: TOKEN BASED DATA PROCESSING SYSTEMS AND METHODS



(57) Abstract: A token based data processing system is disclosed that uses data object tokens (DOTs) to manage the associated data objects within the data processing system. The numerous advantages of a token based data processing system stem from the ability to utilize a DOT separate from its associated data object. A data processing system according to the present invention can include a data object origination system, a data object storage system, a token generating system, a token transmission system, one or more token recipient systems, a token processing system, and a data object processing system. The token generating system can generate a DOT that contains metadata corresponding to the information that the token generating system receives regarding the data object, including its attributes, the parameters stored with it, and other information regarding its originating system and its recipient system. After being distributed by a token transmission system, token recipient systems can analyze and process the DOTs in lieu of the associated data objects.

WO 00/75779 A2

IT, LU, MC, NL, PT, SE), OAPI patent (BF, BJ, CF, CG, CI, CM, GA, GN, GW, ML, MR, NE, SN, TD, TG).

Published:

— *Without international search report and to be republished upon receipt of that report.*

For two-letter codes and other abbreviations, refer to the "Guidance Notes on Codes and Abbreviations" appearing at the beginning of each regular issue of the PCT Gazette.

TOKEN BASED DATA PROCESSING SYSTEMS AND METHODS

RELATED APPLICATION

The present application claims priority to Provisional Application No. 60/137,568, filed June 4, 1999, which is incorporated herein by reference in its entirety.

BACKGROUND

Field of the Invention

This invention relates to a data processing system containing diverse data objects and to methods for using those data objects. More specifically, the invention relates to a data processing system where the system utilizes data objects called "tokens" to perform processes related to, and upon, other data objects to which the tokens refer.

Description of the Related Art

The advent of electronic data interchange, the growth of electronic communications, the growth of the number of electronic communication users, the growth of the availability of information, and the growth in the size, number and types of data objects has greatly increased the requirements of both data storage and data communication bandwidth. Typical use of these data objects requires processing and transfer of the data object itself, often due to the complexity or sensitivity of the data content of the object, or to the limitations of the data systems used to manipulate the data object. These systems generally require transfer or transmission of the entire data object for a user to be able to edit, preserve, alter, or "own" the data on the person's

local computing system. Examples of these systems can include any that use e-mail attachments, ftp file transfer, and http file transfer.

Unfortunately, a modern data object storage event or a transmission event, such as an electronic document transfer via e-mail, may contain a large volume of data that one entity wishes to transfer to another entity. This type of transfer forces both the sending and receiving parties to expend substantial resources in terms of CPU usage, data storage, and transmission bandwidth at both ends of the transmission. The cost and lost productivity from these types of transmissions will increase as the number and size of these transmissions increase, as is the case in modern data communications.

Additionally, the originator of the transmitted data may wish to keep the transfer of the data secure and private. This requires sending the data using encryption, which requires additional CPU usage. The originator may also wish to guarantee that the transmitted data arrived exactly as sent, which requires that all of the data be sent, verified, and the verification returned to the originating system each time the data is transmitted. These security and authentication technologies must also be present on the receiving system. Prior to the data being released to the transmission network, the originator must have knowledge that the receiving system is capable of reliably authenticating, decrypting, and controlling access to the recipient's copy of the data object.

In many cases, the receiver of the data in this type of transmission is forced to receive all of the transmitted data whenever the originator sends it, regardless of whether or not it is convenient for the receiver at that time. An example of this is a mobile computer user being forced to download a large e-mail attachment over an expensive hotel data line so that they can retrieve all of their e-mail. Another example

is a wireless data transceiver being forced to download unnecessary data when only limited bandwidth is available.

In the case where a data object is transmitted to multiple recipients, the originator is required to transmit the data object multiple times, thereby expending valuable CPU and bandwidth resources in a the concentrated time period of transmission Also, each recipient is required to receive and store its copy of the transmitted object, thereby using bandwidth and data storage resources at each recipient. If the recipients are receiving diverse variations on a data object, the originator is required to create, store, and transmit the separate variations of the data object.

It is also possible that the volume of data in the transmitted data object is too large for the receiving system to efficiently store, or that numerous accumulated objects are too large for the system to efficiently store, and that it would be desirable to have another system with greater storage capacity hold the object and provide authorized access to the object only as needed.

It is also possible that the originator of the data would want to control the access and use of the object by the receiver, or create an audit trail of the receiver's use of the data object. The originator may wish that various receiving parties have access only to portions of the data object specific to the nature of the receiver. The originator or the receiver may desire to provide authentication relating to a data object without transmitting or storing the object itself.

Mechanisms have existed in the past for providing information about stored data objects. For example, a computer file (one type of data object) under the Disk Operating System (DOS) or the Windows operating system has certain flags associated

with it to indicate information about that file. The flags indicate, for example, whether the file is "read only", whether it has been archived, and whether it can be written. The UNIX operating system utilizes a similar approach of storing flags with files. However, the flags in all of these systems are rudimentary, and they are not easily extensible. The flags also tend to be stored as part of the object to which they refer and are therefore cumbersome to process without access to the file itself.

Other types of data objects contain elements that refer to other data objects within the context of the containing object. An example of this could be a filename combined with disk sector locations in a storage directory object, such as a file allocation table. Another example could be a hyperlink with addressing information contained within an HTML object. These object elements act as "pointers" to an object, but do not function outside of the context of the containing object. They also require a process to access the referenced object to perform processes or analysis on the referenced object. They do not act as a "surrogate" for the object, allowing processes to perform functions on an object without accessing the object itself.

Unfortunately, current data network transmission and storage systems fail to provide a robust, extensible, and universally compatible means for those users or systems transmitting and receiving data to efficiently use data objects in such as way as to conserve CPU usage, bandwidth usage, and storage through the use of smaller, more efficient objects containing metadata about the data object that can be processed separately from the data object itself. This smaller, more efficient object can contain metadata for use in access control, use auditing, redactability (or the filtering of content in an object), recipient and sender authentication, workflow logic, data integrity, and

timing of data delivery, among others. Consequently, a need exists for a data processing and transfer system that overcomes the foregoing drawbacks.

SUMMARY OF THE INVENTION

The present invention provides a data processing system that can allow a user to manipulate a data object through the use of one or more separate data objects comprised of metadata about the original data object. The metadata can represent many attributes of a data object, including, but not limited to, information about the content of the data object and the context of its occurrence. It can also represent information about searchable elements in the data object or information on the use history of the object. Some specific examples of metadata can include object identification, access parameters, authentication requirements and values (for both originator and user), encryption methods and keys, access history, integrity checking values, redaction values (or restricting access to certain elements or properties of the original data object), and token validity expiration period. In an embodiment, the DOT can also contain references (such as hyperlinks) to other locations where other metadata values can be found. Systems holding metadata values can also control access to and log use of those metadata values in other locations according to the contents of the token. A data processing system according to the present invention can combine diverse metadata into a discrete and processable data object called a Data Object Token (DOT) that refers to one or more data objects. The DOT includes metadata that provides information about a data object.

In one embodiment, a data object originator can submit a data object to a token based data processing system, which can optionally store the data object and generate

one or more tokens containing metadata of varying complexity referring to the stored data object. The one or more tokens can then be returned to the originator, or sent directly to token recipients, or stored separately in the data object storage system for later use. This Data Object Token, the DOT, can be used by the originator or by token recipients to interact with the data object that is optionally stored on a system storage device such as a central data depository attached to a network, or in another location specified in the DOT according to the protocol used to locate a data object, including a URL, a storage device identifier, or other data object locating protocols. The DOT user can submit the DOT to the token based data processing system to retrieve, view, alter, delete, or otherwise interact with the data object. The token based data processing system can analyze the metadata values contained in the DOT to determine use of the data object, including allowing access, processing authentication values, performing redaction, transmiting the object or portions thereof, destroying the data object, or otherwise manipulating the data object. It can also store and transmit metadata about data object as a smaller, discrete object that can be managed using native and pre-existent file and transmission processes.

One advantage of a token based data processing system is that it allows users of data objects to have access to and control over data objects without being required to expend local storage or bandwidth resources in the process. A DOT can be of very small size and still refer to a much larger data object in a remote location, while providing many capabilities to the local processes that would otherwise require the presence of the data object on the local system. A DOT can be very efficiently transmitted to multiple users, requiring much less storage and bandwidth than multiple transmissions of the referenced data object.

Another advantage is that the user of the DOT can retrieve, download, view, or otherwise process the referenced data object at their convenience by submitting the DOT to the token based data processing system. Another advantage is that the user can receive a DOT that allows them to decide upon the usefulness of, and securely retrieve, the referenced data object without requiring them to download the data object at time of originator transmission to them.

Furthermore, a token based data processing system can contain the authentication values and methods for both the originator and the DOT user that validate access and other rights to the data object. The token based data processing system can validate the DOT as it contains the originator authentication, as well as validate other parameters of the DOT based on the user authentication, before access, processing, or retrieval of the original data object is allowed.

The DOT can also contain data redaction information tailored to the nature of the DOT user. The DOT can provide the system information as to which portions of the data object the user will be allowed access. This relieves the data object storage system from the requirement of having to store and process redaction information for every potential user of the data object.

A further advantage is that the DOT can contain data object routing information for workflow and rules processing of data objects. The process can be more efficient and less bandwidth intensive if a token representing the workflow data object is passed around instead of the object itself.

Another advantage is that the DOT can append data object usage information to itself whenever it is used, and thereby become a self-contained data object usage audit

trail. This usage information could contain user identity and authentication, and time and type of use of use of both the token and the referenced object.

Yet another advantage is that the data object originator can stipulate in the DOT an expiration or starting time for processing of the referenced data object. This allows an originator to control the timing of various aspects of processing the stored data object, as well as avoid the need to provide a local copy of the data object to the DOT user.

Another advantage to using a DOT as a surrogate for a data object is that it avoids exposure of the data object to potential security and privacy problems inherent in the transmission of data over public networks such as the Internet. For example, even if a DOT is sent to the wrong recipient, or "sniffed" from the public network, the referenced object cannot be processed or retrieved since the unauthorized token possessor will not have the correct authentication to access the object.

Another advantage to using the DOT as a surrogate to the referenced data object is that the DOT can contain searchable elements of the original data object, such as keywords or text, and allow searching of the original data object, or a set of data objects, without requiring access to them. This allows searching to be distributed to the token user machine, thus conserving bandwidth and CPU usage at the object storage system.

Another advantage of using a DOT to retrieve an object is that an integrity value, such as a hash digest, of the retrieved data object can be compared to an integrity value in the token user's DOT to determine if the data object was received correctly.

Another advantage of using a DOT approach is that a token generating system can be embedded in an e-mail server, allowing the e-mail system to embed a DOT in

the place of large e-mail attachments. This conserves bandwidth and storage for both the sender and receiver.

Another advantage of DOTs is that they can be used as an electronic payment system due to their high level of security and ability to provide an audit trail. For example, the DOT can represent a portion of a pre-paid monetary amount represented by the referenced data object, such as a bank account. The payment DOT can be submitted to a token processing system that reduces the bank account balance by the amount embedded in the DOT, and then expires the DOT by placing its unique identifier on an expired DOT list. This would allow cashless payment in electronic business transactions, without exposing any sensitive financial information to a public network.

Another advantage of using DOTs is that they decouple the object storage technology from the object access technology, allowing object storage technologies to change without changing the way the stored objects are accessed by the users, and allowing the access objects, or DOTs, to remain valid and functional over long periods of time, even if the storage systems that they access change.

Another advantage of using DOTs is they it allow users to maintain representations of libraries of objects on a local system for off-line browsing, management, and processing.

Additional aspects and embodiments of the present invention will become apparent upon a review of the following detailed description and accompanying drawings.

BRIEF DESCRIPTION OF THE DRAWINGS

Fig. 1 is a functional block diagram of an embodiment of the present invention.

Fig. 2 depicts the process flow of a system according to the present invention.

Fig. 3 depicts the basic structure of a data object token.

Fig. 4 is a notational representation of the structure of one embodiment of a data object token.

Fig. 5 is a notational representation of additional elements in the structure of one embodiment of a data object token.

Fig. 6 is a command table for use in an embodiment of the present invention.

Fig. 7 is a flow chart of an embodiment of a token generating system.

Fig. 8 is a flow chart of an embodiment of a token processing system.

Fig. 9 is a flow chart of an embodiment of a token recipient system.

DETAILED DESCRIPTION

Fig. 1 depicts a block diagram of an embodiment of a token based data processing system 100 according to the present invention. Token based data processing system 100 can contain data object origination system 110, data object storage system 120, token generating system 130, token transmission system 140, token recipient system 150, token processing system 160, and data object processing system 170.

Data object origination system 110 can be any device or system that can be utilized to process information and produce data objects, including but not limited to such things as a personal computer, a mainframe computer, or an embedded computing system connected to a network. Data object origination system 110 can further include numerous processes that utilize data objects. For example, data object origination

system 110 could be utilized to create a data object and store that data object in data object storage system 120. Similarly, data object origination system 110 could also be utilized to download and store information, in the form of one or more data objects, from a data network. In particular, data object origination system 110 can utilize and store various data objects including, but not limited to, database files, electronic documents, e-commerce transactions (such as stock trades or merchandise purchases), electronic records (such as medical records or personal histories), multi-part records from document imaging systems, Web sites, sequences of HTTP data, or digitized video and audio data.

Following the creation or reception of one or more data objects in data object origination system 110, the data objects can be stored in data object storage system 120, which can be a local data storage system or a remote data storage system. For example, in the case of a personal computer, a local data object storage system 120 can include a hard drive, or a combination of a floppy disk drive and a floppy disk. Alternatively, a remote version of data object storage system 120 can include storage on a Local Area Network (LAN) or on a Wide Area Network (WAN) using such mechanisms as file servers or digital tape libraries.

In addition to receiving and storing the data objects, data object storage system 120 can expose the data objects to token generating system 130. Token generating system 130 can analyze and process the information about the data object, such as its attributes, the parameters stored with it, and other information regarding its originating system and its recipient system. Token generating system 130 can then generate metadata corresponding to that information, and create a data object called the Data Object Token (DOT) that contains the metadata associated with the data object. Token

generating system 130 can further process the DOT to embed data values that provide or ensure such things as security, privacy, reliability, or efficiency. Numerous means exist for providing these attributes, including encryption, digital signatures, hash digests, other authenticating technologies, compression, or other processing technologies. As one particular example, token generating system 130 can provide authentication, integrity, and non-repudiation information about the data objects using a public key digital signature. Upon successfully verifying the digital signature, the recipient of the digital signature can have the additional assurance of the authenticity of the originator, of the integrity of the data object, and that the data object creation cannot be repudiated.

Once processed, token generating system 130 can further present the DOT to data object storage system 120 and to token transmission system 140 for transmission to one or more token recipient systems 150. Data object storage system 120 or token transmission system 140 can store the DOT or a portion of the DOT for later use, thereby avoiding later reprocessing of the data object.

Token generating system 130 can read the object to be tokenized from data object storage system 120. Token generating system 130 can then determine from a token reference file the types and fields of metadata desired to be embedded in the token. A token reference file can be viewed as a template that contains rules governing the creation or transformation of tokens, including but not limited to encoding of the DOT (e.g. using XML, ASN.1, or comma-separated); required data and metadata fields; organizing structure of metadata fields; and mappings of data types and structures from the source data object to a token. Token generating system 130 can then read the metadata from the object and write it into a file that will become the DOT.

This metadata can include a unique identifier for the object, object location information, object hash digest values, or other metadata fields as required. Token generating system 130 can then determine from the token reference file if it is required to generate metadata from processing the referenced data object's content or metadata fields. Token generating system 130 can then read the content or metadata from the object, process it as required (such as, for example extracting keywords from content text, or reducing images for more efficient searching), and then embed the processed metadata into the token extension fields (see Fig. 3) in the DOT file. The metadata can be written in popular data formats, such as XML or comma delimited fields.

Token generating system 130 can then determine from the token reference file whether token generating system 130 is required to retrieve metadata external to the object, such as the date, a stock price, a digital certificate, encryption keys, other authentication data, or other data as is available to token generating system 130 and required by the token reference file. Token generating system 130 can then embed the externally retrieved data into the appropriate fields in the DOT file. It can also use predetermined time stamp and authentication fields, as well as token extension fields as required. Token generating system 130 can also embed additional metadata into the token file extension fields, such as token validity period, redaction values, or workflow logic to be used when referencing the original data object. Token generating system 130 then can embed its version number in the token file to allow later processing by token processing system 160 and, particularly, to differentiate among different versions of tokens.

Token generating system 130 can then digitally sign the DOT file by methods well known in the art of digital signing, including without limitation, encrypting, with a

private key from a public key infrastructure (PKI) key pair, a combination of a hash digest of the token with a digital certificate, and embedding the digital signature data into the token file. Token generating system 130 can then encrypt the entire token file using well known encryption algorithms, such as triple DES or the RC5 encryption methods.

The DOT file generated by token generating system 130 can then be stored in the data object storage system 120, in the data object generation system 110, or in token generating system 130 itself to await further processing.

Once token generating system 130 creates a DOT, token transmission system 140 can read the DOT from token generating system 130 and determine from a token transmission reference file whether or not the DOT is required to be transmitted to a token recipient system 150, or stored for later use. A token transmission reference file, similar to a token reference file, can contain template rules, such as recipient types, required transmission metadata, and required authentication, that would govern the transmission of DOTs. If the DOT is to be transmitted, the token transmission system 140 can embed information on accessors to the data object in the DOT as required. This information can be used by token processing system 160 to determine the authorization and access rights of the submitted DOT. The token transmission system 140 can embed multiple object accessors values to allow use of the DOT by multiple users or systems. The DOT can then be encrypted in the public key of the transmission destination system, and can then be securely and reliably transmitted to one or more token recipient systems 150 using either a standard communications protocol, such as FTP, e-mail attachment, or HTTP, or by using a proprietary communications protocol.

A token recipient system 150 system can include a system capable of receiving, storing, or processing a DOT. The token transmission system 140 can determine addressing and routing information for the destination token recipient systems 150 from a token transmission reference file, or from the token reference file contained in token generating system 130. In one embodiment, data object origination system 110 can also be a token recipient system. In an embodiment, token transmission system 140 can further verify and log reception of the DOT by one or more token recipient systems 150.

Token recipient system 150 can receive, store, and optionally process or otherwise condition a DOT. Processing on a DOT could include searching on metadata or content from the object embedded in the DOT. This searchable metadata or content could include, for example, text, keywords, or images from the original data object. Other types of processing that could be performed on a DOT by the token recipient system include embedding recipient reception information, such as time stamp and digital signature, appending additional accessors to the referenced object if the recipient system is so authorized, or forwarding the DOT to another token recipient system. The optional processing performed by token recipient system 150 can include other types of analysis and processing of the information contained in the DOT, including its metadata. Since token recipient system 150 can perform its analysis and processing on the metadata contained within the DOT (which is acting as a surrogate for the data object itself), no resources within data object storage system 120 would need to be used.

For example, in one embodiment, token recipient system 150 can contain an e-mail computer application that receives DOTs associated with e-mail messages. In this

case, each message could be a data object. The metadata contained within a particular DOT that is associated with a specific e-mail message can contain information about the e-mail message, including but not limited to such things as the date, time, subject, sender, and, in particular, any attachments associated with the e-mail message. The DOTs can be used by a token recipient system to search for a particular e-mail message or for a group of messages. However, this search could be done solely within the token recipient system using metadata embedded in the DOTs associated with the data objects (i.e. the e-mail messages), not on the messages themselves, which can be stored on a remote data object storage system 120. Also, using this approach, any attachments to the e-mail message would only need to be stored in the data object storage system, not sent to a token recipient system nor stored at the data object origination system.

Furthermore, in one particular embodiment related to e-mail, a token generating system can be embedded in an e-mail server to allow automated storage and tokenization of e-mail, including, without limitation, any of its components such as header, body, or attachments. Token generating system 130 embedded in the e-mail server can read the e-mail objects controlled by the server, generate a DOT for the e-mail object, replace the original e-mail object with the DOT, and store the e-mail object in a more efficient data object storage system location, such as a digital tape library. The e-mail user can then use the DOT from within the e-mail client application to access or process the e-mail data object. Searching for e-mail content within a group of e-mails can be accomplished by searching the search metadata embedded in the DOTs in the e-mail client application, thereby avoiding the CPU and bandwidth costs associated with searching the group of e-mails on the e-mail server or at the data object storage system. This process of substituting DOTs for data objects such as e-mails can

be applied to many types of data objects, including without limitation, scanned or imaged documents, electronic documents, database tables, electronic spreadsheets, web sites, and other types of data objects specified by the user of a token based data object management system.

Upon the completion of a search and a determination that one or more specific e-mail messages, or other type of data objects, need retrieval or processing, token recipient system 150 can submit one or more DOTs to token processing system 160 which can then cause data object storage system 120 to be accessed and the referenced data objects to be processed according to the instructions in the DOT, or transmitted to a data object processing system 170 as requested in the submitted DOT.

After token recipient system 150 submits a DOT to token processing system 160 to retrieve a data object from data object storage system 120, token processing system 160 can interpret the metadata contained in the DOT. Using the metadata, token processing system 160 can submit processing commands to data object processing system 170 according to the metadata contained in the DOT. These commands could include, for example, transmit, redact, alter, or destroy a data object as shown in Fig. 6.

Alternatively, a token recipient system can submit a DOT to a token processing system to retrieve a set of DOTs associated with the submitted DOT. This referenced set of DOTs can reference data objects or other DOTs, or a combination thereof, as required by the system.

Data object processing system 170 can interact with the data object stored in data object storage system 120 according to the commands submitted by token processing system 160. For example and without limitation, in response to a particular

command, data object processing system 170 can transmit, redact, alter, or destroy the data object stored on the data object storage system, according to the authorizations embedded in the submitted DOT, and as interpreted by the token processing system 160, which can pass the commands to the data object storage system 120.

In another embodiment where data object storage system 120 contains data object processing system 170, data object storage system 120 can process the referenced data object according to the instructions embedded in the DOT. For example, if data object storage system 120 is instructed to transmit the object, it can place the object into a data transmission process as is commonly known to the art, or it can encrypt the object using an encryption value submitted with the token, such as the destination system public key, or it can transmit the encrypted object to a system different from the system that submitted the token, or it can transmit the object to multiple systems. Similarly, data object storage system 120 can redact the stored object by recomposing the object prior to transmission, according to redaction rules embedded in the token. An example without limitation of this can be the removal of personal information from a data object for transmission of the data object from within the European Community to a destination outside of the European Community, thereby complying with the European Privacy Act, or other regulations that require redaction of information in transmission or storage.

Data object storage system 120 can additionally alter the data object by changing its retention information, its authorized accessor list, its storage format or location, or even its content if required by the system. Data object storage system 120 can even destroy the data object according to the processing instructions embedded in the submitted token.

Fig. 2 depicts a process flow diagram of an embodiment of the present invention. In data object origination system 210, a data object can be created. Once created, that data object can be transmitted to data object storage system 220. Once received, data object storage system 220 can process the data object so as to enhance the security and reliability of the data object storage process by, for example, performing encryption, compression, hashing, or mirroring, or by digitally signing the data object. For example, the well known DES encryption algorithm can be used to encrypt the data object. In addition, the well known RSA public key digital signature algorithm can be used to digitally sign the data object in data object storage system 220. Following the processing performed on the data object by data object storage system 220, the data object can be stored. The data object storage system can then place the processed object where it will be read by token generating system 230.

Token generating system 230 can generate tokens in standard data formats. For example, token generating system 230 can generate XML-based tokens based on the content of the data objects. To do so, token generating system 230 first reads the stored data object in data object storage system 220. It then analyzes the data object and the associated token reference files. The token reference files can contain instructions and information that can allow the token generating system to determine how the DOT will be constructed from the data object. Once analyzed, token generating system 230 can generate the necessary XML tags to be placed in the DOT for the particular system. In other embodiments, a token generating system can also, for example, generate fixed length fields, comma delimited fields, and other data formats that are commonly used by computer applications. In general, a token generating system can produce any type of DOT that may be needed for a particular data processing system.

In addition, token generating system 230 can encrypt, compress, hash and otherwise process one or more DOTs so as to enhance the security and reliability of the DOTs. Similar to the processing that can be done on the data object itself, the well known DES encryption algorithm can be used to encrypt the DOT. In addition, the well known RSA public key digital signature algorithm can be used to digitally sign the DOT in token generating system 230.

Furthermore, in an embodiment of the present invention, token generating system 230 can embed in the DOT time relevant attributes of the DOT, the data object, or the token recipient system's 250 ability to manipulate it. These time relevant attributes could include, for example, a time stamp of the DOT generation time, expiration of the DOT, the data object, or authentication metadata as determined by the contents of the token reference file. These time relevant attributes could also include "no access until" information that could not allow the token submitting entity to manipulate the data object until a certain time or event.

In an embodiment, once token generating system 230 has generated a DOT, token transmission system 240 can sense the DOT and can cause token generating system 230 to pass the DOT to token transmission system 240. To do so, token transmission system 240 can poll the system holding the DOT and retrieve the DOT in a data "pull" operation once it is available. Token transmission system 240 can then determine the destination for the DOT, followed by determining the methods to be used for the delivery of the DOT, according to mechanisms such as the transmission reference file in the token transmission system, or from the token reference file in the token generating system, or from a workflow logic file. Token transmission system 240 can then transmit DOTs using any well known data transmission technologies,

including, without limitation, the FTP protocol, the HTTP protocol, the SSL protocol, a virtual private network (VPN), or Secure Shell tunneling in the Unix operating system. In another embodiment, a token transmission system can send the DOT as an attachment to an e-mail message. Alternatively, a token transmission system can send a notification to a token recipient system of a token pending transmission to which token recipient system 250 must respond to release or access the token, and a token transmission system 240 can then issue a receipt describing the token delivery event.

After receipt of a DOT from a token transmission system, another token recipient system, or any other means of transmitting a token, such as in an e-mail as an attachment, token recipient system 250 can authenticate the DOT using any authentication techniques that may have been applied by token generating system 230. Once authenticated, token recipient system 250 can store the DOT and subsequently submit the DOT to token processing system 260 for data object manipulation.

In addition to submitting the DOT for data object manipulation, a token recipient system can also add new metadata to the received DOT, or alter the DOT according to the token recipient system's authority to do so as determined by the authorization metadata embedded in the DOT, and optionally transmit that updated DOT to another token recipient system or to the token processing system 260. This can allow metadata within a DOT to accumulate without having to access the associated data object from storage. A data processing system utilizing this type of DOT approach could undergo a change of the native platform and data formats of data object storage system 220 without impacting the capabilities of pre-existing DOTs used to manage and process the data objects within the system, provided the new native platform and data formats remained compatible with the required versions of the DOT structure.

In one embodiment, workflow logic can be embedded within the token or can be captured in an external file. Workflow logic refers to workflow processing rules stored and interpreted both inside and outside of a DOT, and that can affect both the internal operations of the token based data processing system and the overall workflow distribution patterns of tokens and data objects within an organization.

For example, workflow logic could be used by an organization to manage the workflow associated with a particular project. Workflow routing instructions could be embedded in the DOT if it is advantageous for the logic to be contained in the DOT, or the DOT could reference an external workflow logic file that could be dynamically updated. The DOT can reference an object or a set of objects intended to be the subject of a workflow process. This can allow the workflow system to distribute metadata about workflow-related data objects without expending the CPU and bandwidth to distribute the actual data object(s) themselves. The token recipient system 250 itself can execute the workflow logic embedded in the DOT, such as forwarding a DOT automatically to another token recipient system if the referenced data object is not accessed within a certain period of time, or other similar workflow logic.

Upon receiving the DOT from token recipient system 250, token processing system 260 can perform on the DOT any type of processing that might be necessary. For example, the DOT can be decrypted, uncompressed, or otherwise processed to restore it to its original form in response to security and reliability processing performed by the token generating system. The type of processing required can be determined by token processing system 260 by reading the metadata embedded in the DOT, such as the version number, or other metadata indicating how the DOT has been previously processed.

As another example, token processing system 260 can authenticate the entity submitting the DOT using digital certificates, recipient supplied passwords, biometric measurements, or any combination thereof, according to values contained in the token. Alternatively, the entity originating the DOT can be authenticated using digital certificates, recipient supplied passwords, biometric measurements, or any combination thereof, according to values contained in the token.

In yet another embodiment, token processing system 260 can create and append metadata to the DOT that can describe the use of the DOT and the data object. This metadata can be used create a self-contained audit trail within the token. For example, this self-contained audit trail can be comprised of use metadata and authentication values embedded into the token during a workflow execution by the associated token recipient systems.

Token processing system 260 can, in another embodiment, process metadata that describes the token submitting entity's access rights to the referenced data object. Those access rights can be used by data object processing system 270 to control access to all or portions of the data object. Metadata can also be used to describe the time attributes of a DOT, the referenced data object or the authenticating metadata. Token processing system 260 can interpret this time-based metadata to determine proper usage of the token and the associated data object.

The results of the token processing done by token processing system 260 can be passed as commands to data object processing system 270. Upon receiving one or more of these commands, data object processing system 270 can authenticate the command using any well known authentication technique already described herein. Upon successful authentication, data object processing system 270 can determine what

action needs to be taken to respond to the command. For example, destroying or deleting the referenced object from the object store, or appending the audit trail information embedded in the submitted DOT are actions that can be taken in response to a command.

One frequent action by data object processing system 270 can be the accessing of one or more data objects in data object storage system 220. If data object processing system 270 determines that the proper conditions have been met, including, for example, appropriate authentication information, one or more data objects can then be accessed in data object storage system 220. Once accessed, the requested actions can be performed on the accessed data objects by data object processing system 270 and those actions can also be logged by data object processing system 270. Finally, any results of the processing performed by data object processing system 270 can be passed back to token processing system 260. In another embodiment, data object processing system 270 can transmit one or more data objects to the token submitting entity. Prior to doing so, data object processing system 270 can optionally encrypt, compress, hash, sign, or otherwise process the data object to enhance security and reliability in transmission and delivery.

Figure 3 depicts a DOT 300 in accordance with the present invention. The basic DOT structure consists of a version field 304, token information 308, object accessors 328 , token and referenced object use information 352, and optional token extensions 368.

Version field 304 indicates the version of the data object token protocol to which DOT 300 conforms. The inclusion of version field 304 permits a token processing system to properly parse the DOT.

Token information 308 contains the fields necessary for a token processing system to assess the validity of the DOT and the data object to which the DOT is associated. Data object identifier 312 can contain a unique value or character string that identifies the data object to which it refers. Validity information 316 allows a system to make the determination of whether the DOT is valid. For example, validity information 316 can contain date and time information, such that a validity period can be established. Alternatively, validity information 316 can contain a hash digest of the object and the DOT that can be used for purposes of determining data integrity of a received object or DOT.

Data object and DOT authentication information 320 can contain any type of data that can be used to authenticate the data object or the DOT itself. In an embodiment, data object authentication information can contain a public key from a PKI key pair. In another embodiment, data object authentication information 320 can contain a digital signature. The authentication information can be used to authenticate the system creating the referenced data object, the system generating the DOT, or the system submitting the DOT to a token processing system or token recipient system. Timestamp 324 can provide information to an entity about the time of creation of the token. A data object originator could use the timestamp, for example, to prove that a data object representing a transaction was created at a specific time on a specific date.

The next main component of the DOT is object accessor 328. This can provide information about the entity or entities permitted to access and use the data object referenced by the DOT. In an embodiment, there may be multiple instantiations of object accessor 328 for each permitted accessor of the data object. Each object accessor 328 can contain information about the accessor 332, access rights 336,

authentication requirements 340, accessor authentication information 344, and timestamp 348. In an embodiment, the object accessor metadata can contain a list of workflow destinations upon which workflow distribution processing can be based.

Accessor 332 can contain information that identifies the entity being given rights to access the data object. This entity identifier could identify a system, a process, a class of processes, a specific user, a group of users, a reference to a dynamically alterable accessor list, or the general public. For example, accessor 332 can contain a name, an e-mail address, and a unique identifier. In another embodiment, accessor 332 can contain a digital certificate.

Access rights 336 can provide information to the token processing system regarding the rights that the accessor identified by accessor 332 has to the data object to which the DOT refers. Access rights 336 can include, for example, general rights, such as those used to access, download, modify, or destroy a data object. In addition, access rights 336 can include other, more specific rights that are unique to the particular data processing system. For example, a DOT used in a stock trading system might contain rights particular to a stock trader, such as buy unrestricted, buy limited, sell unrestricted, or sell limited. In another example, access rights can also stipulate authorization of the token recipient system to alter the metadata in the token, or the authorization to append audit trail metadata, as required by the system. Authentication requirements 340 provide information about the type of authentication required to be used by the token user to prove that it is the entity identified by accessor 332. Numerous, well understood, technologies exist to provide authentication in several different formats. For example, a particular data processing system may require something to be known by the token user, such as a password or personal identification

number (PIN). Alternatively, a data processing system may require the user to have an object, such as a smart card or a hardware "dongle". Finally, a data processing system might require physical information from the token user (i.e. a biometric), such as a fingerprint, a retinal scan, or a voice print.

Accessor authentication information 344 provides information about the entity that created the object accessor entry. This can be used to audit the granting of rights to access the referenced data object.

Timestamp 348 can provide information to an entity about the time of creation of the object accessor 328 metadata. A token processing system could use the timestamp, for example, to prove that the authentication information 344 was historically valid at the time of its creation. For example, if a digital certificate is used as authentication information 344, time stamp 348 can be used at a later time to determine if the digital certificate had expired or been revoked at the time of object accessor 328 metadata creation.

The next component of DOT 300 in an embodiment of the present invention, token use information 352, contains information about the use of the DOT and use of the data object to which the DOT refers. Token use information 352 can include information on the type of access requested 356 by the token user. Once authenticated using access authentication information 360, a token processing system can utilize token use information 352 to determine how it should process the object referenced by the DOT. This could include uploading the object to the token submitter, changing data in the object, or destroying the object.

One or more token extensions 368 allow the DOT to provide additional information beyond the standard DOT including, but not limited to, such things as

redaction instructions and audit trail data related to the associated data object. The format of each token extension 368 can consist of extension type 372, extension data 376, and extension authentication information 380. Within token extension 368, extensible metadata can be stored that can then be interpreted by the token recipient system or the token processing system. Some examples of the use of token extension 368 include DOT expiration values for a particular system, searchable content metadata values, workflow logic, additional (system specific) access rights, externally retrieved metadata, and any other fields that may be added by a token recipient system.

Fig. 4 and Fig. 5 provide a description of one embodiment of a DOT that can be associated with a Verified Data Object (VDO), as implemented by iWitness, Inc. of Boulder, Colorado in its electronic records management products. The descriptions in Fig. 4 and Fig. 5 use the well known Abstract Syntax Notation 1 (ASN.1) that is defined in "CCITT Recommendation X.208: Specification of Abstract Syntax Notation One (ASN.1)", published in 1988. The ASN.1 descriptions in Fig. 4 and Fig. 5 depict a Verified Data Object Token (VDOT).

Fig. 4 depicts the basic VDOT structure 410, which includes the two main components of tokenInfo and accessors. The tokenInfo component can provide information specific to the VDO, while the accessors component can provide information about the entities that may access the VDO. As shown in 420, the tokenInfo component can include globally unique object identifier vdo_id, the interval of time during which the token is valid, a digital signature and certificate identifying the creation event, and a timestamp. The vdo_id element can be used to uniquely identify the particular VDO to which the VDOT refers and can contain a randomly generated ASCII character string that had been verified as being unique. In a different

embodiment, the vdo_id could be a consecutive number applied to a particular VDO amongst a large set of other VDOs. The validFrom and validTo components together provide the validity period for the VDOT user to access the particular VDO. In an embodiment, if the UniversalTime value within the data processing system utilizing the VDO is less than the validFrom time or greater than the validTo time, the entity submitting the token in an attempt to access the associated VDO would be denied that access.

The signerInfo component provides the digital signature and information about the signer who applied the digital signature used to verify the VDO. In an embodiment, signerInfo can be the signer information defined in "PKCS #7: Cryptographic Message Syntax Standard: An RSA Laboratories Technical Note", Version 1.5, Revised November 1, 1993, which specifies, amongst other things, the signer's digital certificate, the algorithm used to sign the VDO, and the digital signature itself.

The accessors component within VDOT basic structure 410 specifies a collection of entities along with their access rights and authentication requirements for the specified VDO. As shown in accessor information 430, those entities can be identified in a table consisting of a set of DataObjectTokenAccessor elements, wherein each DataObjectTokenAccessor element can contain a PKCS #6 certificate, the access rights needed to access the VDO, the authentication requirements for the particular VDO, the PKCS #7 signer information, the PKCS #6 certificate of the signer of the VDOT, and a timestamp.

Access rights listing 440 for a particular VDOT can provide information on how a particular entity may use the VDOT in an embodiment of the invention. Only entities identified by the distinguished name and serial number from their enclosed PKCS #6

certificate can make use of the VDOT, including, for example, the ability to retrieve the corresponding VDO. Those entities are limited to their corresponding access rights, which are signed by the entity responsible for their assignment.

Authentication requirements 450 for a particular VDOT provides information on how a particular entity must authenticate itself prior to its use of the VDOT. Any use of the token, whether for VDO retrieval or communication to another entity, may require the using entity to satisfy one or more authentication requirements, such as knowledge of a password, a physical match of biometric data, or an external key. Additionally, each entry in the accessor table can be signed by its creator.

Fig. 5 depicts an ASN.1 description of the token use information 510 element and the token extension 520 element in accordance with an embodiment of the present invention. Token use information 510 can contain request information that indicates how a VDOT user would like to use the corresponding VDO and authentication information that allows the token processing system to determine the validity of the request. Token use information 510 can be included upon initial generation of the VDOT or can be appended at some later time after VDOT creation.

In an embodiment of the invention, one or more instantiations of the optional token extension 520 element can be included in a VDOT. In a VDOT, extType can specify the extension type of this particular extension.

Fig. 6 shows a command table implemented in one embodiment of the present invention that contains commands for performing various operations on a data object. A data object processing system can use the information shown in 610 to execute the commands on the specified data object. Transmit object command 620 can result in a data object being transmitted to the destination specified by the destination argument,

and can include information on authorization, transmission time, recipient requirements, and transport mechanism. Destroy object command 630 can result in the destruction of a data object, and can include information on authorization, destruction time, and disposition rules. Copy object command 640 can result in the copying of a data object from one location to the location specified by the destination argument. Alter object command 650 can result in one or more modifications to the data object, and can include information on the module to perform the operation, a reference to the data object, the actual alteration to be performed on the data object, and any necessary parameters. Redact object command 660 can result in certain information within a data object being removed while the rest of the data remains, and can include the module to perform the redaction, a reference to the data object, the redaction to be performed on the data object, and any necessary parameters. Finally, object encryption command 670 can result in the encryption or digital signing of a data object, and can include the operation to be performed on the data object, the method or algorithm to be used, the key to be used to perform the cryptographic operation, and whether to just perform a hash on the data object for data integrity purposes.

Figure 7 depicts a flow chart of the processing that can occur in a token generating system utilized for workflow processing. In step 704, the token generating system can receive a DOT generation request, which contains request attributes specifying the data object to be analyzed, the token reference file that defines the structure and required contents of the generated DOT, and possibly other control data. In order to ensure the security of the token generating system, and thus the security of the DOT and its future use in other systems, the DOT generation request is validated in step 712 through one or more of the various authentication means available including,

but not limited to, public key digital signatures. Once the DOT generation request is validated, the data object to which it refers is analyzed in step 716 to determine the viability of the request. In particular, as shown in the substeps in 720, the DOT generation request is checked against workflow logic (if any), the data object is then checked against workflow logic carried in the DOT generation request (if any), and finally, the data object is assessed to determine if it contains the necessary data/metadata to fulfill the request according to the token reference file. If the request cannot be fulfilled, a result indicating this can be returned in step 736.

If the authentication and subsequent analysis succeed, the token generation system can proceed to generate the requested DOT in step 724. As shown in substeps 728, all requested metadata that is derived directly from the data object can first be retrieved and attached. Then, all requested metadata that is derived external to the data object (if any) can be attached, possibly including workflow logic defined in the token request file. Finally, any requested DOT access rights can be constructed and attached. The newly formed DOT can then be digitally signed in step 732 by the token generation system, and possibly encrypted and enveloped following the Cryptographic Message Syntax as defined in PKCS #7. As shown in step 736, the DOT can then be returned synchronously to the originator of the DOT request and/or asynchronously transmitted through a token transmission system (as shown in the substeps in 740). Control could then be returned to the token transmission system in step 744.

Figure 8 depicts a flow chart of the processing that can occur in a token processing system utilized for workflow processing. In step 804, the token recipient system can send a processing request. In step 808, the token processing system can receive a processing request, which can contain, as shown in 812, a request DOT and

request attributes specifying one or more operations to be performed on the data object that the DOT references. Before processing, the token processing system can validate the public-key digital signatures on the DOT in step 816, to ensure the authenticity of the DOT and the authenticity of the requestor. Once validated, the token recipient system can open the cryptographic envelope of the DOT in step 816 (if it is encrypted) with the private-key that corresponds to the public-key used in securing the DOT for transmission. This ensures that the token can only be used by the intended token processing system.

Once validated and decrypted, the token processing system can check the token-embedded or external file workflow logic in step 820 and in the substeps shown in 824. The token processing system can then interpret the processing request according to the workflow logic of the token in step 828 in order to cause the data object processing system in 836 to perform the requested operations in step 832, such as fetch data/metadata, update metadata, destroy object, and transmit object/token, all of which can be performed on the data object within the data object processing system.

Results of the data object operation can be validated by the token processing system in step 840 to ensure their authenticity and conformance to the requirements of the request. The results can then be returned in step 844, and in particular, as shown in substeps 848, the results can be returned synchronously to the originator of the processing request and/or asynchronously transmitted as an updated token through a token transmission system. Once the results have been returned, control can return to the token transmission system in step 852.

Figure 9 depicts a flow chart of the processing that can occur in a token recipient system utilized for workflow processing. The token transmission system can

send a DOT in step 904 and token recipient system can receive a DOT from the token transmission system in step 908. Before processing, the token recipient system can validate the public-key digital signatures on the DOT in step 912, to ensure the authenticity of the DOT and the authenticity of the sender. Once validated, the token recipient system can open the cryptographic envelope of the token in step 912 (if it is encrypted) with the private-key that corresponds to the public-key used in securing the token for transmission. This ensures that the token can only be used by the intended token recipient system.

Once validated and decrypted, the token recipient system checks both the token-embedded and external file workflow logic for conflicts in steps 916 and the substeps in 920. If non-critical conflicts are found, resolution may require priority-based assessment or possibly user intervention. The token recipient system can then process the token according to the workflow logic in step 924. As shown in the substeps in 928, processing can include presentation to the user, storage for archival or for later use. At some point, the user of the token recipient system or the system itself can use the received (and possibly stored) token to request one or more operations on the data object to which it refers, as shown in step 932. This processing can be done through the token processing system as shown in step 936.

While the invention has been described in detail, including references to specific embodiments, it will be apparent to one skilled in the art that changes and modifications can be made to the invention without departing from the spirit and scope thereof. Thus, it is intended that the present invention cover the modifications and variations of this invention provided they come within the scope of the appended claims and their equivalents.

What is claimed is:

1. A method of managing one or more data objects within a data processing system using one or more data object tokens, the method comprising the steps of:

generating one or more data objects within a data object origination system;

storing said data objects in a data object storage system;

analyzing said data objects to determine information to be placed in one or more data object tokens;

generating said one or more data object tokens;

transmitting said one or more data object tokens to one or more token recipient systems;

processing said one or more data object tokens; and

processing said one or more data objects based upon said data object tokens.

2. A method as in claim 1, wherein said storing step further comprises storing said data objects in a local storage system.

3. A method as in claim 1, wherein said storing step further comprises storing said data objects in a remote storage system over a network.

4. A method for generating one or more data object tokens associated with a data object, the method comprising:

receiving a token generation request;

validating said token generation request;

analyzing said data object to determine information to be placed in said one or more data object tokens;

populating said data object token with metadata;

creating validation information related to said data object token; and

returning a completed data object token.

5. A method as in claim 4, wherein said validating step further comprises the step of verifying a digital signature using a public key.

6. A method as in claim 4, wherein said analyzing step further comprises the step of determining information to be redacted in said data object.

7. A method as in claim 4, wherein said analyzing step further comprises the step of determining information to be altered in said data object.

8. A method as in claim 4, wherein said populating step further comprises the step of appending one or more optional token extensions.

9. A method for processing one or more data object tokens associated with a data object, the method comprising:

receiving a token processing request;

validating said token processing request;

interpreting said token processing request to determine the operations to be performed on one or more data objects;

commanding said operations to be performed on said one or more data objects by a data object processing system;

validating the results of said operations; and

returning the resulting data object.

10. A method for limiting access by a user to data objects in a data processing system, the method comprising:

storing said data object securely within said data processing system;

creating a data object token corresponding to said data object, said data object containing authentication requirements;

distributing said data object token;

receiving an access request from a data object token user, said request containing data object token user authentication information and access authentication information;

authenticating said access request;

processing the data object to secure it for use by the data object token user; and

transmitting said processed data object to said data object token user.

11. A method as in claim 10, wherein said storing step further comprises performing local encryption on said data object.

12. A method as in claim 10, wherein said storing step further comprises locating said data object behind a firewall.

13. A method as in claim 10, wherein said first validation step further comprises verification of a first public key digital signature, and wherein said second validation step further comprises verification of a second public key digital signature.

14. A method for validating the use of a data object using a data object token in a token based data processing system, the method comprising:

performing a first validation on data object originator authentication information contained within said data object token; and

performing a second validation on data object token user authentication information contained within said data object token.

15. A method as in claim 14, wherein said first validation step further comprises verification of a first public key digital signature, and wherein said second validation step further comprises verification of a second public key digital signature.

16. A method of retrieving a data object in a token-based data processing system, the method comprising the steps of:

receiving a data object token from a token generating system;

storing said data object token;

appending data object token user authentication information;

submitting said data object token to a token processing system; and

receiving said data object.

17. A method as in claim 16, wherein said data object token user information comprises a public key digital signature.

18. A method for distributing analysis and processing activities related to one or more data objects stored in a data object storage system within a token based data processing system, comprising the steps of:

receiving one or more data object tokens in a token recipient system;

storing said data object tokens; and

processing said data object tokens within said token recipient system.

19. A method as in claim 18, wherein said processing step further comprises the steps of:

extracting metadata associated with said one or more data objects from said one or more data object tokens;

performing an analysis on said metadata; and

accessing one or more data objects within said data object storage system based upon said analysis.

20. A method as in claim 18, wherein:

said processing step further comprises a search through data object tokens containing electronic mail header information based on a set of search criteria; and

said accessing step further comprises accessing e-mail messages meeting said search criteria.

21. A token based data processing system that manages one or more data objects within a data processing system using one or more data object tokens, comprising:

a data object origination system for generating one or more data objects;

a data object storage system for storing said one or more data objects;

a token generating system for generating one or more data object tokens;

a token transmission system for transmitting said one or more data object tokens to one or more token recipient systems;

a token processing system for processing said one or more data object tokens; and

a data object processing system for processing said one or more data objects based upon the processing of said one or more data object tokens.

22. A token based data processing system as in claim 21, wherein said data object storage system is contained within said data object origination system.

23. A token based data processing system as in claim 21, wherein said data object storage system is located over a network remotely from said data object origination system.

24. A token based data processing system as in claim 21, wherein said data objects further comprise scanned or imaged documents.

25. A token generating system for generating one or more data object tokens that refer to one or more data objects, comprising:

a means for receiving a token generation request;

a means for validating said token generation request;

a means for analyzing said data object to determine information to be placed in said one or more data object tokens;

a means for populating said data object token with metadata;

a means for creating validation information related to said data object token; and

a means for returning a completed data object token.

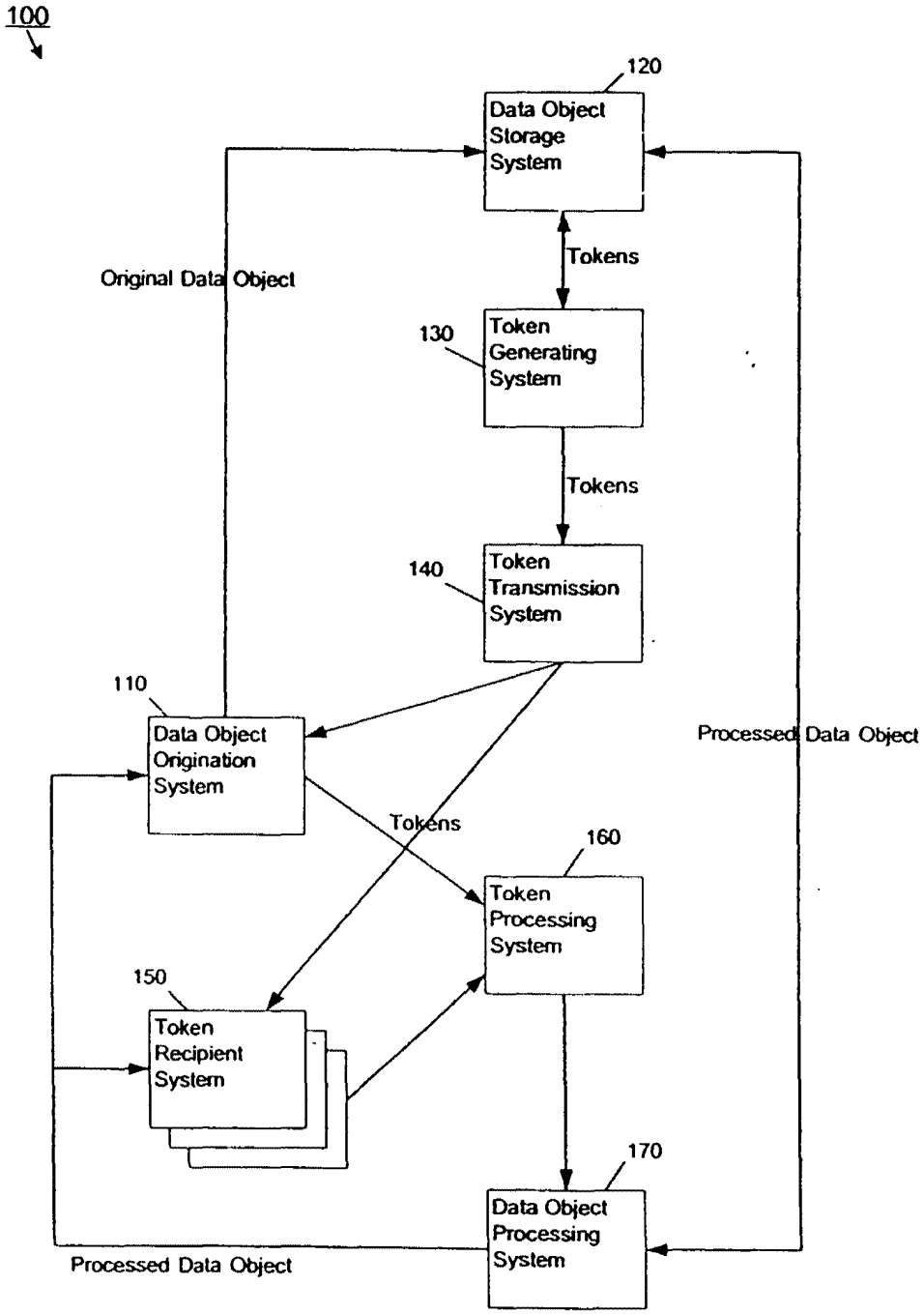

100
120
Data Object Storage System
Tokens
Original Data Object
130
Token Generating System
Tokens
140
Token Transmission System
110
Data Object Origination System
Processed Data Object
Tokens
160
Token Processing System
150
Token Recipient System
170
Data Object Processing System
Processed Data Object

Figure 1

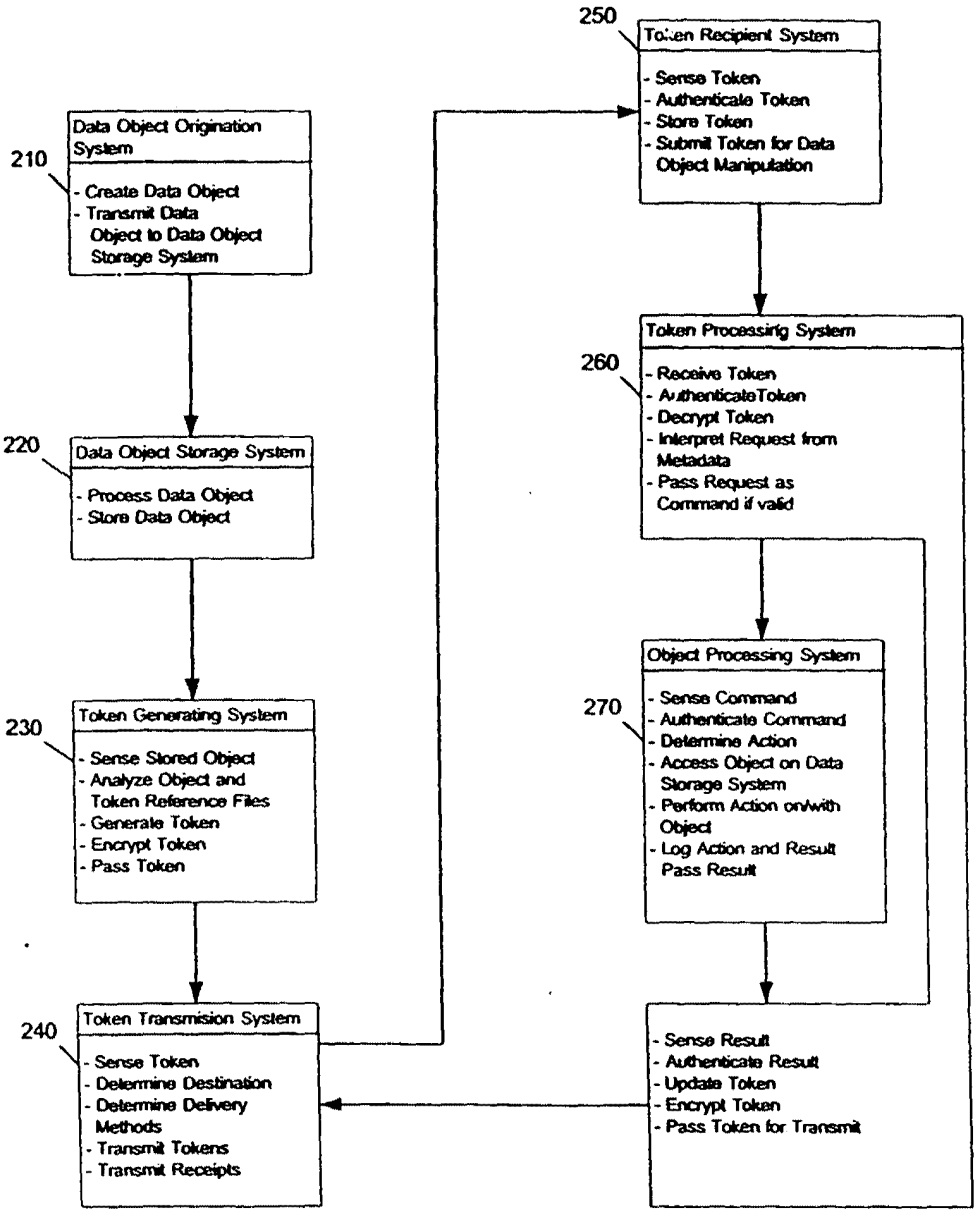

250
Token Recipient System
- Sense Token
- Authenticate Token
- Store Token
- Submit Token for Data Object Manipulation
Data Object Origination System
210
- Create Data Object
- Transmit Data Object to Data Object Storage System
Token Processing System
260
- Receive Token
- AuthenticateToken
- Decrypt Token
- Interpret Request from Metadata
- Pass Request as Command if valid
220
Data Object Storage System
- Process Data Object
- Store Data Object
Object Processing System
270
- Sense Command
- Authenticate Command
- Determine Action
- Access Object on Data Storage System
- Perform Action on/with Object
- Log Action and Result
Pass Result
230
Token Generating System
- Sense Stored Object
- Analyze Object and Token Reference Files
- Generate Token
- Encrypt Token
- Pass Token
240
Token Transmision System
- Sense Token
- Determine Destination
- Determine Delivery Methods
- Transmit Tokens
- Transmit Receipts
- Sense Result
- Authenticate Result
- Update Token
- Encrypt Token
- Pass Token for Transmit

Figure 2

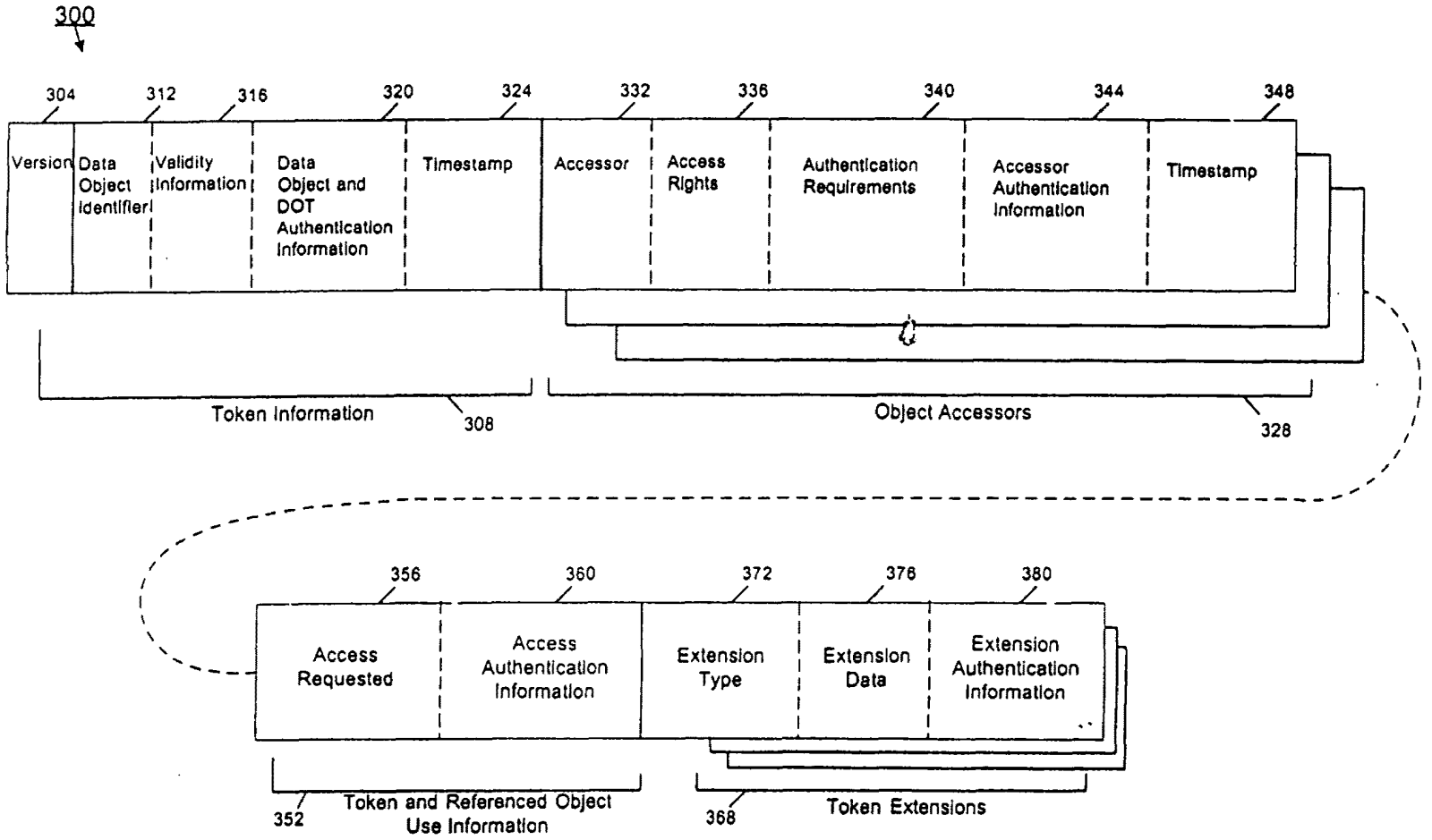


Figure 3

SUBSTITUTE SHEET (RULE 26)

SUBSTITUTE SHEET (RULE 26)

410

```
DataObjectToken::= SEQUENCE {
        version         OCTET STRING,
        tokenInfo       DataObjectTokenInfo,
        accessors       DataObjectTokenAccessors,
        useInfo         DataObjectTokenUseInfo OPTIONAL,
        objectExt       DataObjectTokenExtensions OPTIONAL
        }
```

420

```
DataObjectTokenInfo ::= SEQUENCE {
        vdo_id          OCTET STRING,
        validFrom       UniversalTime,
        validTo         UniversalTime
        extAttributes   InfoExtAttributes,
        signerInfo      PKCS_7_SignerInfo,
                            // sig on previous
attributes
        signerCert      PKCS_6_Certificate,
        timeStamp       VerifiedTimeStamp,
        }

InfoExtAttributes ::= Attributes

Attributes ::= SET OF Attributes
Attribute ::= SEQUENCE {
        key     OCTET STRING,
        value OCTET STRING
        }
```

430

```
DataObjectTokenAccessors ::= SET OF
                        DataObjectTokenAccessor
DataObjectTokenAccessor ::= SEQUENCE {
        accessor                PKCS_6_Certificate,
        accessRights            AccessRights,
        authRequirements        AuthRequirements
        signerInfo              PKCS_7_SignerInfo,
        signerCert              PKCS_6_Certificate
        timestamp               VerifiedTimeStamp,
        }
```

440

```
AccessRights ::= SET OF AccessRight
AccessRight ::= SEQUENCE {
        type ENUMERATED {
                read_whole,
                read_partial,
                update_whole,
                update_partial,
                propagate_access_same,
                propagate_access_limited
                },
        extended UserObject     // extended info
                                    regarding
                                        partials/limited
        }
```

450

```
AuthRequirements ::= SET OF AuthRequirement
AuthRequirement ::= SEQUENCE {
        type ENUMEPATED {
                basic,          // certificate signature
                                   (always at least
                                   this)
                password,
                biometric,
                external_key,
                other
                },
        extended UserObject     // extended info
                                    regarding other
        }
```

Figure 4

510

```
DataObjectTokenUseInfo ::= SEQUENCE {
        accessRequested         Access,
        authentication          UserObject,
        accessorInfo                    PKCS_7_SignerInfo,
        accessorCert                    PKCS_6_Certificate
        }

Access ::= SEQUENCE {
        type ENUMERATED {
                read_whole,
                read_partial,
                update_whole,
                update_partial
                },
        extended UserObject
                }
```

520

```
DataObjectTokenExtensions ::= SET OF DataObjectTokenExtension
DataObjectTokenExtension ::= SEQUENCE {
        extType         OCTET STRING,
        extData         UserObject,
        signerInfos     PKCS_7_SignerInfos,
        signerCerts     PKCS_6_Certificates
        }

PKCS_7_SignerInfos ::= SET OF PKCS_7_SignerInfo
PKCS_6_Certificates ::= SET OF PKCS_6_Certificate
```

Figure 5

SUBSTITUTE SHEET (RULE 26)

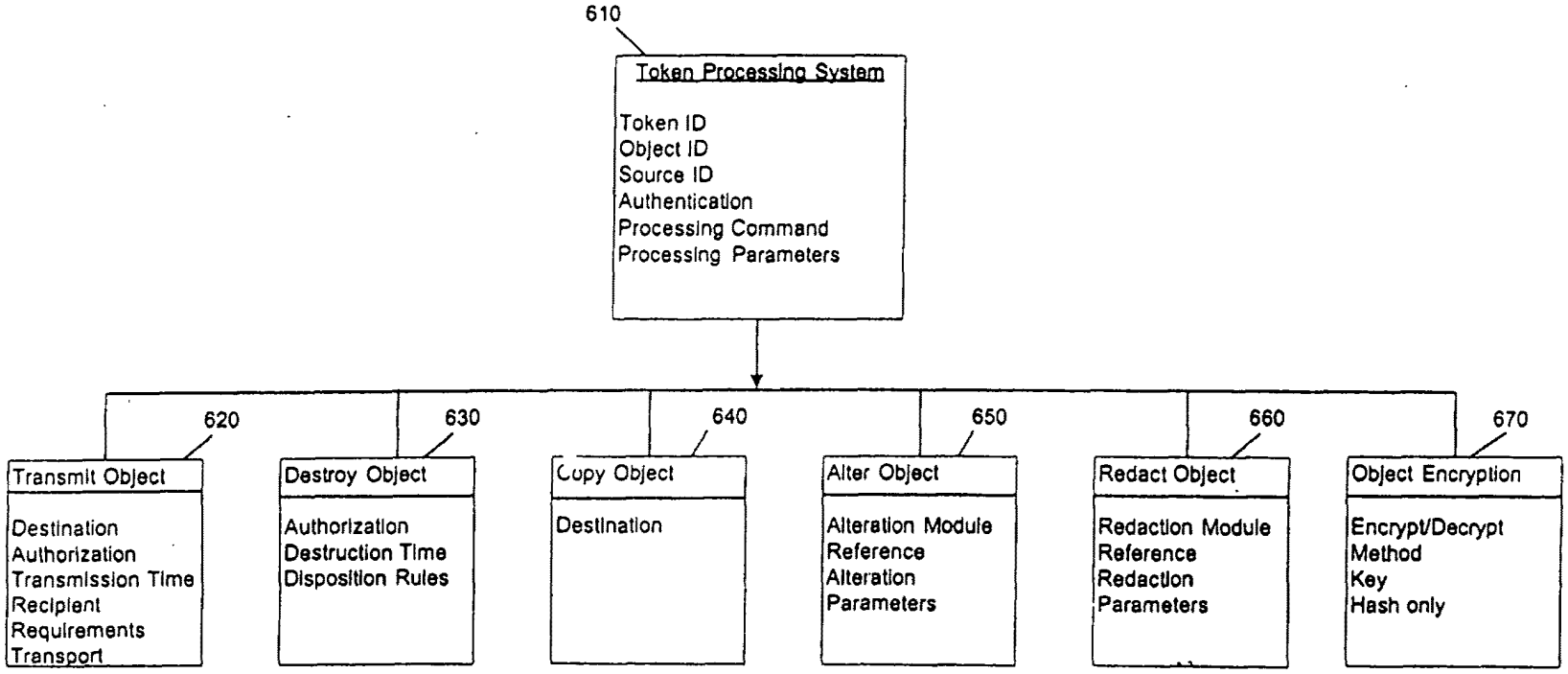


Figure 6

Partial Token Based Data Processing System Command Table

SUBSTITUTE SHEET (RULE 26)

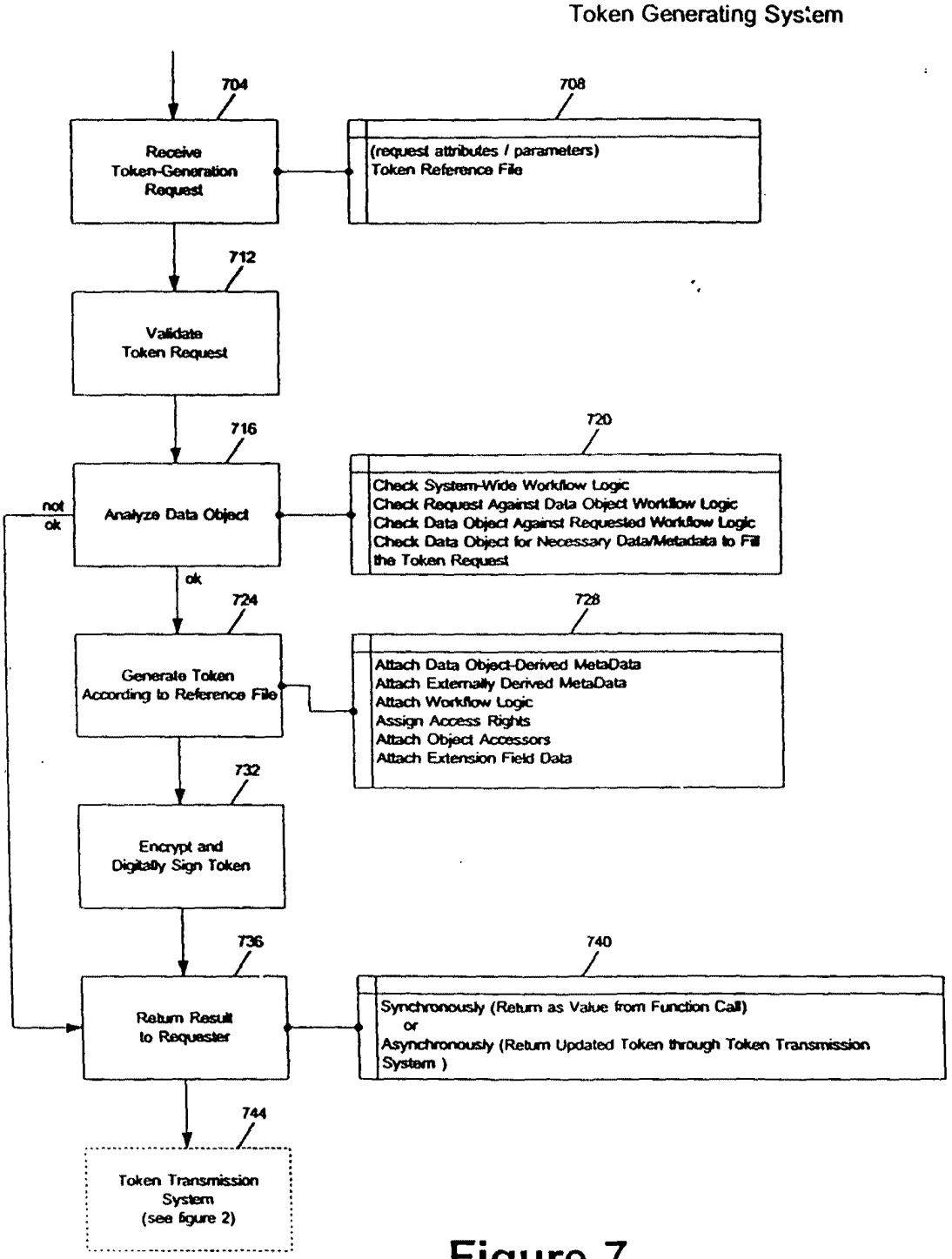

Token Generating System
704
Receive Token-Generation Request
708
(request attributes / parameters)
Token Reference File
712
Validate Token Request
716
Analyze Data Object
not ok
ok
720
Check System-Wide Workflow Logic
Check Request Against Data Object Workflow Logic
Check Data Object Against Requested Workflow Logic
Check Data Object for Necessary Data/Metadata to Fill the Token Request
724
Generate Token According to Reference File
728
Attach Data Object-Derived MetaData
Attach Externally Derived MetaData
Attach Workflow Logic
Assign Access Rights
Attach Object Accessors
Attach Extension Field Data
732
Encrypt and Digitally Sign Token
736
Return Result to Requester
740
Synchronously (Return as Value from Function Call)
or
Asynchronously (Return Updated Token through Token Transmission System)
744
Token Transmission System
(see figure 2)

Figure 7

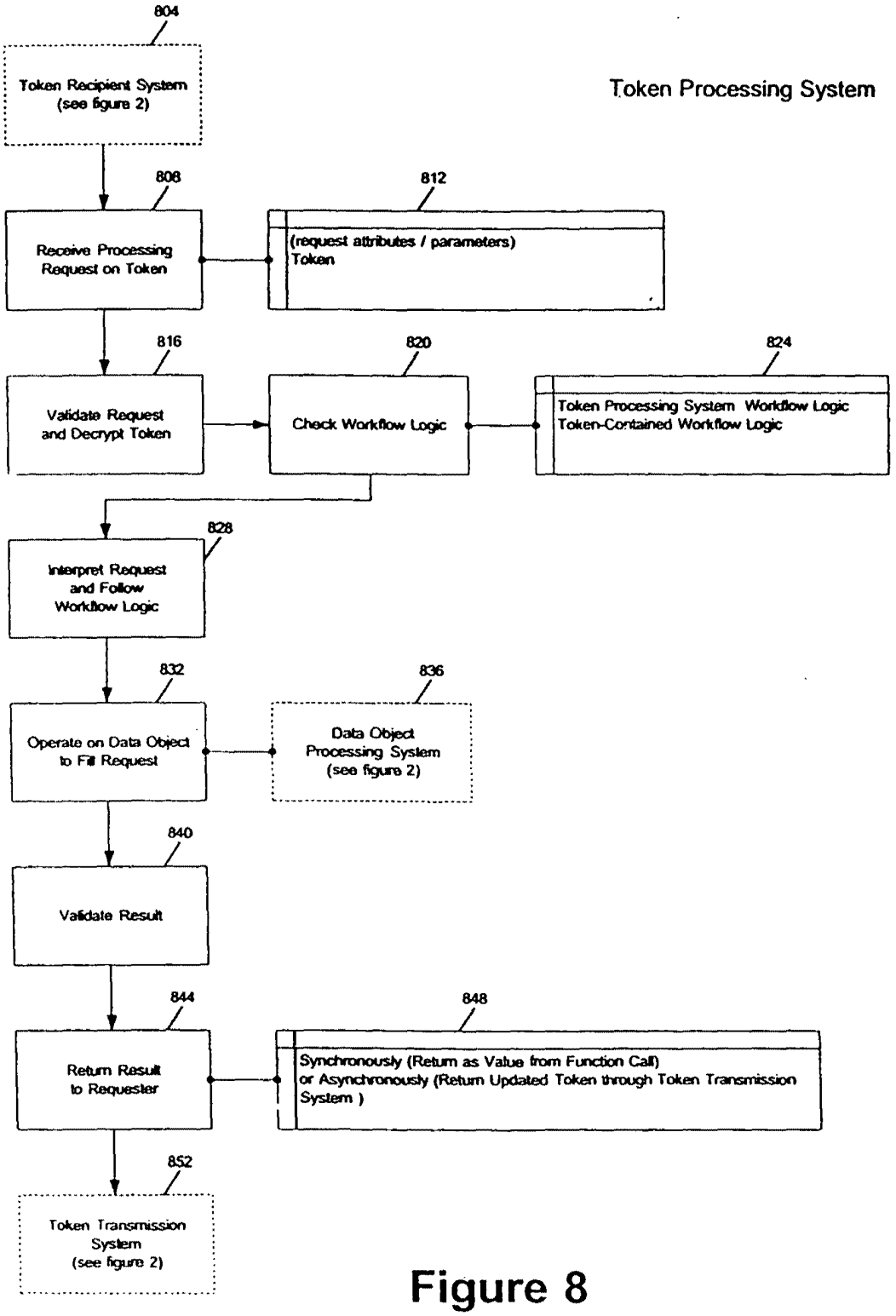

804
Token Recipient System
(see figure 2)
Token Processing System
808
Receive Processing
Request on Token
812
(request attributes / parameters)
Token
816
Validate Request
and Decrypt Token
820
Check Workflow Logic
824
Token Processing System Workflow Logic
Token-Contained Workflow Logic
828
Interpret Request
and Follow
Workflow Logic
832
Operate on Data Object
to Fill Request
836
Data Object
Processing System
(see figure 2)
840
Validate Result
844
Return Result
to Requester
848
Synchronously (Return as Value from Function Call)
or Asynchronously (Return Updated Token through Token Transmission
System)
852
Token Transmission
System
(see figure 2)

Figure 8

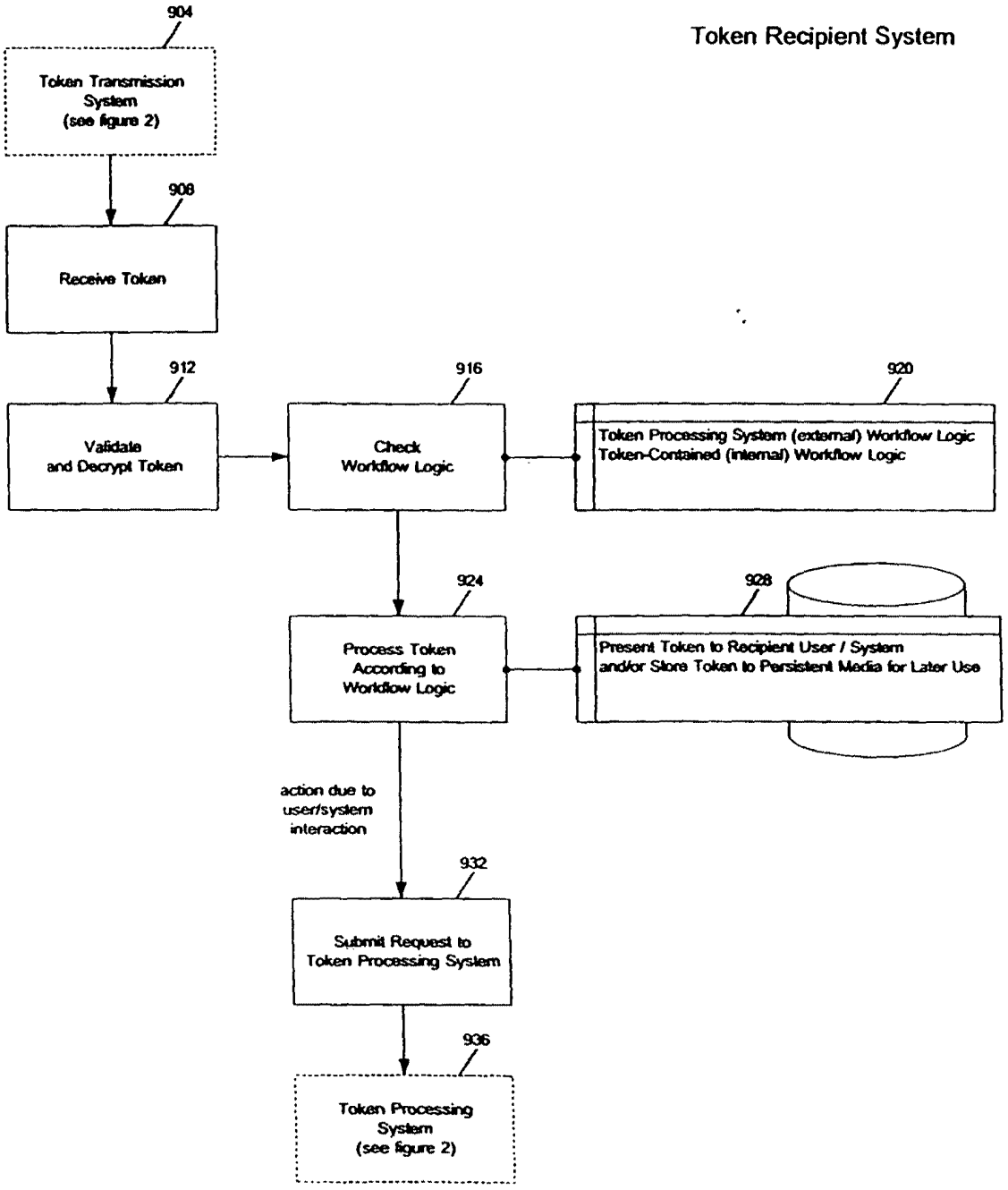

Token Recipient System
904
Token Transmission System (see figure 2)
908
Receive Token
912
Validate and Decrypt Token
916
Check Workflow Logic
920
Token Processing System (external) Workflow Logic
Token-Contained (internal) Workflow Logic
924
Process Token According to Workflow Logic
928
Present Token to Recipient User / System and/or Store Token to Persistent Media for Later Use
action due to user/system interaction
932
Submit Request to Token Processing System
936
Token Processing System (see figure 2)

Figure 9

Ref #	Hits	Search Query	DBs	Default Operator	Plurals	Time Stamp
L1	1040	security adj clearance	US-PGPUB; USPAT	OR	ON	2005/07/27 22:58
L2	268	L1 near5 level	US-PGPUB; USPAT	OR	ON	2005/07/27 22:58
L3	87	L2 and @ad < "20010620"	US-PGPUB; USPAT	OR	ON	2005/07/27 22:58
L4	42	L3 and document	US-PGPUB; USPAT	OR	ON	2005/07/27 22:58
L5	32	L4 and (billing or payment or pay or charge or bill)	US-PGPUB; USPAT	OR	ON	2005/07/27 22:58
L6	5	(US-20020116641-$).did. or (US-6092101-$ or US-5867651-$ or US-6073165-$ or US-6584454-$). did.	US-PGPUB; USPAT	OR	ON	2005/07/27 22:58
L7	32	L5 and (billing or payment or pay or charge or bill)	US-PGPUB; USPAT	OR	ON	2005/07/27 22:59
L8	2	L6 and (billing or payment or pay or charge or bill)	US-PGPUB; USPAT	OR	ON	2005/07/27 23:01
L9	2	L6 and (bill$3 pay$4 or charg$3)	US-PGPUB; USPAT	OR	ON	2005/07/27 23:05
L10	15	(sanitiz$4 or innoculat$5) near3 (email or message)	US-PGPUB; USPAT	OR	ON	2005/07/27 23:05
L11	6	L10 and @ad < "20010620"	US-PGPUB; USPAT	OR	ON	2005/07/27 23:05
L12	4	L11 and (bill$3 pay$4 or charg$3)	US-PGPUB; USPAT	OR	ON	2005/07/27 23:12
L13	634	L1 and (bill$3 pay$4 or charg$3)	US-PGPUB; USPAT	OR	ON	2005/07/27 23:12
L14	6	(US-20020116641-$).did. or (US-6757830-$ or US-6092101-$ or US-5867651-$ or US-6073165-$ or US-6400810-$).did.	US-PGPUB; USPAT	OR	ON	2005/07/27 23:12
L15	1	L14 and (bill$3 pay$4 or charg$3)	US-PGPUB; USPAT	OR	ON	2005/07/27 23:14
L16	4	L11 and (bill$3 pay$4 or charg$3)	US-PGPUB; USPAT	OR	ON	2005/07/27 23:14
L17	360	email near2 filter$5	US-PGPUB; USPAT	OR	ON	2005/07/27 23:20
L18	75	L17 and charg$3	US-PGPUB; USPAT	OR	ON	2005/07/27 23:20
L19	9	l18 and email.ti.	US-PGPUB; USPAT	OR	ON	2005/07/27 23:30
L20	702	insert near3 advertisement	US-PGPUB; USPAT	OR	ON	2005/07/27 23:57

L22	1493	insert$4 near3 advertisement	US-PGPUB; USPAT	OR	ON	2005/07/27 23:57
L23	2827	(message or email) near3 advertisement	US-PGPUB; USPAT	OR	ON	2005/07/27 23:59
L24	242	l22 and l23	US-PGPUB; USPAT	OR	ON	2005/07/27 23:31
L25	107	l24 and @ad < "20010620"	US-PGPUB; USPAT	OR	ON	2005/07/27 23:56
L26	16	l25 and (email or message).ti.	US-PGPUB; USPAT	OR	ON	2005/07/27 23:32
L27	45	l25 and url	US-PGPUB; USPAT	OR	ON	2005/07/27 23:56
L37	1583	(insert$3 or append$3) near3 advertisement	US-PGPUB; USPAT	OR	ON	2005/07/27 23:58
L41	2198	(message or email) near3 url	US-PGPUB; USPAT	OR	ON	2005/07/27 23:59
L42	2428	(message or email) near3 (uri or url)	US-PGPUB; USPAT	OR	ON	2005/07/28 00:00
L44	2198	l41 and l42	US-PGPUB; USPAT	OR	ON	2005/07/28 00:00
L46	44	l37 and l42	US-PGPUB; USPAT	OR	ON	2005/07/28 00:11
L47	20	l46 and @ad < "20010620"	US-PGPUB; USPAT	OR	ON	2005/07/28 00:52
L48	18	(email or messsage) near3 (append$4 or insert$4) near5 (url or ad or advertisement)	US-PGPUB; USPAT	OR	ON	2005/07/28 00:15
L49	7	l48 and @ad < "20010620"	US-PGPUB; USPAT	OR	ON	2005/07/28 00:15
L50	221	vendor near3 url	US-PGPUB; USPAT	OR	ON	2005/07/28 00:52
L51	101	l50 and @ad < "20010620"	US-PGPUB; USPAT	OR	ON	2005/07/28 00:53
L52	36	l51 and advertisement	US-PGPUB; USPAT	OR	ON	2005/07/28 00:58
L53	13	l52 and (advertisement same url)	US-PGPUB; USPAT	OR	ON	2005/07/28 01:22
L55	325	715/511.ccls.	US-PGPUB; USPAT	OR	ON	2005/07/28 01:23
L56	3003	705/14.ccls.	US-PGPUB; USPAT	OR	ON	2005/07/28 01:23
L57	3143	709/219.ccls.	US-PGPUB; USPAT	OR	ON	2005/07/28 01:24
L58	1606	709/218.ccls.	US-PGPUB; USPAT	OR	ON	2005/07/28 01:24

L59	167	709/244.ccls.	US-PGPUB; USPAT	OR	ON	2005/07/28 01:24
L62	140	712/201.ccls.	US-PGPUB; USPAT	OR	ON	2005/07/28 01:25
L64	208	380/59.ccls.	US-PGPUB; USPAT	OR	ON	2005/07/28 01:26
L69	99	704/5.ccls.	US-PGPUB; USPAT	OR	ON	2005/07/28 01:27
L70	218	704/4.ccls.	US-PGPUB; USPAT	OR	ON	2005/07/28 01:28
L73	2247	"704"/1-10.ccls.	US-PGPUB; USPAT	OR	ON	2005/07/28 01:28
S1	8477	email and @ad < "20010620"	US-PGPUB; USPAT	OR	ON	2005/07/25 11:56
S2	1474	S1 and attachment	US-PGPUB; USPAT	OR	ON	2005/07/25 11:33
S4	1256	S2 and stor$5 and file	US-PGPUB; USPAT	OR	ON	2005/07/25 11:34
S5	328	email.ti.	US-PGPUB; USPAT	OR	ON	2005/07/25 11:34
S6	33	S5 and S4	US-PGPUB; USPAT	OR	ON	2005/07/25 11:55
S7	883	file near3 split$5	US-PGPUB; USPAT	OR	ON	2005/07/25 19:10
S8	495	S7 and @ad < "20010620"	US-PGPUB; USPAT	OR	ON	2005/07/25 19:20
S10	5033	file near3 (split$5 or break$5 or divid$5)	US-PGPUB; USPAT	OR	ON	2005/07/25 19:19
S12	129	file near3 (split$5 or break$5 or divid$5) near5 distribut$5	US-PGPUB; USPAT	OR	ON	2005/07/25 19:19
S14	43	S12 and @ad < "20010620"	US-PGPUB; USPAT	OR	ON	2005/07/25 19:09
S15	6795	(distributed or decentralized) near3 storage	US-PGPUB; USPAT	OR	ON	2005/07/25 19:10
S16	15700	storage near3 distribut$5	US-PGPUB; USPAT	OR	ON	2005/07/25 19:20
S17	6795	storage near3 (distributed or decentralized)	US-PGPUB; USPAT	OR	ON	2005/07/25 19:20
S18	3444	S17 and @ad < "20010620"	US-PGPUB; USPAT	OR	ON	2005/07/25 19:20
S19	1896	S18 and file	US-PGPUB; USPAT	OR	ON	2005/07/25 19:21
S20	184	S19 and parse	US-PGPUB; USPAT	OR	ON	2005/07/25 19:22

S21	17	(US-20020112008-$ or US-20010047400-$ or US-20030028587-$ or US-20020095674-$ or US-20020065919-$).did. or (US-6781609-$ or US-5499293-$ or US-6401091-$ or US-6192365-$ or US-5568554-$ or US-6430563-$ or US-6289375-$ or US-6813645-$ or US-6424989-$ or US-6415278-$ or US-6886130-$ or US-6446048-$). did.	US-PGPUB; USPAT	OR	ON	2005/07/25 21:05
S22	4	S21 and encrypt$5	US-PGPUB; USPAT	OR	ON	2005/07/25 21:29
S23	17	S21 and distributed	US-PGPUB; USPAT	OR	ON	2005/07/25 21:29
S24	8	(US-20020065919-$ or US-20020112008-$ or US-20010047400-$).did. or (US-6430563-$ or US-6886130-$ or US-6401091-$ or US-5568554-$ or US-6415278-$).did.	US-PGPUB; USPAT	OR	ON	2005/07/25 22:00
S25	6	(US-20020065919-$ or US-20020112008-$ or US-20010047400-$).did. or (US-6430563-$ or US-6401091-$ or US-5568554-$).did.	US-PGPUB; USPAT	OR	ON	2005/07/25 22:01
S26	6	(US-20020065919-$ or US-20020112008-$ or US-20010047400-$).did. or (US-6430563-$ or US-6401091-$ or US-5568554-$).did.	US-PGPUB; USPAT	OR	ON	2005/07/25 22:09
S27	3	(US-20020065919-$ or US-20020112008-$ or US-20010047400-$).did.	US-PGPUB	OR	ON	2005/07/25 22:11
S28	3	S27 and distributed	US-PGPUB; USPAT	OR	ON	2005/07/25 23:47
S29	360	email near2 filter$5	US-PGPUB; USPAT	OR	ON	2005/07/27 23:19
S30	101	S29 and @ad < "20010620"	US-PGPUB; USPAT	OR	ON	2005/07/25 22:40
S31	101	S30 and filter$5	US-PGPUB; USPAT	OR	ON	2005/07/25 22:39
S32	967215	filter$5	US-PGPUB; USPAT	OR	ON	2005/07/25 22:39
S33	360	S32 and S29	US-PGPUB; USPAT	OR	ON	2005/07/25 22:39
S34	101	S33 and @ad < "20010620"	US-PGPUB; USPAT	OR	ON	2005/07/25 22:40

S35	7	(US-20010047400-$ or US-20020112008-$).did. or (US-5958006-$ or US-6397261-$ or US-6671818-$ or US-6816884-$ or US-5944787-$).did.	US-PGPUB; USPAT	OR	ON	2005/07/25 23:46
S36	3764	content near2 filter	US-PGPUB; USPAT	OR	ON	2005/07/25 23:47
S37	582	S36 and security	US-PGPUB; USPAT	OR	ON	2005/07/25 23:47
S38	418	S37 and (extract$5 or censor$5 or delet$5)	US-PGPUB; USPAT	OR	ON	2005/07/25 23:57
S39	159	S38 and @ad < "20010620"	US-PGPUB; USPAT	OR	ON	2005/07/25 23:57
S42	73	S39 and (restor$5 or reconstruct$5)	US-PGPUB; USPAT	OR	ON	2005/07/25 23:54
S44	47	S42 and email	US-PGPUB; USPAT	OR	ON	2005/07/25 23:53
S45	5	S39 and ((restor$5 or reconstruct$5) near3 (message or email))	US-PGPUB; USPAT	OR	ON	2005/07/26 00:04
S46	1701	S36 and (extract$5 or censor$5 or delet$5)	US-PGPUB; USPAT	OR	ON	2005/07/26 00:01
S47	940	S46 and @ad < "20010620"	US-PGPUB; USPAT	OR	ON	2005/07/25 23:57
S48	6	S47 and ((restor$5 or reconstruct$5) near3 (message or email))	US-PGPUB; USPAT	OR	ON	2005/07/25 23:58
S49	7978	(extract$5 or censor$5) near3 (word or character)	US-PGPUB; USPAT	OR	ON	2005/07/26 00:02
S50	11	S49 and (email near3 filter)	US-PGPUB; USPAT	OR	ON	2005/07/26 00:03
S51	1	S50 and @ad < "20010620"	US-PGPUB; USPAT	OR	ON	2005/07/26 00:04
S52	122	S49 and ((email or message or content) near3 filter)	US-PGPUB; USPAT	OR	ON	2005/07/26 00:16
S53	59	S52 and @ad < "20010620"	US-PGPUB; USPAT	OR	ON	2005/07/26 00:16
S54	1	S53 and ((restor$5 or reconstruct$5) near3 (message or email))	US-PGPUB; USPAT	OR	ON	2005/07/26 00:16
S55	8	S53 and ((restor$5 or reconstruct$5) near3 (message or email or data))	US-PGPUB; USPAT	OR	ON	2005/07/26 00:17
S56	474	S49 and ((email or message or content or data or information) near3 filter)	US-PGPUB; USPAT	OR	ON	2005/07/26 00:19

S57	263	S56 and @ad < "20010620"	US-PGPUB; USPAT	OR	ON	2005/07/26 00:19
S58	41	S57 and ((reassembl$4 or restor$5 or reconstruct$5) near3 (message or email or data))	US-PGPUB; USPAT	OR	ON	2005/07/26 00:20
S59	266	S49 and ((email or message or content or information) near3 filter)	US-PGPUB; USPAT	OR	ON	2005/07/26 00:19
S60	141	S59 and @ad < "20010620"	US-PGPUB; USPAT	OR	ON	2005/07/26 00:22
S61	2	S60 and ((reassembl$4 or restor$5 or reconstruct$5) near3 (message or email))	US-PGPUB; USPAT	OR	ON	2005/07/26 00:23
S62	5403	S49 and @ad < "20010620"	US-PGPUB; USPAT	OR	ON	2005/07/26 00:22
S63	47	S62 and ((reassembl$4 or restor$5 or reconstruct$5) near3 (message or email))	US-PGPUB; USPAT	OR	ON	2005/07/26 00:24
S64	9	(US-20010047400-$ or US-20020112008-$).did. or (US-5958006-$ or US-6397261-$ or US-6671818-$ or US-6816884-$ or US-5944787-$ or US-6549957-$ or US-6745231-$).did.	US-PGPUB; USPAT	OR	ON	2005/07/27 23:04
S65	8	extract$3 near3 word near5 email	US-PGPUB; USPAT	OR	ON	2005/07/26 00:39
S66	1	S65 and @ad < "20010620"	US-PGPUB; USPAT	OR	ON	2005/07/26 00:40
S67	58	filter$5 near5 (email or message) near5 (keyword or (key adj word))	US-PGPUB; USPAT	OR	ON	2005/07/26 00:40
S68	17	S67 and @ad < "20010620"	US-PGPUB; USPAT	OR	ON	2005/07/26 00:50
S69	15	(sanitiz$4 or innoculat$5) near3 (email or message)	US-PGPUB; USPAT	OR	ON	2005/07/27 23:14
S70	6	S69 and @ad < "20010620"	US-PGPUB; USPAT	OR	ON	2005/07/27 23:14
S71	8	(US-20020112008-$).did. or (US-6397261-$ or US-5944787-$ or US-6549957-$ or US-6745231-$ or US-6654787-$ or US-6571275-$ or US-5960080-$).did.	US-PGPUB; USPAT	OR	ON	2005/07/26 00:53
S72	1040	security adj clearance	US-PGPUB; USPAT	OR	ON	2005/07/27 11:41
S73	459	S72 and @ad < "20010620"	US-PGPUB; USPAT	OR	ON	2005/07/27 11:04
S74	322	decrypt$5.ti.	US-PGPUB; USPAT	OR	ON	2005/07/27 11:04
S75	4	S74 and S72	US-PGPUB; USPAT	OR	ON	2005/07/27 11:04

S76	1402	content adj filter	US-PGPUB; USPAT	OR	ON	2005/07/27 11:56
S77	965	S76 and @ad < "20010620"	US-PGPUB; USPAT	OR	ON	2005/07/27 11:58
S78	1	S77 and email.ti.	US-PGPUB; USPAT	OR	ON	2005/07/27 11:42
S79	10	S77 and (message or email).ti.	US-PGPUB; USPAT	OR	ON	2005/07/27 11:42
S80	6680	filter near5 word	US-PGPUB; USPAT	OR	ON	2005/07/27 11:56
S83	104	S77 and (email or message)	US-PGPUB; USPAT	OR	ON	2005/07/27 13:49
S84	75	S83 and word	US-PGPUB; USPAT	OR	ON	2005/07/27 12:00
S85	9	S83 and S80	US-PGPUB; USPAT	OR	ON	2005/07/27 12:00
S86	5	(US-20020112008-$).did. or (US-6397261-$ or US-6745231-$ or US-6654787-$ or US-5960080-$). did.	US-PGPUB; USPAT	OR	ON	2005/07/27 13:33
S87	4828	filter$5 near5 (email or message)	US-PGPUB; USPAT	OR	ON	2005/07/27 13:49
S88	2491	S87 and @ad < "20010620"	US-PGPUB; USPAT	OR	ON	2005/07/27 15:09
S89	111	S88 and ((keyword or word) near5 filter$5)	US-PGPUB; USPAT	OR	ON	2005/07/27 13:51
S90	30	S89 and ((add$3 or delet$3 or remov$3) near5 (keyword or word))	US-PGPUB; USPAT	OR	ON	2005/07/27 14:28
S91	10	S90 and ((add$3 or delet$3 or remov$3) near5 filter$4)	US-PGPUB; USPAT	OR	ON	2005/07/27 13:52
S92	127	S88 and ((add$3 or delet$3 or remov$3) near5 (keyword or word))	US-PGPUB; USPAT	OR	ON	2005/07/27 14:44
S93	22	S92 and (email or message).ti.	US-PGPUB; USPAT	OR	ON	2005/07/27 14:29
S94	76674	(stor$3 or sav$3) near3 (configur$6 or keyword)	US-PGPUB; USPAT	OR	ON	2005/07/27 15:10
S95	27	S92 and S94	US-PGPUB; USPAT	OR	ON	2005/07/27 14:42
S96	394	S88 and ((add$3 or delet$3 or remov$3) near5 filter)	US-PGPUB; USPAT	OR	ON	2005/07/27 14:45
S97	24148	filter near3 configur$6	US-PGPUB; USPAT	OR	ON	2005/07/27 15:11
S98	73	S96 and S97	US-PGPUB; USPAT	OR	ON	2005/07/27 14:45
S99	6	S98 and (email or message).ti.	US-PGPUB; USPAT	OR	ON	2005/07/27 15:11

S100	6	(US-20020112008-$).did. or (US-6397261-$ or US-6745231-$ or US-6654787-$ or US-5960080-$ or US-6571234-$).did.	US-PGPUB; USPAT	OR	ON	2005/07/27 14:48
S101	34077	(stor$3 or sav$3) near5 filter$3	US-PGPUB; USPAT	OR	ON	2005/07/27 15:10
S102	519	S88 and S101	US-PGPUB; USPAT	OR	ON	2005/07/27 15:10
S103	106	S97 and S102	US-PGPUB; USPAT	OR	ON	2005/07/27 15:11
S104	16	S103 and (email or message).ti.	US-PGPUB; USPAT	OR	ON	2005/07/27 18:12
S105	6	(US-20020116641-$).did. or (US-6757830-$ or US-6092101-$ or US-5867651-$ or US-6073165-$ or US-6400810-$).did.	US-PGPUB; USPAT	OR	ON	2005/07/27 15:27
S106	1040	security adj clearance	US-PGPUB; USPAT	OR	ON	2005/07/27 18:43
S107	263	S106 near3 level	US-PGPUB; USPAT	OR	ON	2005/07/27 18:12
S108	268	S106 near5 level	US-PGPUB; USPAT	OR	ON	2005/07/27 18:12
S109	87	S108 and @ad < "20010620"	US-PGPUB; USPAT	OR	ON	2005/07/27 18:55
S111	21	S109 and email	US-PGPUB; USPAT	OR	ON	2005/07/27 18:50
S112	1	S111 and (email or message or document).ti.	US-PGPUB; USPAT	OR	ON	2005/07/27 18:14
S113	459	S106 and @ad < "20010620"	US-PGPUB; USPAT	OR	ON	2005/07/27 18:43
S114	231	S113 and document	US-PGPUB; USPAT	OR	ON	2005/07/27 18:43
S115	42	S109 and document	US-PGPUB; USPAT	OR	ON	2005/07/27 22:57

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.
10/155,192	05/23/2002	Ron M. Redlich	6851-2-CIP-3	9510

4897 7590 08/11/2005

ROBERT C. KAIN, JR.
750 SOUTHEAST THIRD AVENUE
SUITE 100
FT LAUDERDALE, FL 333161153

EXAMINER	
CHUNG, JI YONG DAVID	
ART UNIT	**PAPER NUMBER**
2143	

DATE MAILED: 08/11/2005

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

Office Action Summary	Application No.	Applicant(s)	
	10/155,192	REDLICH ET AL.	
	Examiner	Art Unit	
	Ji-Yong D. Chung	2143	

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

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If the period for reply specified above is less than thirty (30) days, a reply within the statutory minimum of thirty (30) days will be considered timely.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *23 May 2002*.

2a)☐ This action is **FINAL**. 2b)☒ This action is non-final.

3)☐ Since this application is in condition for allowance 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.

Disposition of Claims

4)☒ Claim(s) *1-27* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-27* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☐ The drawing(s) filed on _____ is/are: a)☐ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☒ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08) Paper No(s)/Mail Date 05/23/02; 1/27/2003 JDC
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____.
5) ☐ Notice of Informal Patent Application (PTO-152)
6) ☐ Other: _____.

JDC

DETAILED ACTION

Claim Rejections - 35 USC § 103

1. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

2. **Claims 1, 2, 6-12, 16, 17, and 21-27** are rejected under 35 U.S.C. 103(a) as being unpatentable over Fahlman et al (Pat. No. 5,960,080, Fahlman hereinafter) in view Kirshenbaum (Pat No. 6,602,298).

In reference to **claim 1,** Fahlman shows a *method of securing data having one or more security sensitive words, characters or icons in a computer system with memories designated as a remainder store and an extract store* [See lines 27-67, column 3] comprising:

extracting said security sensitive words, characters or icons from said data to obtain extracted data and remainder data therefrom [See lines 61-67, column 3. See also Fig 2.];

storing said extracted data and said remainder data in said extract store and said remainder store, respectively [See lines 61-67, column 3. See Fig. 2] and,

permitting reconstruction of said data via said extracted data and remainder data [See lines 27-32, column 5. The merging is "permitted" at the destination terminal].

Fahlman does not show, but Kirshenbaum shows that "reconstruction" of the message is permitted *only in the presence of a predetermined security clearance* [See lines 34-62, column 5. The user (sender or recipient) privilege on the messages is based on security clearance].

It would have been obvious to one of ordinary skill in the art at the time of the invention to add a feature that allows merging of the segregated information based on the security clearance of the intended receiver, because Kirshenbaum illustrates the general idea of accessing the original message based on security clearance level to increase security. See lines 34-62, column 5.

With respect to **claim 2**, Fahlman shows the *method including filtering said data while extracting*. The "extraction" is filtering; it filters out the security sensitive words.

With respect to **claim 6**, Fahlman does not show *a method including establishing a plurality of security levels each with a respective security clearance, said security sensitive words, characters or icons being correlated with said plurality of security levels with subsets of said security sensitive words, characters or icons and the step of* permitting reconstruction including the step of permitting a *plurality of partial reconstructions of said data in the presence of respective ones of said plurality of security clearance levels.*

To be more precise, what Fahlman does not show is not that a document is separated and recombined, but that document is not conveyed at multiple security clearance levels.

Conceptually, Kirshenbaum shows "reconstruction" of portions of document based on security clearance level. Some portions require greater clearance than others. See lines 34-62,

column 5. The hierarchy of clearance allows those with greater clearance can always access those of lower clearance. If a document has two portions, one of which required greater security clearance than the preceding one, user of at least one clearance level would be required to "reconstruct" the relevant portion.

It would have been obvious to one of ordinary skill in the art at the time of the invention to apply Kirshenbaum to create multiple portion document with multiple security clearance level, with the mechanism for associating each filter with different security clearance level and with the mechanism for repeated filtering, because as explained in lines 47-59, column 1, Kirshenbaum's concept addresses inconvenience of generally passing the whole document from one person to another. The combination of Fahlman and Kirshenbaum can pass portions of document. The implementation of multiple filters (each of different security level) flows from the necessarily of partitioning document into multiple security level portions.

With respect to **claim 7**, all of its limitations have been discussed with respect to claim 1, except that word "parsing" is used. However, Fahlman shows "parsing." See lines 34-47, column 3.

With respect to **claim 8**, Fahlman shows parsing *parses the data granularly*. See lines 34-47 in column 3, which indicates that data is parsed at word level.

With respect to **claim 9**, Fahlman shows that the step of *parsing* uses said parsing algorithm which is based on one or more of a bit count, *a word* [See line 34-47, column 3] a

word count, a page count, a line count, a paragraph count, an identifiable document characteristic, *an identifiable word characteristic* [See lines 43-37, column 3], an identifiable icon characteristic, an identifiable data object characteristic, *capital letters* [See line 41-43, column 3], italics, and underline.

With respect to **claim 10**, Fahlman shows the method wherein *said data is financial data and the step of parsing uses said parsing algorithm based upon one of characters in said data and character count in said data.* See lines 27-37, column 4. Financial data is mentioned. See lines 43-46, column 3. It mentions dollar sign (which is "one of characters." Numbers are "character count." See lines 47-52, column 3 for dates, which include "character count." Dates are included as part of transaction date. See lines 27-37, column 4.

With refernce to **claim 11**, Fahlman does not shows *establishing a plurality of security levels each with a respective security clearance, increasingly greater reconstruction if said data being correlated with increasingly higher levels of said plurality of security levels, and the step of permitting reconstruction including the step of permitting a plurality of partial reconstructions of said data in the presence of respective ones of said plurality of security clearance levels.*

Kirshenbaum shows "reconstruction" of document with increasingly higher levels of security levels and the reconstruction requires respective ones of the security clearances. See lines 34-62, column 5.

It would have been obvious to one of ordinary skill in the art at the time of the invention to apply Kirshenbaum to create multiple portion document with multiple security clearance level,

with the mechanism for associating each filter with different security clearance level and with the mechanism for repeated filtering, because as explained in lines 47-59, column 1, Kirshenbaum's concept addresses inconvenience of generally passing the whole document from one person to another. The combination of Fahlman and Kirshenbaum can pass portions of a document. The implementation of multiple filters (each of different security level) flows from the necessarily of partitioning document into multiple security level portions.

With respect to **claim 12**, Fahlman shows the method including a further data security step of one from the group of *storing said data without encryption* [See step 213, Fig. 2], storing said data with encryption and destroying said data; *said further data security step being independent of said parsing, storing and permitting reconstruction* [See steps 217 and 219 in Fig. 2. Parsing step is 203. Storing step is 213, reconstruction is decrypting step (shown in Fig. 3) together with steps 217 and 219. The steps are independent in the sense that each step does not depend on the other steps themselves, but only on their outputted data.].

Claims 16, 17, 21-27 incorporate all the corresponding limitations of claims 1, 2, and 6-12, but in computer product (software) form rather than in method form. The reasons for the rejection of claims 1, 2, 6-12 apply to claims 16, 17, and 22-27.

3. **Claims 3-5 and 18-20** are rejected under 35 U.S.C. 103(a) as being unpatentable over Fahlman and Kirshenbaum and in further in view of Mastrianni (Pub. No. 2002/ 0116641).

With respect to **claim 3**, neither Falhlman nor Kirshenbaum shows, but Mastrianni shows the method including

defining the filter with a group of security sensitive words [See paragraph 0035, which indicates the step of configuring ("defining") the Semantic Engine ("defining"). See paragraph 0039, which shows that filtering action based on "security sensitive" words.], characters or icons and

the method includes one of storing said filter or destroying said filter after storing said extracted data [Storing is part of "configuration" procedure, in the context of software application].

It would have been obvious to one of ordinary skill in the art at the time of the invention to apply the steps of defining the filter and storing it, because the process would allow multiple use of the filters in different instances. In addition, the mechanism would allow one to improve and modify the filters over time.

With respect to **claim 4**, Fahlman shows *encrypting one or both of said extracted data and said remainder data*. See Fig. 6.

With respect to **claim 5**, Fahlman shows wherein *decrypting one or both of said extracted data and remainder data*. See Fig. 6. Note that "decrypting 309 in Fig. 6 maybe viewed as part of "reconstruction" step.

Claims 18-20 incorporate all the corresponding limitations of claims 3-5, but in computer product (software) form rather than in method form. The reasons for the rejection of claims 3-5 apply to claims 18-20.

4. **Claims 13 and 15** are rejected under 35 U.S.C. 103(a) as being unpatentable over Fahlman and Kirshenbaum and in further in view of Gupta et al. (Pat. No. 6,487,538, Gupta hereinafter)

With reference to **claim 13**, neither Fahlman nor Kirshenbaum shows its limitations. However, Gupta shows *displaying an advertisement prior to one or the other or both of said steps of extracting and permitting reconstruction.* [Lines 10-22, column 6]

It would have been obvious to one of ordinary skill in the art at the time of the invention to use Gupta's system to insert advertisement into Fahlman and Kirshenbaum's combination that would use an application server to render the document delivery service, because Gupta's system allows such service to charge for advertisements and obtain additional revenue.

With reference to **claim 15**, neither Fahlman nor Kirshenbaum shows the limitation or charging for *extracting, storing or permitting reconstruction* (all of which have been shown as explained in reference to claim 1).

Gupta illustrates the general concept of charging for its services. See lines 36-47, column 16.

It would have been obvious to one of ordinary skill in the art at the time of the invention to charge clents, as Gupta shows, for using Fahlman and Kirshenbaum combination services, because charging the clients would allows the service provider to generate revenue.

5. **Claim 14** is rejected under 35 U.S.C. 103(a) as being unpatentable over Fahlman and Kirshenbaum and Gupta in view of Hoyle (Pat. No. 6,771,290).

With reference to **claim 14**, neither Fahlman, Kirshenbaum, nor Gupta shows its limitation.

Hoyle shows a hyperlink to a vendor's web site with said advertisement. See lines 29-41, column 17.

It would have been obvious to one of ordinary skill in the art at the time of the invention to use a banner advertisement with URL as explained in Hoyle, because such banner advertisements are much smaller in size than a full page advertisement, and therefore would provide savings in advertisement cost.

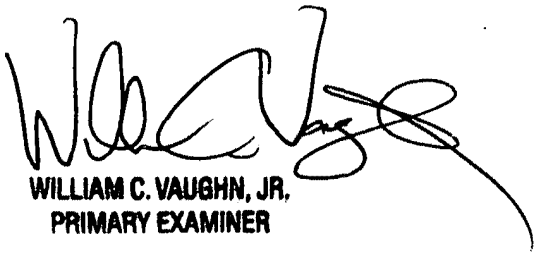
Conclusion

6. Any inquiry concerning this communication or earlier communications from the examiner should be directed to Ji-Yong D. Chung whose telephone number is (571) 272-7988. The examiner can normally be reached on Monday-Friday 9:30-6:00.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, David Wiley can be reached on (571) 272-3923. The fax phone number for the organization where this application or proceeding is assigned is 703-872-9306.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

Ji-Yong D. Chung
Patent Examiner
Art Unit: 2143



WILLIAM C. VAUGHN, JR.
PRIMARY EXAMINER

JDC

PTO/SB/42 (10-96)

JC971 U.S. PTO
10/155192
05/23/02

INFORMATION DISCLOSURE CITATION *(Use several sheets if necessary)*	Docket Number (Optional) 6851-2 CIP-3	Application Number 10 155 192
	Applicant(s) Ron Redlich	
	Filing Date	Group Art Unit 2143

U.S. PATENT DOCUMENTS

Examiner Initial	Document Number	Date	Name	Class	Subclass	Filing Date if Appropriate
JPC	6,192,472 B1	2/20/2001	Garay et al.	713	165	
JPC	6,148,342	11/14/2000	Ho	709	225	
JPC	6,094,483	7/25/2000	Fridrich et al.	380	28	
JPC	6,055,544	4/25/2000	DeRose et al.	707	104	
JPC	5,996,011	11/30/1999	Humes	709	225	
JPC	5,905,980	5/18/1999	Masuichi et al.	707	1	
JPC	5,832,212	11/3/1998	Cragun et al.	395	188.01	
JPC	5,485,474	1/16/1996	Rabin	371	37.1	

FOREIGN PATENT DOCUMENTS

	Document Number	Date	Country	Class	Subclass	Translation YES NO

OTHER DOCUMENTS (Including Author, Title, Date, Pertinent Pages, Etc.)

	~~Ingram i100, Content Security Appliance (2 pages)~~	NO COPY
	Element-Wise XML Encryption, Hiroshi Maruyama and Takeshi ~~Imamura, IBM Research, Tokyo Research Laboratory (4 pages)~~	NO COPY
JPC	Survival Information Storage Systems by Jay J. Wylie, Michael W. Bigrigg, John D. Strunk, Gregory R. Ganger, Han Kiliccote Pradeep K. Khosla (8 pages)	
JPC	ZD Net Interactive iWeek - IBS, SAP: XML to Boost Security Integration (1 page)	
JPC	Myers, A.C. "Mostly-Static Decentralized Information Flow Control" M.I.T. Doctoral Thesis January, 1999	
JPC	Cisco Systems' Intrusion Detection System (IDS) Host Sensor 2.0 product data sheet and technical overview, 10 pgs., July 3, 2000, www.cisco.com	

EXAMINER J Yong Chung DATE CONSIDERED 7/22/2005

JPC

INFORMATION DISCLOSURE CITATION *(Use several sheets if necessary)*	Docket Number (Optional) 6851-2 CIP-3	Application Number 10/155,192
	Applicant(s) Ron Redlich	
	Filing Date	Group Art Unit 2143

U.S. PATENT DOCUMENTS

Examiner Initial	Document Number	Date	Name	Class	Subclass	Filing Date if Appropriate
GPC	5,892,908	4/6/1999	Hughes et al.	395	200.8	
GPC	5,423,032	6/6/1995	Byrd et al.	395	600	
GPC	6,044,375	3/28/2000	Shmueli et al.	707	101	
GPC	5,715,399	2/3/1998	Bezos	395	227	

FOREIGN PATENT DOCUMENTS

	Document Number	Date	Country	Class	Subclass	Translation YES NO

OTHER DOCUMENTS (Including Author, Title, Date, Pertinent Pages, Etc.)

GPC	"Protecting Privacy using the Decentralized Label Model" A. Myers, Cornell Univ. 1/1/2000	
GPC	Mimesweeper Product Data sheet, Resoft Int'l LLC., Nov. 12, 2001 re-soft.com/product/mimeswep.	
GPC	Complete Content Security for Internet Gateways and Mail Servers, Aladdin, July 2000.	
GPC	Aladdin Product announcement for e-Safe Appliance, Aladdin Knowledge Sys., Ltd., Nov. 8, 2001, Aks.com/news/2001/e-safe.	
GPC	Mail Attender 4.1x product white paper by Sherpa Software, Oct., 2001.	
	G Yung Chang	7/21/2005

EXAMINER DATE CONSIDERED

mimori6851-2-cip-3 form ptn-1449

GPC

OIPE JC
JAN 27 2003
PATENT & TRADEMARK

INFORMATION DISCLOSURE CITATION *(Use several sheets if necessary)*	Docket Number (Optional) 6851-2 CIP3	Application Number 10/155,192
	Applicant(s) Ron Redlich	
	Filing Date May 23, 2002	Group Art Unit 2143

U.S. PATENT DOCUMENTS

Examiner Initial	Document Number	Date	Name	Class	Subclass	Filing Date if Appropriate
JNC	5,933,498	8/3/99	Schneck et al.	380	4	
JNC	6,078,907	6/20/00	Lamm	705	40	
JNC	5,581,682	12/3/96	Anderson et al.	395	792	
JNC	US2002/0073311	6/13/02	Brown et al.	713	165	

RECEIVED
JAN 28 2003
Technology Center 2100

FOREIGN PATENT DOCUMENTS

	Document Number	Date	Country	Class	Subclass	Translation YES NO
JNC	WO 00/75779-A2	12/14/00	Lambert			

OTHER DOCUMENTS (Including Author, Title, Date, Pertinent Pages, Etc.)

EXAMINER J. Jung Chang DATE CONSIDERED 7/21/2005

\\Tiger\rck\CLIENTS\Redlich\6851-2-cip3 Form 1449 for Suppl.IDS.wpd

JNC

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

Bib Data Sheet

CONFIRMATION NO. 9510

SERIAL NUMBER	FILING DATE	CLASS	GROUP ART UNIT	ATTORNEY DOCKET NO.
10/155,192	05/23/2002 RULE	709	2143	6851-2-CIP-3

APPLICANTS

Ron M. Redlich, Miami Beach, FL;

Martin A. Nemzow, Miami Beach, FL;

** CONTINUING DATA ********** ppc yes

This application is a CIP of 10/008,209 12/06/2001
and is a CIP of 10/008,218 12/06/2001
which is a CIP of 09/916,397 07/27/2001
which claims benefit of 60/260,398 01/09/2001 *
and claims benefit of 60/287,813 05/02/2001
and claims benefit of 60/267,944 02/12/2001
and claims benefit of 60/247,242 11/13/2000
and claims benefit of 60/247,232 11/13/2000
(*)Data provided by applicant is not consistent with PTO records.

** FOREIGN APPLICATIONS ********** ppc no

IF REQUIRED, FOREIGN FILING LICENSE GRANTED ** SMALL ENTITY **
** 07/03/2002

Foreign Priority claimed ☐ yes ☑ no 35 USC 119 (a-d) conditions met ☑ yes ☐ no ☐ Met after Allowance Verified and Acknowledged Examiner's Signature ppc Initials	STATE OR COUNTRY FL	SHEETS DRAWING 17	TOTAL CLAIMS 27	INDEPENDENT CLAIMS 4	

ADDRESS
004897
ROBERT C. KAIN, JR.
750 SOUTHEAST THIRD AVENUE
SUITE 100
FT LAUDERDALE , FL
333161153

TITLE
Data security system and method with parsing and dispersion techniques

ppc

Index of Claims

Application/Control No.	Applicant(s)/Patent under Reexamination
10/155,192	REDLICH ET AL.
Examiner	**Art Unit**
Ji-Yong D. Chung	2143

√	Rejected	–	(Through numeral) Cancelled	N	Non-Elected	A	Appeal
=	Allowed	÷	Restricted	I	Interference	O	Objected

Claim		Date
Final	Original	07/22/2005
	1	
	2	
	3	
	4	
	5	
	6	
	7	
	8	
	9	
	10	
	11	
	12	
	13	
	14	
	15	
	16	
	17	
	18	
	19	
	20	
	21	
	22	
	23	
	24	
	25	
	26	
	27	
	28	
	29	
	30	
	31	
	32	
	33	
	34	
	35	
	36	
	37	
	38	
	39	
	40	
	41	
	42	
	43	
	44	
	45	
	46	
	47	
	48	
	49	
	50	

Claim		Date
Final	Original	
	51	
	52	
	53	
	54	
	55	
	56	
	57	
	58	
	59	
	60	
	61	
	62	
	63	
	64	
	65	
	66	
	67	
	68	
	69	
	70	
	71	
	72	
	73	
	74	
	75	
	76	
	77	
	78	
	79	
	80	
	81	
	82	
	83	
	84	
	85	
	86	
	87	
	88	
	89	
	90	
	91	
	92	
	93	
	94	
	95	
	96	
	97	
	98	
	99	
	100	

Claim		Date
Final	Original	
	101	
	102	
	103	
	104	
	105	
	106	
	107	
	108	
	109	
	110	
	111	
	112	
	113	
	114	
	115	
	116	
	117	
	118	
	119	
	120	
	121	
	122	
	123	
	124	
	125	
	126	
	127	
	128	
	129	
	130	
	131	
	132	
	133	
	134	
	135	
	136	
	137	
	138	
	139	
	140	
	141	
	142	
	143	
	144	
	145	
	146	
	147	
	148	
	149	
	150	

Search Notes

Application/Control No.	Applicant(s)/Patent under Reexamination
10/155,192	REDLICH ET AL.
Examiner	**Art Unit**
Ji-Yong D. Chung	2143

SEARCHED			
Class	Subclass	Date	Examiner
715	511	7/28/2005	JDC
705	14	7/28/2005	JDC
709	219,218	7/28/2005	JDC
709	244	7/28/2005	JDC
712	201	7/28/2005	JDC
380	59	7/28/2005	JDC
704	1-10	7/28/2005	JDC

INTERFERENCE SEARCHED			
Class	Subclass	Date	Examiner

SEARCH NOTES (INCLUDING SEARCH STRATEGY)		
	DATE	EXMR
EAST (See attached Search History)	7/26/2005	JDC

JDC

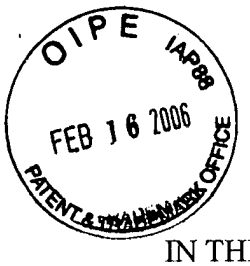
Search Notes	Application/Control No.	Applicant(s)/Patent under Reexamination	
	10/155,192	REDLICH ET AL.	
	Examiner	Art Unit	
	Ji-Yong D. Chung	2143	

SEARCHED			
Class	Subclass	Date	Examiner
715	511	7/28/2005	JDC
705	14	7/28/2005	JDC
709	219,218	7/28/2005	JDC
709	244	7/28/2005	JDC
712	201	7/28/2005	JDC
380	59	7/28/2005	JDC
704	1-10	7/28/2005	JDC

INTERFERENCE SEARCHED			
Class	Subclass	Date	Examiner

SEARCH NOTES (INCLUDING SEARCH STRATEGY)		
	DATE	EXMR
EAST (See attached Search History)	7/26/2005	JDC

JDC



PATENTS

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re Application of Ron M. Redlich, et al.

Serial No. 10/155,192 Examiner Chung

Filed: May 23, 2002 Group 2143

For: Data Security System and Method With Parsing and Dispersion Techniques

AMENDMENT

Mail Stop Amendment
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Sir:

This amendment responds to the Office Action dated August 11, 2005 and the interview conducted with examiner Michael J. Pyzocha, Group 2137, on December 8, 2005 relative to the parent case Serial No. 09/916,397, filed July 27, 2001. A three (3) month extension of time is filed concurrently herewith.

Please charge Deposit Account no. 03-1231 for the $510.00 fee.

[Page left blank]

CLAIMS

1. (currently amended) A method of securing data having one or more security sensitive words, characters or icons in a computer system with a plurality of memories respectively designated as a remainder store and ~~an extract store~~ a plurality of extract stores corresponding to a plurality of security levels for respective subsets of said security sensitive words, characters or icons (FIG. 6, store 1, 2 or 3; FIG. 7, step 422) comprising:

extracting said security sensitive words, characters or icons from said data to obtain extracted data and remainder data therefrom;

storing said extracted data and said remainder data in said ~~extract store~~ respective extract stores corresponding to an associated security level and said remainder store, respectively; and,

permitting reconstruction of said data via said extracted data from respective extract stores and remainder data only in the presence of a predetermined security clearance corresponding to said associated security level for said respective extract stores.

2. (original) A method as claimed in claim 1 including filtering said data while extracting.

3. (currently amended) A method as claimed in claim 2 including defining ~~the filter with a group~~ a plurality of filters for said respective subsets of security sensitive words, characters or icons and the method includes one of storing ~~said filter~~ or destroying ~~said filter~~ one or more of said plurality of filters after storing said extracted data.

4. (original) A method as claimed in claim 3 including encrypting one or both of said extracted data and said remainder data.

5. (original) A method as claimed in claim 4 wherein the step of permitting reconstruction includes decrypting one or both of said extracted data and remainder data.

6. (currently amended) A method as claimed in claim 1 including establishing a plurality of security levels each with a respective security clearance, each said subset of said security sensitive words, characters or icons being correlated with said plurality of security levels ~~with subsets of said security sensitive words, characters or icons~~ and the step of permitting reconstruction including the step of permitting a plurality of partial reconstructions of said data in the presence of respective ones of said plurality of security clearance levels.

7. (currently amended) A method of securing data with a parsing algorithm via a computer system having memories designated as a remainder store and ~~an extract store~~ a plurality of extract stores corresponding to a plurality of security levels for respective subsets of said security sensitive words, characters or icons comprising:

parsing said data to obtain extracted data for respective associated security levels and remainder data therefrom;

storing said extracted data and said remainder data in said respective extract ~~store~~ stores and said remainder store, respectively; and,

permitting reconstruction of said data via said extracted data from said respective extract stores and remainder data only in the presence of a predetermined security clearance corresponding to said associated security level for said respective extract stores.

8. (original) A method as claimed in claim 7 wherein the step of parsing parses the data granularly.

9. (original) A method as claimed in claim 7 wherein the step of parsing uses said parsing algorithm which is based on one or more of a bit count, a word, a word count, a page count, a line count, a paragraph count, an identifiable document characteristic, an identifiable word characteristic, an identifiable icon characteristic, an identifiable data object characteristic, capital letters, italics, and underline.

10. (original) A method as claimed in claim 7 wherein said data is financial data and the step of parsing uses said parsing algorithm based upon one of characters in said data and character count in said data.

11. (currently amended) A method as claimed in claim 7 including establishing a plurality of security levels each with a respective security clearance, permitting increasingly greater reconstruction if said data being correlated with increasingly higher levels of said plurality of security levels includes greater secuiryt level clearances, and the step of permitting reconstruction including the step of permitting a plurality of partial reconstructions of said data in the presence of respective ones of said plurality of security clearance levels.

12. (original) A method as claimed in claim 7 including a further data security step of one from the group of storing said data without encryption, storing said data with encryption and destroying said data; said further data security step being independent of said parsing, storing and permitting reconstruction.

13. (original) A method as claimed in claim 1 including displaying an advertisement prior to one or the other or both of said steps of extracting and permitting reconstruction.

14. (original) A method as claimed in claim 13 including providing a hyperlink to a vendor's web site with said advertisement.

15. (original) A method as claimed in claim 1 including assessing a charge for extracting, storing or permitting reconstruction.

16. (currently amended) A computer readable medium containing programming instructions for securing data having one or more security sensitive words, characters or icons in a computer system with a plurality memories respectively designated as a remainder store and ~~an extract store~~ a plurality of extract stores corresponding to a plurality of security levels for respective subsets of said security sensitive words, characters or icons comprising:

extracting said security sensitive words, characters or icons from said data to obtain extracted data and remainder data therefrom;

storing said extracted data and said remainder data in said ~~extract store~~ respective extract stores corresponding to an associated security level and said remainder store, respectively; and,

permitting reconstruction of said data via said extracted data from respective extract stores and remainder data only in the presence of a predetermined security clearance corresponding to said associated security level for said respective extract stores.

17. (original) A medium with programming instructions as claimed in claim 16 including filtering said data while extracting.

18. (currently amended) A medium with programming instructions as claimed in claim 16 including defining ~~the filter with a group~~ a plurality of filters for respective subsets of security sensitive words, characters or icons and the method includes one of storing ~~said filter~~ or destroying ~~said filter~~ one or more of said plurality of filters after storing said extracted data.

19. (original) A medium with programming instructions as claimed in claim 18 including encrypting one or both of said extracted data and said remainder data.

20. (original) A medium with programming instructions as claimed in claim 19 wherein the step of permitting reconstruction includes decrypting one or both of said extracted data and remainder data.

21. (currently amended) A medium with programming instructions as claimed in claim 16 including establishing a plurality of security levels each with a respective security clearance, each said subset of said security sensitive words, characters or icons being correlated with said plurality of security levels ~~with subsets of said security sensitive words, characters or icons~~ and the step of permitting reconstruction including the step of permitting a plurality of partial reconstructions of said data in the presence of respective ones of said plurality of security clearance levels.

22. (currently amended) A computer readable medium containing programming instructions for securing data with a parsing algorithm via a computer system having memories designated as a remainder store and ~~an extract store~~ a plurality of extract stores corresponding to a plurality of security levels for respective subsets of said security sensitive words, characters or icons comprising:

parsing said data to obtain extracted data for respective associated security levels and remainder data therefrom;

storing said extracted data and said remainder data in said respective extract ~~store~~ stores and said remainder store, respectively; and,

permitting reconstruction of said data via said extracted data from said respective extract stores and remainder data only in the presence of a predetermined security clearance corresponding to said associated security level for said respective extract stores.

23. (original) A medium with programming instructions as claimed in claim 22 wherein the step of parsing parses the data granularly.

24. (original) A medium with programming instructions as claimed in claim 22 wherein the step of parsing uses said parsing algorithm which is based on one or more of a bit count, a word, a word count, a page count, a line count, a paragraph count, an identifiable document characteristic, an identifiable word characteristic, an identifiable icon characteristic, an identifiable data object characteristic, capital letters, italics, and underline.

25. (original) A medium with programming instructions as claimed in claim 22 wherein said data is financial data and the step of parsing uses said parsing algorithm based upon one of characters in said data and character count in said data.

26. (currently amended) A medium with programming instructions as claimed in claim 22 including establishing a plurality of security levels each with a respective security clearance, permitting increasingly greater reconstruction if said data being correlated with increasingly higher levels of said plurality of security levels includes greater secuiryt level clearances, and the step of permitting reconstruction including the step of permitting a plurality of partial reconstructions of said data in the presence of respective ones of said plurality of security clearance levels.

27. (original) A medium as claimed in claim 22 including a further data security step of one from the group of storing said data without encryption, storing said data with encryption and destroying said data; said further data security step being independent of said parsing, storing and permitting reconstruction.

REMARKS

This amendment responds to the Office Action dated August 11, 2005 and the interview conducted with examiner Michael J. Pyzocha, Group 2137, on December 8, 2005 relative to the parent case Serial No. 09/916,397, filed July 27, 2001. A three (3) month extension of time is filed concurrently herewith. Please charge Deposit Account no. 03-1231 for the $510.00 fee.

In the Office Action dated August 11, 2005, the examiner, on pages 2-9, paragraphs 1-5, rejected all claims 1-27 as being non-patentable in view of certain prior or pre-existing technology or art disclosed in the following references:

U.S. Patent No. 5,960,080 to Fahlman (PRIMARY REFERENCE)

U.S. Patent No. 6,602,298 to Kirshenbaum

U.S. Patent No. 6,487,538 to Gupta

U.S. Patent No. 6,771,290 to Hoyle

Published U.S. Application 2002/0116641 to Mastrianni

It is respectfully submitted that Mastrianni '641 is not prior art in that "defining a filter with a group of security sensitive words" and "storing the filter (or destroying the filter)" was disclosed in the parent case, Serial No. 09/916,397, filed July 27, 2001, which parent case was filed before the publication date (August 22, 2002) of Mastrianni '641.

In the parent case, Serial No. 09/916,397, filed July 27, 2001, examiner Pyzocha rejected the claims in light of Kirshenbaum '298 (U.S. Patent No. 6,602,298), Lamm '907 (U.S. Patent No. 6,078,907) and/or Fahlman '080 (U.S. Patent No. 5,960,080). Kirshenbaum '298 and Fahlman '080 were cited in a rejection in the parent case. At the interview on December 8, 2005, the examiner indicated

that a proposed claim (to "multi-level security") was patentable and distinct over the cited references but also indicated that an additional search was necessary.

In this case, independent claims 1, 7, 16 and 22 have been amended toto refer to multiple levels of security. Claim 1 is an example which refers to:

> A method of securing data having one or more security sensitive words... with a plurality of memories respectively designated as a remainder store and ~~an extract store~~ a plurality of extract stores corresponding to a plurality of security levels for respective subsets of said security sensitive words, characters or icons [for support, please see claim 7 and FIG. 6, store 1, 2 or 3; or FIG. 7, step 422] comprising:
>
> extracting ...
>
> storing said extracted data and said remainder data in said ~~extract store~~ respective extract stores corresponding to an associated security level and said remainder store, respectively; and,
>
> permitting reconstruction of said data via said extracted data from respective extract stores and remainder data only in the presence of a predetermined security clearance corresponding to said associated security level for said respective extract stores.

Claims 3, 6, 11, 18, 21 and 26 have been amended to conform to the changes to the independent claims. In general, the foregoing amendments are referred to herein as "multi-level security" features. It is respectfully submitted that the amended claims 1, 7, 16 and 22 all include the concept that various levels of secret data are stored in "respective extract stores" and this is essentially the same as multiple levels of security with multiple levels of secure storage. This concept conforms to the interview on December 8 with Examiner Pyzocha.

In summary, Kirshenbaum '298, Lamm '907 and/or Fahlman '080 do not show, teach or suggest multiple extractions of security sensitive words related to each of a plurality of security levels, and separate storage of those security sensitive words in different secured locations for each security level (multiple

extract stores), and the requirement that the user input a password ("security clearance") for each security level to reconstruct the data from "respective extract stores.".

The advantages of the multiple security level - multiple dispersal and storage claim are:

- distributed, secure storage of multiple levels of secure data

- less risk of loss of secure data by attacks (intrusions) due to the dispersed storage of data

- system provides more physical barriers to electronic intrusions into storage sites

- avoids single point of machine-security clearance failure (possible with prior art systems)

- enhances the greater sharing of data by granular separation and dispersed storage of data at multiple levels of security

Kirshenbaum '298 does not show separate storage of secured data, separate and apart from unsecured data. Both secured and non-secured data is stored in a single database 14. Col. 3, lines 40-44; col. 5, lines 7-10 ("The data set is stored in a database ... the document comprises secure portion and non-secure portions"); col. 5, lines 36-37 ("to retrieve those secure and non-secure portions of the document and to send the retrieved portions of the document to the output device.").

Kirshenbaum '298 does not have storing said remainder [non-secret] data in said remainder store. Kirshenbaum '298 teaches away from this claimed aspect of the invention.

Kirshenbaum '298 does not show extraction of multiple levels of secured data from a document. At best, Kirshenbaum '298 discloses scanning and identification of identifying codes but the identifier codes are not secured data. "In a particular embodiments, the processor is configured to identify a security clearance level of a user and then to enable the user to print portions of a document at any security level below the user's level." Col. 5, line 54. "The identifier codes printed on non-secure portions of a

document are machine readable ... and are provided to a processor by extracting the identifier codes from the machine readable format with a scanning machine." Col. 5, line 64 - Col. 6, line 1.

The present invention requires multiple security levels, each having a sub-set of security sensitive words, extraction and storage for each of the multiple security levels at respective extract stores, a plurality of predetermined security clearances, a particular security clearance needed to reconstruct data from respective ones of said extract stores.

In the present invention, different security clearances must be input to reconstruct data from each extract store (each security level storage facility) in order to obtain the data and permit full or partial reconstruction only in the presence of respective security clearances. Therefore, the present invention requires multiple extraction, each extraction for each security level, separate storage thereof, storage of the remainder or non-secured data, and, permitting full or partial reconstruction of said data via said extracted data and remainder data only with predetermined security clearances.

Fahlman '080 does not show, teach or suggest a remainder store for non-secured data, multiple security levels, multiple extraction of security data, storage of multiple levels, presentment of corresponding security codes at each security level. In fact, nowhere does Fahlman '080 discuss password or security clearance control.

In Fig. 1, step 111 of Fahlman "automatically" merges sensitive information into non-sensitive information. In FIG. 2, step 217, the Fahlman system again "automatically" merges secured data with un-secured data. Fahlman discusses merging secured data with unsecured data at col. 2, line 22, col. 2, line 34, and col. 2, line 49, and col. 2, line 54 and col. 2, line 40. Fahlman '080 discloses identifying security information and extracting that information and replacing it with place holders. Col. 2, line 37 and col. 3,

line 47. "The sanitized message is then transmitted with a low level of security." Col. 3, line 54. The sensitive information is stripped from the original message and stored in a separate location or together with the sanitized message. Col. 3, line 64. A map showing the location of the security information is also stored with the secured information. Col. 3, line 63. "Then, in step 111, the sensitive terms received from the second path are merged with the sanitized message to create a final confidential message." Col. 4, line 1. The examiner should note that Fahlman does not discuss reconstruction or merger in the presence of any type of security clearance. In contrast, the present invention requires multiple security clearances, each unique to a security level. Fahlman also discusses: "Then, in step 217, the sensitive terms are automatically merged back into the serviced message to create a final message." Col. 5, line 27. No discussion of multi-level security clearance is noted. The same is true regarding merger of security information and non-secured information at col. 6, line 62.

Fahlman does not seem to store remainder or non-secure data in a separate remainder store location apart from secure data. Fahlman (1) identifies secret words, (2) replaces the words with placeholders, then (3) transmits the "sanitized message." Fig. 1, step 107, Fig. 2, step 209, col 2, line 19, col. 2, line 34, col. 2, line 49, col. 3, line 54 (transmission - no storage of non-secure data), col. 4, line 64 (transmission - no separate storage). Fahlman does discuss storing the sanitized message, that is, the non-secure data and the placeholders, but not separate storage of the remainder data. Col. 3, line 63, col. 4, line 47, col. 5, line 1. In the present invention, remainder, non-secret data is stored separately from secret data. See claim 63, "storing" step. Fahlman teaches away from this aspect of the present invention.

Fahlman does not discuss multiple security levels nor multiple storage sites at each security level. Since multiple security levels are not addressed in Fahlman, multiple security clearance codes to obtain

access and reconstruct the secured data levels is not discussed. Since Fahlman does not use any type of password or security clearance procedures, there is no need for a plurality of predetermined security clearances to reconstruct said data from multiple extract stores only in the presence of said predetermined security clearances.

There is no motivation nor suggestion to combine Kirshenbaum's presentment of a security code with Fahlman's single level extraction and storage of secret data. There is no suggestion nor motivation to use security codes to reconstruct "full or partial" data from respective extract stores in the presence of such security codes.

Lamm '907 stores and has multiple copies of all secret-secured data about the consumer in three (3) different computers, to wit, consumer computer 12 (see legends FIG. 2, consumer computer 20, col. 5, line 48), billing - processor computer 26 (see col. 13, line 5) and enrollment server 21 (see col. 9, line 42). The three computers in Lamm '907 provide an integrated bill payment system which cannot be deconstructed into operable components. In contrast, the present invention extracts secured data, for multiple security levels, and then stores the extracted data in extract stores. Lamm's process of storing secret data in three computers is completely different than the claimed system of storing secret data in multiple, extract stores for respective security levels.

Because Lamm '907 stores secret and non-secret data in multiple locations, Lamm teaches away from the basic concept of the entire invention, that is, secure, dispersed, distributed storage of data, which can only be "reconstructed" under several password controls.

First, Lamm '907 does not have a single remainder store for non-secure data. Lamm does not have separate storage of secret data, separate and apart from non-secret data.

In Lamm '907, the non-secret or non-sensitive data is transmitted via Internet 28 and is merged with secret data by all three computers completely independent of each other, operationally or otherwise. In other words, a 1st user on consumer computer 12 can reconstruct the data with secret and non-secret data without a security code (col. 12, line 10), [1] completely separate and apart from 2nd user on billing-processor computer 26 (col. 9, line 62). [2] The claimed invention uses multiple secret-extract stores which store different data. To reconstruct a singular document, 1st and 2nd users must access the same extract store at the same security level.

In Lamm '907, there is no corresponding security clearances to get data from respective ones of said extract stores because the secret data is stored locally at all locations. Lamm does not require nor need to present a security code prior reconstruction with multiple extract stores. Since the secret data in Lamm is on all three computers, there is no need to present a security code clearance, nor is there a need to present multiple security clearances to use data form respective extract stores. Lamm does not reconstruct data with predetermined security clearances because each computer locally stores secret data and non-secret data thereon.

The prime purpose of Lamm '907 is as follows: "A primary consequence of this distinction is that billing messages 18 are prepared and sent by the processor server computer system 26 to the electronic

---

[1] "This information from the secured billing information database 38 can appear on the reconstructed bill 154 even though it was not sent with the non-sensitive billing information to the consumer's computer 12, because it is stored locally on the consumer's computer 12. An encryption program 36 may be used on the consumer's computer 12 if data sent from the processing computer system 20 is encrypted." Col. 12, lines 36-43 (emphasis added).

[2] "As noted earlier, an authentication identifier, such as an EPO-mail address supplemented by a password [sent by the consumer], is set up to allow the processor or billing party to receive payment instructions from a consumer without the transfer of secured billing information." (col. 9, line 62)

post office 16 with redacted content only. Similarly, payment instruction messages 19 are prepared by consumer computer 12 and sent to the electronic post office 16 with redacted content only." Col. 6, lines 25-32. Lamm is not concerned with (a) extraction, (b) secure storage of extracted data at multiple levels, (c) full or partial reconstruction with multiple extract stores and predetermined security clearances. Lamm does not require a high level of security because the security objective of Lamm '907 is limited to protection of the sensitive information while the information travels the internet or phone wires. Lamm does not consider the storage at the three computers unsafe because it does not even require a password to access that secret information.

A combination of Kirshenbaum '298, Lamm '907 and/or Fahlman '080 would result in a three headed monster with no discernable utility. No reference specifically discusses separate storage of non-secret, remainder data, apart from secret data. No reference shows, discusses or suggests multiple security levels and multiple stores for each level of security. Lamm stores secret data and non-secret data locally on many computers. Kirshenbaum stores all data, secret and non-secret, in one location. Fahlman extracts only a single level of data and stores it separately from the non-secure data (which is transmitted, not stored "in a remainder store" as per the claimed invention) but never discusses password control to reconstruct with multiple extract stores. It is respectfully submitted that it is improper to select disparate parts from each of these complicated systems and "cobble together" the claimed invention. There is no suggestion nor motivation to do so absent the disclosure of the present invention.

Each of the references, Kirshenbaum '298, Lamm '907 and/or Fahlman '080, describe complete systems and there is no reason to add to or substitute portions of one disclosure with another. For

example, there is no reason to combine Lamm's bill paying system with Kirshenbaum's singular document storage and reproduction system and/or Fahlman's secret transmission system.

Gleichauf '668 (U.S. Patent No. 6,301,668, cited in Serial No. 10/008,218) discloses an adaptive network security system and FIG. 4 therein shows the use of a system which determines, in step 112, the type of attack (see col. 8, line 28, especially line 36) and then applies the associated response. Priority is assigned based upon the type of attack. "System services are prioritized based upon a level of criticality of each services as can be determined from the network information." Col. 8, line 49.

Wylie's book on Survival Information (cited in Serial No. 10/008,218) describes a decentralized storage system that "divide[s] the information into multiple pieces, or shares, that can be stored at different storage nodes." p. 62, left col. However, the Wylie system includes a data redundancy feature. p. 62, right col. "As the 'General Threshold Schemes' sidebar describes, a p-m-n threshold scheme breaks information into n shares so that any m of the shares can reconstruct the information and fewer than p shares reveal no information" p. 63, right col. In direct contrast to the presently claimed invention, the Wylie system would not work in that Wylie would store multiple versions or copies of the secret information over multiple storage sites to achieve the p-m-n data redundancy feature. Wylie always employs the p-m-n redundancy feature in his distributed storage system. Storing multiple copies of the same secret data is completely different than the claimed storing a extracted data in respective secret storage stores per the present invention. For example, which security code would one use with the multiple M copies of secured data in the Wylie system? Therefore, the Wylie system is closer to the system in Lamm '907 wherein multiple copies of the same secret information are stored throughout the system and the user can access the same secret information at multiple sites. In the present invention, the secret information is stored one-

time in multiple, secure extract stores which stores are complimentary to the security level. Wylie teaches away from the present invention as does Lamm '907 since multiple copies of the same secret data are stored at multiple locations.

With respect to Schneier's book (Applied Cryptography)(cited in the parent case, Serial No. 09/916,397), Schneier does not show, teach or suggest multi-level extract stores nor predetermined security clearances, nor reconstruction of said data via multiple extract stores only in the presence of said predetermined security clearances. Schneier discusses encryption and key destruction.

Kluttz '161 (U.S. Patent No. 6,598,161, cited in the parent case)does not show, teach or suggest multi-level extract stores nor permitting reconstruction of said data via said extract data stores only in the presence of predetermined security clearances. Kluttz '161 shows utilizing multiple encryption portions in a singular document. See Abstract and FIG. 3. The keys are maintained in the document 100. Col. 6, lines 28-30. FIGS. 5 and 6 show the flowcharts for document decryption which includes utilizing the encryption key in the document itself (step 304, FIG. 5; step 404, FIG. 6). There is no suggestion of utilizing an extracted store and a remainder store.

Kluttz '161 does not show different levels of security for subsets of information. It discloses "dividing the document into at least a first portion having a first security level and a second portion having a second security level" and then encrypting these two levels differently. There is no different storage of different secure information with different password keys for each level, AND separate storage of remainder, non-secure data.

U.S. Patent No. 5,036,315 to Gurley (cited in the parent case) does not cure the defects identified above with respect to Lamm '907 and the differences with respect to the present invention. Gurley does

not show, teach or suggest (a) filtering data; (b) utilizing multiple extract stores and a remainder store; (c) multiple security clearances for the extract stores to permit reconstruction of said data via the multiple extract stores only in the presence of predetermined security clearances. Gurley '315 discusses a video display control which accepts and processes two (2) video signals, one displayed in a defined window of the second video display.

With respect to Gupta '538 and Hoyle '290, those references are cited by the examiner as secondary references. The secondary references do not cure the defects described above with respect to the primary references.

Applicant respectfully requests that the examiner withdraw the rejections and approve the patentability of claims 1 - 27.

Respectfully submitted,

By _____

Robert C. Kain, Jr.
Reg. No. 30,648
Fleit, Kain, Gibbons, Gutman, Bongini & Bianco, P.L.
750 Southeast Third Avenue, Suite 100
Fort Lauderdale, FL 33316-1153
Telephone: 954-768-9002
Facsimile: 954-768-0158

Certificate of Mailing

I hereby certify that this correspondence is being deposited with the U.S. Postal Service as First Class Mail in an envelope addressed to Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450 on February 13, 2006.

_____

Robert C. Kain, Jr.
Reg. No. 30,648

\\Tiger\data share\RCK\CLIENTS\Redlich\Patents\6851-02cip3-amdt-021006.wpd

OIPE IAP88
FEB 16 2006
PATENT & TRADEMARK OFFICE

PATENTS

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re Application of Ron M. Redlich, et al.

Serial No. 10/155,192

Filed: May 23, 2002

For: Data Security System and Method with Parsing and Dispersion Techniques

INFORMATION DISCLOSURE STATEMENT

Mail Stop IDS
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Sir:

As a means of complying with the duty of disclosure set forth in 37 C.F.R. §1.56, applicant, through the undersigned attorney, files this Information Disclosure Statement pursuant to 37 C.F.R. §1.97.

The examiner in Serial No. 09/916,397 cited the following references:

U.S. Patent No. 6,078,907 to Lamm
U.S. Patent No. 6,598,161 to Kluttz
U.S. Patent No. 5,036,315 to Gurley
The 1996 book, Applied Cryptography, by Schneier (attached to Office Action)
The Uniform Resource Locator article "FOLD OC" (attached to Office Action)

The examiner in Serial No. 10/008,209 cited the following references:

U.S. Patent No. 6,078907 to Lamm (PRIMARY REFERENCE)
U.S. Patent No. 6,598,161 to Kluttz
U.S. Patent No. 5,036,315 to Gurley
U.S. Patent No. 5,532,950 to Moses
The 1996 book, Applied Cryptography, by Schneier (attached to Office Action)
The Uniform Resource Locator article "FOLD OC" (attached to Office Action)

The examiner in Serial No. 10/008,218 cited the following references:

U.S. Patent No. 6,078,907 to Lamm
U.S. Patent No. 6,301,668 to Gleichauf
U.S. Patent No. 6,598,161 to Kluttz
U.S. Patent No. 5,036,315 to Gurley
U.S. Patent No. 5,532,950 to Moses
U.S. Patent No. 6,714,977 to Fowler
The 1996 book, Applied Cryptography, by Schneier (attached to Office Action)
The Uniform Resource Locator article "FOLD OC" (attached to Office Action)
Wylie "Survivable Information Storage Systems" (cited by Redlich)

The examiner in Serial No. 10/155,192 cited the following references:

U.S. Patent No. 5,960,080 to Fahlman (PRIMARY REFERENCE)
U.S. Patent No. 6,602,298 to Kirshenbaum
U.S. Patent No. 6,487,538 to Gupta
U.S. Patent No. 6,771,290 to Hoyle
Published U.S. Application 2002/0116641 to Mastrianni

The examiner in Serial No. 10/155,525 cited the following references:

U.S. Patent No. 5,960,080 to Fahlman (PRIMARY REFERENCE)
U.S. Patent No. 6,602,298 to Kirshenbaum
U.S. Published Patent Application 2002/0116641 to Mastrianni
U.S. Published Patent Application 2004/0054630 to Ginter
Microsoft Word "Learning Microsoft Word 7.0"

This Information Disclosure Statement is filed in accordance with 37 C.F.R. §1.97, and this statement shall not be construed as a representation that a search was made, has been made, or that the information cited above is, or is considered to be, material to patentability as defined in §1.56(b), or that no other material information, as defined in 37 C.F.R. §1.56(b), exists. Translations of the pertinent portions of foreign language patents or publications listed above are supplied if existing translations are readily available to the applicant or the applicant's attorney. A concise explanation of the relevance, as it

is presently understood by the individual designated in 37 CFR §1.56(c) most knowledgeable about the content of the information, of each patent, publication, or other information listed that is not in the English language is provided either in the patent specification or in the enclosed attachment.

Respectfully submitted,

By_____

Robert C. Kain, Jr.
Reg. No. 30,648
Fleit, Kain, Gibbons, Gutman, Bongini & Bianco, P.L.
750 Southeast Third Avenue
Suite 100
Ft. Lauderdale, Florida 33316-1153
Telephone: 954-768-9002
Facsimile: 954-768-0158

Certificate of Mailing

I hereby certify that this correspondence is being deposited with the U.S. Postal Service as First Class Mail in an envelope addressed to Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450 on February 13, 2006.

_____

Robert C. Kain, Jr.
Reg. No. 30,648

\\Tiger\data share\RCK\CLIENTS\Redlich\6851-02-cip3 IDS.wpd

OIPE IAP88 FEB 16 2006 PATENT & TRADEMARK OFFICE

INFORMATION DISCLOSURE CITATION *(Use several sheets if necessary)*	Docket Number (Optional) 6851-02-CIP3	Application Number 10/155,192
Title: Data Security System and Method with Parsing and Dispersion Techniques	Applicant(s) Ron M. Redlich, et al.	
	Filing Date May 23, 2002	Group Art Unit 2143

U.S. PATENT DOCUMENTS

Examiner Initial	Document Number	Date	Name	Class	Subclass	Filing Date if Appropriate
	6,078,907	6/20/2000	Lamm	705	40	
	6,598,161	7/22/2003	Kluttz	713	166	
	5,036,315	7/30/1991	Gurley	340	721	
	5,532,950	7/02/1996	Moses	364	724.19	
	6,301,668	10/09/2001	Gleichauf	713	201	
	6,714,977	3/30/2004	Fowler	709	224	

FOREIGN PATENT DOCUMENTS

	Document Number	Date	Country	Class	Subclass	Translation YES NO

OTHER DOCUMENTS (Including Author, Title, Date, Pertinent Pages, Etc.)

	The 1996 book, Applied Cryptography, by Schneier	
	The Uniform Resource Locator article "FOLD OC"	
	Wylie "Survival Information Storage Systems"	
	Microsoft Word "Learning Microsoft Word 7.0"	

EXAMINER DATE CONSIDERED

OIPE
FEB 16 2006
PATENT & TRADEMARK OFFICE
IAP88

INFORMATION DISCLOSURE CITATION *(Use several sheets if necessary)*	Docket Number (Optional) 6851-02-CIP3	Application Number 10/155,192
Title: Data Security System and Method with Parsing and Dispersion Techniques	Applicant(s) Ron M. Redlich, et al.	
	Filing Date May 23, 2002	Group Art Unit 2143

U.S. PATENT DOCUMENTS

Examiner Initial	Document Number	Date	Name	Class	Subclass	Filing Date if Appropriate
	5,960,080	9/28/1999	Fahlman	380	4	
	6,602,298	8/05/2003	Kirshenbaum	715	502.2	
	6,487,538	11/26/2002	Gupta	705	24	
	6,771,290	8/03/2004	Hoyle	345	745	
	US2002/0116641	8/22/2002	Mastrianni	713	201	
	US2004/0054630	3/18/2004	Ginter	705	53	

FOREIGN PATENT DOCUMENTS

	Document Number	Date	Country	Class	Subclass	Translation YES NO

OTHER DOCUMENTS (Including Author, Title, Date, Pertinent Pages, Etc.)

EXAMINER DATE CONSIDERED

OIPE IAP88 FEB 16 2006 PATENT & TRADEMARK OFFICE

PATENTS

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re Application of Ron M. Redlich, et al.

Serial No. 10/155,192 Examiner Chung

Filed: May 23, 2002 Group 2143

For: Data Security System and Method with Parsing and Dispersion Techniques

TRANSMITTAL LETTER

Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450
Mail Stop Amendment

Sir:

Please find enclosed for filing:

X Amendment with certificate of mailing (18 pages)

X Request for Extension of Time with certificate of mailing (1 page)

X Information Disclosure Statement with PTO Form 1449 with certificate of mailing (3 refs)

X Check No. 4965 in the amount of $510.00

X Return Receipt Postcard

X Please debit/credit Deposit Account No. 03-1231 for any underpayments/overpayments.

This Transmittal Letter is submitted in duplicate.

Respectfully submitted,

By _____

Robert C. Kain, Jr.
Reg. No. 30,648
Fleit, Kain, Gibbons, Gutman, Bongini & Bianco, P.L.
750 Southeast Third Avenue, Suite 100
Ft. Lauderdale, FL 33316-1153
Telephone: 954-768-9002
Facsimile: 954-768-0158

Docket No.: 6851-02-CIP3

Certificate of Mailing

I hereby certify that this correspondence is being deposited with the U.S. Postal Service as First Class Mail in an envelope addressed to Mail Stop Amendment, Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450 on February 13, 2006.

_____

Robert C. Kain, Jr.
Reg. No. 30,648

TFW 2143

OIPE IAP68
FEB 16 2006
PATENT & TRADEMARK OFFICE

PATENTS

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re Application of Ron M. Redlich, et al.

Serial No. 10/155,192 Examiner Chung

Filed: May 23, 2002 Group 2143

For: Data Security System and Method with Parsing and Dispersion Techniques

REQUEST FOR EXTENSION OF TIME

Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Sir:

This is a request for extension of time pursuant to 37 CFR §1.136. A check in the amount of $510.00 accompanies this request for a three month extension of time pursuant to 37 CFR §1.17(c). Please charge any underpayment or credit any overpayment to Deposit Account No. 03-1231. A duplicate copy of this communication is enclosed.

Respectfully submitted,

By _____

Robert C. Kain, Jr.
Reg. No. 30,648
Fleit, Kain, Gibbons, Gutman, Bongini & Bianco, P.L.
750 Southeast Third Avenue
Suite 100
Ft. Lauderdale, Florida 33316-1153
Telephone: 954-768-9002
Facsimile: 954-768-0158

Certificate of Mailing

I hereby certify that this correspondence is being deposited with the U.S. Postal Service as First Class Mail in an envelope addressed to Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450 on February 13, 2006.

02/16/2006 NNGUYEN1 00000059 10155192
01 FC:2253 510.00 OP

Robert C. Kain, Jr.
Reg. No. 30,648

\\Tiger\data share\RCK\CLIENTS\Redlich\6851-2cip3 TIME3.REQ.wpd

PATENT APPLICATION FEE DETERMINATION RECORD
Effective October 1, 2001

Application or Docket Number
~~6851-2 CIP-3~~ 10/155,192

CLAIMS AS FILED - PART I

	(Column 1)	(Column 2)
TOTAL CLAIMS	27	
FOR	NUMBER FILED	NUMBER EXTRA
TOTAL CHARGEABLE CLAIMS	27 minus 20=	* 7
INDEPENDENT CLAIMS	4 minus 3 =	* 1
MULTIPLE DEPENDENT CLAIM PRESENT	☐	

* If the difference in column 1 is less than zero, enter "0" in column 2

SMALL ENTITY TYPE ☐		OR	OTHER THAN SMALL ENTITY	
RATE	FEE		RATE	FEE
BASIC FEE	370.00	OR	BASIC FEE	740.00
X$ 9=	63	OR	X$18=	
X42=	42	OR	X84=	
+140=		OR	+280=	
TOTAL	475	OR	TOTAL	

CLAIMS AS AMENDED - PART II

AMENDMENT A

	(Column 1) CLAIMS REMAINING AFTER AMENDMENT		(Column 2) HIGHEST NUMBER PREVIOUSLY PAID FOR	(Column 3) PRESENT EXTRA
Total	* 27	Minus	** 27	=
Independent	* 4	Minus	*** 4	=
FIRST PRESENTATION OF MULTIPLE DEPENDENT CLAIM ☐				

SMALL ENTITY		OR	OTHER THAN SMALL ENTITY	
RATE	ADDITIONAL FEE		RATE	ADDITIONAL FEE
X$ 9=		OR	X$18=	
X42=		OR	X84=	
+140=		OR	+280=	
TOTAL ADDIT. FEE		OR	TOTAL ADDIT. FEE	

AMENDMENT B

	(Column 1) CLAIMS REMAINING AFTER AMENDMENT		(Column 2) HIGHEST NUMBER PREVIOUSLY PAID FOR	(Column 3) PRESENT EXTRA
Total	*	Minus	**	=
Independent	*	Minus	***	=
FIRST PRESENTATION OF MULTIPLE DEPENDENT CLAIM ☐				

RATE	ADDITIONAL FEE		RATE	ADDITIONAL FEE
X$ 9=		OR	X$18=	
X42=		OR	X84=	
+140=		OR	+280=	
TOTAL ADDIT. FEE		OR	TOTAL ADDIT. FEE	

Best Available Copy

AMENDMENT C

	(Column 1) CLAIMS REMAINING AFTER AMENDMENT		(Column 2) HIGHEST NUMBER PREVIOUSLY PAID FOR	(Column 3) PRESENT EXTRA
Total	*	Minus	**	=
Independent	*	Minus	***	=
FIRST PRESENTATION OF MULTIPLE DEPENDENT CLAIM ☐				

RATE	ADDITIONAL FEE		RATE	ADDITIONAL FEE
X$ 9=		OR	X$18=	
X42=		OR	X84=	
+140=		OR	+280=	
TOTAL ADDIT. FEE		OR	TOTAL ADDIT. FEE	

* If the entry in column 1 is less than the entry in column 2, write "0" in column 3.
** If the "Highest Number Previously Paid For" IN THIS SPACE is less than 20, enter "20."
*** If the "Highest Number Previously Paid For" IN THIS SPACE is less than 3, enter "3."
The "Highest Number Previously Paid For" (Total or Independent) is the highest number found in the appropriate box in column 1.

EAST Search History

Ref #	Hits	Search Query	DBs	Default Operator	Plurals	Time Stamp
S164	1	"6078907".pn.	US-PGPUB; USPAT	OR	ON	2006/05/15 15:25
S165	115	(extracting adj security sensitive adj word)	US-PGPUB; USPAT	OR	ON	2006/05/15 15:26
S166	10	(extracting adj security adj sensitive adj words)	US-PGPUB; USPAT	OR	ON	2006/05/15 15:43
S167	66	mastrianni.inv.	US-PGPUB; USPAT	OR	ON	2006/05/15 15:43



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.
10/155,192	05/23/2002	Ron M. Redlich	6851-2-CIP-3	9510

4897 7590 06/01/2006

ROBERT C. KAIN, JR.
750 SOUTHEAST THIRD AVENUE
SUITE 100
FT LAUDERDALE, FL 333161153

EXAMINER	
CHUNG, JI YONG DAVID	
ART UNIT	**PAPER NUMBER**
2143	

DATE MAILED: 06/01/2006

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

Office Action Summary	Application No.	Applicant(s)	
	10/155,192	REDLICH ET AL.	
	Examiner	Art Unit	
	Ji-Yong D. Chung	2143	

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

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on 2/16/2006.

2a)☒ This action is **FINAL**. 2b)☐ This action is non-final.

3)☐ Since this application is in condition for allowance 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.

Disposition of Claims

4)☒ Claim(s) 1-27 is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) 1-27 is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☐ The drawing(s) filed on _____ is/are: a)☐ accepted or b)☐ objected to by the Examiner.

- Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☒ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08) Paper No(s)/Mail Date 2/16/2006.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____.
5) ☐ Notice of Informal Patent Application (PTO-152)
6) ☐ Other: _____.

nnc

DETAILED ACTION

Response to Remarks

1. Applicant's arguments and amendments filed on February 16, 2006 have been carefully considered. While the applicant's amendments substantively advance the prosecution, the Office raises a number of issues that the Office raises, in the form of 35 U. S. C. 103 rejections, in the instant Office action.

The applicant is reminded that the rejections are not indicative of unpatentability of the underlying subject matter and that they have been issued to facilitate removal of any ambiguities or any potential issues that may challenge the integrity of valid patents.

The applicant's arguments provided in Remarks are deemed moot in light of the new grounds of rejection (at least for the independent claims), set forth in the remainder of the instant Office action. While the same references have been cited in the rejection of the number of claims, the grounds of rejections have been changed.

The applicant has indicated that Mastrianni (Pub. No. 2002/0116641) cited in the prior action is not a prior art reference. The Office acknowledges that Mastrianni does not qualify as a prior art reference. In relevant sections below, Narasimhan et al. (Pat. No. 6,073,165) takes the place of Mastrianni in setting forth new grounds of rejections, with regard to claims 3-5 and claims 8-20.

Claim Rejections - 35 USC § 103

2. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

3. **Claims 1, 2, 6-12, 16, 17, and 21-27** are rejected under 35 U.S.C. 103(a) as being unpatentable over Fahlman et al (Pat. No. 5,960,080, Fahlman hereinafter) in view Kirshenbaum (Pat No. 6,602,298).

In reference to **claim 1,** Fahlman shows a *method of securing data having one or more security sensitive words, characters or icons in a computer system with memories designated as a remainder store and an extract store* [See lines 27-67, column 3] comprising:

extracting said security sensitive words, characters or icons from said data to obtain extracted data and remainder data therefrom [See lines 61-67, column 3. See also Fig 2.];

storing said extracted data and said remainder data in said extract store and said remainder store, respectively [See lines 61-67, column 3. See Fig. 2. Note that the storage of "remainder" data is implicit. Any computing operation that requires extraction of data requires storage, at least in RAM. Note that messages discussed in Fahlman are "electronic messages" (or email), which requires at least storage of information at email servers] and,

permitting reconstruction of said data via said extracted data and remainder data [See lines 27-32, column 5. The merging is "permitted" at the destination terminal].

Fahlman does not show, but Kirshenbaum shows that "reconstruction" of the message (i.e., seeing the whole document and not merely the portions with lower levels of security) is permitted *only in the presence of a predetermined security clearance* [See lines 34-62, column 5. The user (sender or recipient) privilege on the messages is based on security clearance].

It would have been obvious to one of ordinary skill in the art at the time of the invention to combine Kirshenbaum's feature of letting ("permitting") only users with the proper security clearance level to view of the parts of the original document (i.e, permitting the recipient in Fig. 2 of Fahlman with the proper security clearance to view ('assemble") the document), because requiring the security clearance, as Kirshenbaum's system suggests, increases the security of the document distribution (i.e., prevents unauthorized access).

The amended portions of the claim cite *plurality of extract stores corresponding to a plurality of security levels for respective subsets of said security sensitive words, characters or icons*.

Kirshenbaum illustrates the concept that a document may have portions with different levels of security. Thus, one portion of a given document may require a high security clearance to view; another portion may require less rigorous level of security clearance; and another portion may require minimal security clearance. This concept of having different levels of clearance to view different portions of document is described in lines 56-65 in column 4.

Furthermore, Kirshenbaum shows that a document with different security levels have different subcomponents. Each subcomponent is interpreted as a "store" and thus, Kirshenbaum's embodiment that supports multiple security levels has multiple components of a

document and thus multiple "stores." That is, a database referred to in Kirshenbaum is not one monolithic store, but composed of multiple stores. See lines 33-47 of column 3, for a description referring to "stores" (subcomponents). Kirshenbaum shows that there are multiple stores and the stores maybe on different locations (e.g., the Internet). While Kirshenbaum does not state the following, one can see that each subcomponent that corresponds to a security level is thus stored in a separate "store" based on Kirshenbaum disclosure.

It would have been obvious to include the multi-level security feature in Kirshenbaum, because it increases the granularity of security control. Note that this implies multiple filtering, each filtering corresponding to one security level.

With respect to **claim 2**, Fahlman shows the *method including filtering said data while extracting*. The "extraction" is filtering; it filters out the security sensitive words.

Claim 6 speaks of *a method including establishing a plurality of security levels each with a respective security clearance, each said subset of said security sensitive words, characters or icons being correlated with said plurality of security levels and the step of* permitting reconstruction including the step of permitting a *plurality of partial reconstructions of said data in the presence of respective ones of said plurality of security clearance levels.*

The above discussion of claim 1 has also addressed the substantive portion of the limitations in claim 6 involving multiple security clearance levels. Claim 6 further defines the concept introduced in claim 1, with introduction of "plurality of clearance levels."

However, as explained above, Kirshenbaum provides the conceptual basis.

Conceptually, Kirshenbaum shows "reconstruction" of portions of document based on security clearance level. Some portions require greater clearance than others. See lines 34-62, column 5. The hierarchy of clearance allows those with greater clearance can always access those of lower clearance. If a document has two portions, one of which required greater security clearance than the preceding one, user of at least one clearance level would be required to "reconstruct" the relevant portion.

It would have been obvious to one of ordinary skill in the art at the time of the invention to apply Kirshenbaum to create a multiple portion document with multiple security clearance. The reason for the combination is, as explained in lines 47-59, column 1, Kirshenbaum's concept addresses inconvenience of generally passing the whole document from one person to another. Combining Kirshenbaum with Fahlman allows one to pass portions of document, with proper levels of security. The implementation of multiple filters (each of different security level) flows from the necessarily of partitioning document into multiple security level portions.

With respect to **claim 7**, all of its limitations have been discussed with respect to claims 1 and 6, except that word "parsing" is used. However, Fahlman shows "parsing." See lines 34-47, column 3.

With respect to **claim 8**, Fahlman shows parsing *parses the data granularly*. See lines 34-47 in column 3, which indicates that data is parsed at word level.

With respect to **claim 9**, Fahlman shows that the step of *parsing* uses said parsing algorithm which is based on one or more of a bit count, *a word* [See line 34-47, column 3] a word count, a page count, a line count, a paragraph count, an identifiable document characteristic, *an identifiable word characteristic* [See lines 43-37, column 3], an identifiable icon characteristic, an identifiable data object characteristic, *capital letters* [See line 41-43, column 3], italics, and underline.

With respect to **claim 10**, Fahlman shows the method wherein *said data is financial data and the step of parsing uses said parsing algorithm based upon one of characters in said data and character count in said data.* See lines 27-37, column 4. Financial data is mentioned. See lines 43-46, column 3. It mentions dollar sign (which is "one of characters." Numbers are "character count." See lines 47-52, column 3 for dates, which include "character count." Dates are included as part of transaction date. See lines 27-37, column 4.

With refernce to **claim 11**, Fahlman does not shows *establishing a plurality of security levels each with a respective security clearance, permitting increasingly greater reconstruction if said data being correlated with increasingly higher levels of said plurality of security levels includes greater security level clearances, and the step of permitting reconstruction including the step of permitting a plurality of partial reconstructions of said data in the presence of respective ones of said plurality of security clearance levels.*

Kirshenbaum shows "reconstruction" of document with increasingly higher levels of security levels and the reconstruction requires respective ones of the security clearances. See lines 34-62, column 5.

It would have been obvious to one of ordinary skill in the art at the time of the invention to apply Kirshenbaum to create multiple portion document with multiple security clearance level, with the mechanism for associating each filter with different security clearance level and with the mechanism for repeated filtering, because as explained in lines 47-59, column 1, Kirshenbaum's concept addresses inconvenience of generally passing the whole document from one person to another. The combination of Fahlman and Kirshenbaum can pass portions of a document. The implementation of multiple filters (each of different security level) flows from the necessarily of partitioning document into multiple security level portions.

With respect to **claim 12**, Fahlman shows the method including a further data security step of one from the group of *storing said data without encryption* [See step 213, Fig. 2], storing said data with encryption and destroying said data; *said further data security step being independent of said parsing, storing and permitting reconstruction* [See steps 217 and 219 in Fig. 2. Parsing step is 203. Storing step is 213, reconstruction is decrypting step (shown in Fig. 3) together with steps 217 and 219. The steps are independent in the sense that each step does not depend on the other steps themselves, but only on their outputted data.].

Claims 16, 17, 21-27 incorporate all the corresponding limitations of claims 1, 2, and 6-12, but in computer product (software) form rather than in method form. The reasons for the rejection of claims 1, 2, 6-12 apply to claims 16, 17, and 22-27.

4. **Claims 3-5 and 18-20** are rejected under 35 U.S.C. 103(a) as being unpatentable over Fahlman and Kirshenbaum and in further in view of Narasimhan et al. (Pat. No. 6,073,165).

With respect to **claim 3**, neither Falhlman nor Kirshenbaum shows, but Narasimhan shows the method including

defining the filter with a group of security sensitive words [See lines 3-38 in column 5. It shows selection of search terms], characters or icons and

the method includes one of storing said filter or destroying said filter after storing said extracted data [Predetermining the criteria is equivalent to storing filter (by storing the filtering criteria)].

It would have been obvious to one of ordinary skill in the art at the time of the invention to apply the steps of defining the filter and storing it, because the process would allow multiple use of the filters in different instances. In addition, the mechanism would allow one to improve and modify the filters over time.

With respect to **claim 4**, Fahlman shows *encrypting one or both of said extracted data and said remainder data.* See Fig. 6.

With respect to **claim 5**, Fahlman shows wherein *decrypting one or both of said extracted data and remainder data*. See Fig. 6. Note that "decrypting 309 in Fig. 6 maybe viewed as part of "reconstruction" step.

Claims 18-20 incorporate all the corresponding limitations of claims 3-5, but in computer product (software) form rather than in method form. The reasons for the rejection of claims 3-5 apply to claims 18-20.

5. **Claims 13 and 15** are rejected under 35 U.S.C. 103(a) as being unpatentable over Fahlman and Kirshenbaum and in further in view of Gupta et al. (Pat. No. 6,487,538, Gupta hereinafter)

With reference to **claim 13**, neither Fahlman nor Kirshenbaum shows its limitations. However, Gupta shows *displaying an advertisement prior to one or the other or both of said steps of extracting and permitting reconstruction.* [Lines 10-22, column 6]

It would have been obvious to one of ordinary skill in the art at the time of the invention to use Gupta's system to insert advertisement into Fahlman and Kirshenbaum's combination that would use an application server to render the document delivery service, because Gupta's system allows such service to charge for advertisements and obtain additional revenue.

With reference to **claim 15**, neither Fahlman nor Kirshenbaum shows the limitation or charging for *extracting, storing or permitting reconstruction* (all of which have been shown as explained in reference to claim 1).

Gupta illustrates the general concept of charging for its services. See lines 36-47, column 16.

It would have been obvious to one of ordinary skill in the art at the time of the invention to charge clients, as Gupta shows, for using Fahlman and Kirshenbaum combination services, because charging the clients would allows the service provider to generate revenue.

6. **Claim 14** is rejected under 35 U.S.C. 103(a) as being unpatentable over Fahlman and Kirshenbaum and Gupta in view of Hoyle (Pat. No. 6,771,290).

With reference to **claim 14**, neither Fahlman, Kirshenbaum, nor Gupta shows its limitation.

Hoyle shows a hyperlink to a vendor's web site with said advertisement. See lines 29-41, column 17.

It would have been obvious to one of ordinary skill in the art at the time of the invention to use a banner advertisement with URL as explained in Hoyle, because such banner advertisements are much smaller in size than a full page advertisement, and therefore would provide savings in advertisement cost.

Conclusion

7. Applicant's amendment necessitated the new ground(s) of rejection presented in this Office action. Accordingly, **THIS ACTION IS MADE FINAL.** See MPEP § 706.07(a). Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire THREE MONTHS from the mailing date of this action. In the event a first reply is filed within TWO MONTHS of the mailing date of this final action and the advisory action is not mailed until after the end of the THREE-MONTH shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than SIX MONTHS from the date of this final action.

8. Any inquiry concerning this communication or earlier communications from the examiner should be directed to Ji-Yong D. Chung whose telephone number is (571) 272-7988. The examiner can normally be reached on Monday-Friday 9:30-6:00.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, David Wiley can be reached on (571) 272-3923. The fax phone number for the organization where this application or proceeding is assigned is 703-872-9306.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

Ji-Yong D. Chung
Patent Examiner
Art Unit: 2143

DAVID WILEY
SUPERVISORY PATENT EXAMINER
TECHNOLOGY CENTER 2100

Notice of References Cited	Application/Control No. 10/155,192	Applicant(s)/Patent Under Reexamination REDLICH ET AL.	
	Examiner Ji-Yong D. Chung	Art Unit 2143	Page 1 of 1

U.S. PATENT DOCUMENTS

*		Document Number Country Code-Number-Kind Code	Date MM-YYYY	Name	Classification
*	A	US-6,073,165	06-2000	Narasimhan et al.	709/206
	B	US-			
	C	US-			
	D	US-			
	E	US-			
	F	US-			
	G	US-			
	H	US-			
	I	US-			
	J	US-			
	K	US-			
	L	US-			
	M	US-			

FOREIGN PATENT DOCUMENTS

*		Document Number Country Code-Number-Kind Code	Date MM-YYYY	Country	Name	Classification
	N					
	O					
	P					
	Q					
	R					
	S					
	T					

NON-PATENT DOCUMENTS

*		Include as applicable: Author, Title Date, Publisher, Edition or Volume, Pertinent Pages)
	U	
	V	
	W	
	X	

*A copy of this reference is not being furnished with this Office action. (See MPEP § 707.05(a).)
Dates in MM-YYYY format are publication dates. Classifications may be US or foreign.

JDC

OIPE IAP68
FEB 16 2006
PATENT & TRADEMARK OFFICE

INFORMATION DISCLOSURE CITATION *(Use several sheets if necessary)*	Docket Number (Optional) 6851-02-CIP3	Application Number 10/155,192
Title: Data Security System and Method with Parsing and Dispersion Techniques	Applicant(s) Ron M. Redlich, et al.	
	Filing Date May 23, 2002	Group Art Unit 2143

U.S. PATENT DOCUMENTS

Examiner Initial	Document Number	Date	Name	Class	Subclass	Filing Date if Appropriate
JRC	6,078,907	6/20/2000	Lamm	705	40	
	6,598,161	7/22/2003	Kluttz	713	166	
	5,036,315	7/30/1991	Gurley	340	721	
	5,532,950	7/02/1996	Moses	364	724.19	
	6,301,668	10/09/2001	Gleichauf	713	201	
	6,714,977	3/30/2004	Fowler	709	224	

FOREIGN PATENT DOCUMENTS

	Document Number	Date	Country	Class	Subclass	Translation YES NO

OTHER DOCUMENTS (Including Author, Title, Date, Pertinent Pages, Etc.)

JRC	The 1996 book, Applied Cryptography, by Schneier	
	The Uniform Resource Locator article "FOLD OC"	
	Wylie "Survival Information Storage Systems"	
	Microsoft Word "Learning Microsoft Word 7.0"	

EXAMINER J Jay Chung DATE CONSIDERED 5/15/06

OIPE
FEB 16 2006
PATENT & TRADEMARK OFFICE

INFORMATION DISCLOSURE CITATION *(Use several sheets if necessary)*	Docket Number (Optional) 6851-02-CIP3	Application Number 10/155,192
Title: Data Security System and Method with Parsing and Dispersion Techniques	Applicant(s) Ron M. Redlich, et al.	
	Filing Date May 23, 2002	Group Art Unit 2143

U.S. PATENT DOCUMENTS

Examiner Initial	Document Number	Date	Name	Class	Subclass	Filing Date if Appropriate
JPC	5,960,080	9/28/1999	Fahlman	380	4	
	6,602,298	8/05/2003	Kirshenbaum	715	502.2	
	6,487,538	11/26/2002	Gupta	705	24	
	6,771,290	8/03/2004	Hoyle	345	745	
	US2002/0116641	8/22/2002	Mastrianni	713	201	
	US2004/0054630	3/18/2004	Ginter	705	53	

FOREIGN PATENT DOCUMENTS

	Document Number	Date	Country	Class	Subclass	Translation YES NO

OTHER DOCUMENTS (Including Author, Title, Date, Pertinent Pages, Etc.)

EXAMINER J Yong Chung DATE CONSIDERED 5/15/06

Index of Claims

Application/Control No.	Applicant(s)/Patent under Reexamination
10/155,192	REDLICH ET AL.
Examiner	**Art Unit**
Ji-Yong D. Chung	2143

√	Rejected	–	(Through numeral) Cancelled	N	Non-Elected	A	Appeal
=	Allowed	÷	Restricted	I	Interference	O	Objected

Claim		Date
Final	Original	09/15/06
	1	
	2	
	3	
	4	
	5	
	6	
	7	
	8	
	9	
	10	
	11	
	12	
	13	
	14	
	15	
	16	
	17	
	18	
	19	
	20	
	21	
	22	
	23	
	24	
	25	
	26	
	27	
	28	
	29	
	30	
	31	
	32	
	33	
	34	
	35	
	36	
	37	
	38	
	39	
	40	
	41	
	42	
	43	
	44	
	45	
	46	
	47	
	48	
	49	
	50	

Claim		Date
Final	Original	
	51	
	52	
	53	
	54	
	55	
	56	
	57	
	58	
	59	
	60	
	61	
	62	
	63	
	64	
	65	
	66	
	67	
	68	
	69	
	70	
	71	
	72	
	73	
	74	
	75	
	76	
	77	
	78	
	79	
	80	
	81	
	82	
	83	
	84	
	85	
	86	
	87	
	88	
	89	
	90	
	91	
	92	
	93	
	94	
	95	
	96	
	97	
	98	
	99	
	100	

Claim		Date
Final	Original	
	101	
	102	
	103	
	104	
	105	
	106	
	107	
	108	
	109	
	110	
	111	
	112	
	113	
	114	
	115	
	116	
	117	
	118	
	119	
	120	
	121	
	122	
	123	
	124	
	125	
	126	
	127	
	128	
	129	
	130	
	131	
	132	
	133	
	134	
	135	
	136	
	137	
	138	
	139	
	140	
	141	
	142	
	143	
	144	
	145	
	146	
	147	
	148	
	149	
	150	

gDC

Search Notes	Application/Control No.	Applicant(s)/Patent under Reexamination	
	10/155,192	REDLICH ET AL.	
	Examiner	Art Unit	
	Ji-Yong D. Chung	2143	

SEARCHED			
Class	Subclass	Date	Examiner

INTERFERENCE SEARCHED			
Class	Subclass	Date	Examiner

SEARCH NOTES (INCLUDING SEARCH STRATEGY)		
	DATE	EXMR
EAST (See attached updated seach history)	5/15/2006	JDC

JDC

IN THE SPECIFICATION

[0001] This is a continuation-in-part of patent application no. 10/008,209 and 10/008,218, both filed on December 6, 2001, which are continuations-in-part of patent application no. 09/916,397 filed July 27, 2001 which is a regular patent application is based upon provisional patent application no. 60/260,398, filed ~~June 5,~~ January 9, 2001; application no. 60/287,813, filed on May, 2, 2001; application no. 60/267,944, filed February 12, 2001; application no. 60/247,242, filed November 13, 2000 and application no. 60/247,232, filed November 13, 2000.

IN THE CLAIMS

1. (currently amended) A method of securing data having one or more security sensitive words, characters or icons in a computer system with a plurality of physically distinguishable memories respectively designated as a remainder store and a plurality of physically distinct extract stores corresponding to a plurality of security levels for respective subsets of said security sensitive words, characters or icons ~~(FIG. 6, store 1, 2 or 3; FIG. 7, step 422)~~ comprising:

extracting said security sensitive words, characters or icons from said data to obtain extracted data and remainder data therefrom;

storing said extracted data and said remainder data in said respective extract stores corresponding to an associated security level and said remainder store, respectively; and,

permitting full or partial reconstruction of said data via said extracted data from respective extract stores and remainder data only in the presence of a predetermined security clearance corresponding to said associated security level for said respective extract stores.

2. (original) A method as claimed in claim 1 including filtering said data while extracting.

3. (previously amended) A method as claimed in claim 2 including defining a plurality of filters for said respective subsets of security sensitive words, characters or icons and the method includes one of storing or destroying one or more of said plurality of filters after storing said extracted data.

4. (original) A method as claimed in claim 3 including encrypting one or both of said extracted data and said remainder data.

5. (original) A method as claimed in claim 4 wherein the step of permitting reconstruction includes decrypting one or both of said extracted data and remainder data.

6. (previously amended) A method as claimed in claim 1 including establishing a plurality of security levels each with a respective security clearance, each said subset of said security sensitive words, characters or icons being correlated with said plurality of security levels and the step of permitting reconstruction including the step of permitting a plurality of partial reconstructions of said data in the presence of respective ones of said plurality of security clearance levels.

7. (currently amended) A method of securing data with a parsing algorithm via a computer system having physically distinguishable memories designated as a remainder store and a plurality of physically distinct extract stores corresponding to a plurality of security levels for respective subsets of said security sensitive words, characters or icons comprising:

parsing said data to obtain extracted data for respective associated security levels and remainder data therefrom;

storing said extracted data and said remainder data in said extract stores and said remainder store, respectively; and,

permitting full or partial reconstruction of said data via said extracted data from said respective extract stores and remainder data only in the presence of a predetermined security clearance corresponding to said associated security level for said respective extract stores.

8. (original) A method as claimed in claim 7 wherein the step of parsing parses the data granularly.

9. (original) A method as claimed in claim 7 wherein the step of parsing uses said parsing algorithm which is based on one or more of a bit count, a word, a word count, a page count, a line count, a paragraph count, an identifiable document characteristic, an identifiable word characteristic,

an identifiable icon characteristic, an identifiable data object characteristic, capital letters, italics, and underline.

10. (original) A method as claimed in claim 7 wherein said data is financial data and the step of parsing uses said parsing algorithm based upon one of characters in said data and character count in said data.

11. (previously amended) A method as claimed in claim 7 including establishing a plurality of security levels each with a respective security clearance, permitting increasingly greater reconstruction if said data being correlated with increasingly higher levels of said plurality of security levels includes greater ~~securyt~~ security level clearances, and the step of permitting reconstruction including the step of permitting a plurality of partial reconstructions of said data in the presence of respective ones of said plurality of security clearance levels.

12. (original) A method as claimed in claim 7 including a further data security step of one from the group of storing said data without encryption, storing said data with encryption and destroying said data; said further data security step being independent of said parsing, storing and permitting reconstruction.

13. (original) A method as claimed in claim 1 including displaying an advertisement prior to one or the other or both of said steps of extracting and permitting reconstruction.

14. (original) A method as claimed in claim 13 including providing a hyperlink to a vendor's web site with said advertisement.

15. (original) A method as claimed in claim 1 including assessing a charge for extracting, storing or permitting reconstruction.

16. (currently amended) A computer readable medium containing programming instructions for securing data having one or more security sensitive words, characters or icons in a computer system with a plurality physically distinguishable memories respectively designated as a remainder store and a plurality of physically distinct extract stores corresponding to a plurality of security levels for respective subsets of said security sensitive words, characters or icons comprising:

extracting said security sensitive words, characters or icons from said data to obtain extracted data and remainder data therefrom;

storing said extracted data and said remainder data in said respective extract stores corresponding to an associated security level and said remainder store, respectively; and,

permitting full or partial reconstruction of said data via said extracted data from respective extract stores and remainder data only in the presence of a predetermined security clearance corresponding to said associated security level for said respective extract stores.

17. (original) A medium with programming instructions as claimed in claim 16 including filtering said data while extracting.

18. (currently amended) A medium with programming instructions as claimed in claim 16 including defining a plurality of filters for respective subsets of security sensitive words, characters or icons and ~~the method~~ includes one of storing or destroying one or more of said plurality of filters after storing said extracted data.

19. (original) A medium with programming instructions as claimed in claim 18 including encrypting one or both of said extracted data and said remainder data.

20. (original) A medium with programming instructions as claimed in claim 19 wherein the step of permitting reconstruction includes decrypting one or both of said extracted data and remainder data.

21. (previously amended) A medium with programming instructions as claimed in claim 16 including establishing a plurality of security levels each with a respective security clearance, each said subset of said security sensitive words, characters or icons being correlated with said plurality of security levels and the step of permitting reconstruction including the step of permitting a plurality of partial reconstructions of said data in the presence of respective ones of said plurality of security clearance levels.

22. (currently amended) A computer readable medium containing programming instructions for securing data with a parsing algorithm via a computer system having physically distinguishable memories designated as a remainder store and a plurality of physically distinct extract stores corresponding to a plurality of security levels for respective subsets of said security sensitive words, characters or icons comprising:

parsing said data to obtain extracted data for respective associated security levels and remainder data therefrom;

storing said extracted data and said remainder data in said respective extract stores and said remainder store, respectively; and,

permitting full or partial reconstruction of said data via said extracted data from said respective extract stores and remainder data only in the presence of a predetermined security clearance corresponding to said associated security level for said respective extract stores.

23. (original) A medium with programming instructions as claimed in claim 22 wherein the step of parsing parses the data granularly.

24. (original) A medium with programming instructions as claimed in claim 22 wherein the step of parsing uses said parsing algorithm which is based on one or more of a bit count, a word, a word count, a page count, a line count, a paragraph count, an identifiable document characteristic, an identifiable word characteristic, an identifiable icon characteristic, an identifiable data object characteristic, capital letters, italics, and underline.

25. (original) A medium with programming instructions as claimed in claim 22 wherein said data is financial data and the step of parsing uses said parsing algorithm based upon one of characters in said data and character count in said data.

26. (currently amended) A medium with programming instructions as claimed in claim 22 including establishing a plurality of security levels each with a respective security clearance, permitting increasingly greater reconstruction if said data being correlated with increasingly higher levels of said plurality of security levels includes greater ~~securyt~~ security level clearances, and the step of permitting reconstruction including the step of permitting a plurality of partial reconstructions of said data in the presence of respective ones of said plurality of security clearance levels.

27. (original) A medium as claimed in claim 22 including a further data security step of one from the group of storing said data without encryption, storing said data with encryption and destroying said data; said further data security step being independent of said parsing, storing and permitting reconstruction.

REMARKS

This amendment responds to the final Office Action dated June 1, 2006.

In the Office Action dated June 1, 2006, on pages 2-11, the patent examiner rejects all the claims as being non-patentable in view of certain prior art disclosed in the following references:

U.S. Patent No. 5,960,080 to Fahlman et al.
U.S. Patent No. 6,602,298 to Kirshenbaum
U.S. Patent No. 6,073,165 to Narasimhan et al.
U.S. Patent No. 6,487,538 to Gupta
U.S. Patent No. 6,771,290 to Hoyle

Previously in the Office Action dated August 11, 2005, the examiner, rejected all claims 1-27 as being non-patentable in view of:

U.S. Patent No. 5,960,080 to Fahlman
U.S. Patent No. 6,602,298 to Kirshenbaum
U.S. Patent No. 6,487,538 to Gupta
U.S. Patent No. 6,771,290 to Hoyle
Published U.S. Application 2002/0116641 to Mastrianni

In the present Office Action, the examiner confirmed that Mastrianni '641 is not prior art.

In the parent case, Serial No. 09/916,397, filed July 27, 2001, examiner Pyzocha rejected the claims in light of Kirshenbaum '298 (U.S. Patent No. 6,602,298), Lamm '907 (U.S. Patent No. 6,078,907) and/or Fahlman '080 (U.S. Patent No. 5,960,080) but ultimately withdrew such rejections when the claims were directed to a "multi-level security" system wherein different security words, for different security levels, were extracted and stored in different extract stores corresponding to respective security levels. Full or partial reconstruction is permitted with different security clearances for each security level (corresponding to different extract stores). The parent case 09/916,397 has been allowed after another search. A sample of one allowed claim is attached as an Addendum

hereto. Examiner Pyzocha located new reference, U.S. Patent No. 6,253,203 to O'Flaherty, during that further search but did not reject the claims based thereon.

In this case, independent claims 1, 7, 16 and 22 were amended on February 16, 2006 to refer to multiple levels of security. Currently, claims 1, 7, 16 and 22 are amended to refer to "physically distinguishable memories" and "physically distinct extract stores" and "full or partial reconstruction ... from respective extract stores ...[with] predetermined security clearances." Claims 11 and 18 and 26 are amended to correct typographic errors. The specification has been amended to correct the claim for priority (see supplemental declaration). Claim 1 is an example which refers to multiple security levels:

> A method of securing data having one or more security sensitive words... with a plurality of physically distinguishable memories respectively designated as a remainder store and a plurality of physically distinct extract stores corresponding to a plurality of security levels for respective subsets of said security sensitive words, characters or icons comprising:
>
> extracting ...
>
> storing said extracted data and said remainder data in said respective extract stores corresponding to an associated security level and said remainder store, respectively; and,
>
> permitting full or partial reconstruction of said data via said extracted data from respective extract stores and remainder data only in the presence of a predetermined security clearance corresponding to said associated security level for said respective extract stores.

It is respectfully submitted that the amended claims 1, 7, 16 and 22 all include the concept that various levels of secret data are stored in "respective extract stores" which are physically distinguishable and physically distinct for multiple levels of security with multiple levels of secure storage.

In summary, Kirshenbaum '298 and/or Fahlman '080 do not show, teach or suggest multiple extractions of security sensitive words related to each of a plurality of security levels, and separate

storage of those security sensitive words in different, physically distinct secured locations for each security level ("respective extract stores"), and the requirement that the user input a password ("security clearance") for each security level to reconstruct the data from "respective extract stores."

The advantages of the multiple security level - multiple dispersal and storage claim are:

- distributed, secure storage of multiple levels of secure data
- less risk of loss of secure data by attacks (intrusions) due to the dispersed storage of data
- system provides more physical barriers to electronic intrusions into storage sites
- avoids single point of machine-security clearance failure (possible with prior art systems)
- enhances the greater sharing of data by granular separation and dispersed storage of data at multiple levels of security

Kirshenbaum '298 does not show separate storage of secured data, separate and apart from unsecured data.[1] Both secured and non-secured data is stored in a single database 14. Col. 3, lines 40-44; col. 5, lines 7-10 ("The data set is stored in a database ... the document comprises secure portion and non-secure portions"); col. 5, lines 36-37 ("to retrieve those secure and non-secure portions of the document and to send the retrieved portions of the document to the output device.").

Contrary to the Examiner's contention, Kirshenbaum '298 does not show nor teach nor suggest physically distinguishable memories, nor physically distinct extract stores. The Examiner suggests that the different portions of the single document in Kirshenbaum '298 represent different stores, which different stores can be called "extract stores" (Examiner's characterization). However, it is well known that computer memory, when it stores a single document, does not locate all portions of that document in a single, physically distinct and physically distinguishable memory location. The document storage location, and all parts thereof, are stored at generally randomly assigned locations on the hard drive or memory disk. The randomness is based, in part, upon the

[1] Due to the confusing nature of Kirshenbaum '298, a brief analysis is provided here in conjunction with Fahlman '080. A detailed analysis is presented later.

storage of other documents on the hard drive or memory disk. Physically separate memories with physically distinct extract stores are not the same as the general concept of storing a document in a general purpose computer. The confusion caused by the ambiguity in Kirshenbaum '298 is described in detail in Applicant's Detailed Analysis at the end of these Remarks.

To have physically distinguishable memories and physically distinct extract stores, the location of the stores must be effectively static. These effectively static physically distinguishable memories and physically distinct extract stores are almost a necessity in light of the secure nature of the entire system. In other words, to maintain multiple security levels, each with a respective security clearance, the corresponding extract stores must be defined and confined within certain physically distinguishable memories and physically distinct extract stores. Otherwise, one could not maintain a high degree of security for the entire system.

If the Kirshenbaum "store anywhere" system was employed, the security system could be easily compromised by a hack attack on the document itself. With physically distinguishable memories and physically distinct extract stores, each with a separate security clearance, the quality of security is greatly enhanced.

Kirshenbaum '298 does not have storing said remainder [non-secret] data in said remainder store. Kirshenbaum '298 teaches away from this claimed aspect of the invention.

Kirshenbaum '298 does not show extraction of many levels of secured data from a document. At best, Kirshenbaum '298 discloses scanning and storing of identifying codes but the identifier codes are not secured data. "In a particular embodiments, the processor is configured to identify a security clearance level of a user and then to enable the user to print portions of a document at any security level below the user's level." Col. 5, line 54. "The identifier codes printed on non-secure

portions of a document are machine readable ... and are provided to a processor by extracting the identifier codes from the machine readable format with a scanning machine." Col. 5, line 64 - Col. 6, line 1.

The present invention requires multiple security levels, each having a sub-set of security sensitive words, extraction and storage for each of the security levels to respective physically distinct extract stores, a plurality of predetermined security clearances, and particular security clearances needed to reconstruct data from each of the respective extract stores.

In the present invention, different security clearances must be input to reconstruct data from each extract store (each security level storage facility) in order to obtain the data and permit full or partial reconstruction. Therefore, the present invention requires multiple extraction, each extraction for each security level, separate storage thereof, storage of the remainder or non-secured data, and, permitting full or partial reconstruction of said data via said extracted data and remainder data only with predetermined security clearances for respective extract stores.

Fahlman '080 does not show, teach or suggest a remainder store for non-secured data, multiple security levels, multiple extraction of security data, storage of extract data at multiple levels, nor presentment of corresponding security codes at each security level. In fact, nowhere does Fahlman '080 discuss password or security clearance control.

In Fig. 1, step 111 of Fahlman "automatically" merges sensitive information into non-sensitive information. In FIG. 2, step 217, the Fahlman system again "automatically" merges secured data with un-secured data. Fahlman discusses merging secured data with unsecured data at col. 2, line 22, col. 2, line 34, and col. 2, line 49, and col. 2, line 54 and col. 2, line 40. Fahlman '080 discloses identifying security information and extracting that information and replacing it with

place holders. Col. 2, line 37 and col. 3, line 47. "The sanitized message is then transmitted with a low level of security." Col. 3, line 54. The sensitive information is stripped from the original message and stored in a separate location or together with the sanitized message. Col. 3, line 64. A map showing the location of the security information is also stored with the secured information. Col. 3, line 63. "Then, in step 111, the sensitive terms received from the second path are merged with the sanitized message to create a final confidential message." Col. 4, line 1. The examiner should note that Fahlman does not discuss reconstruction or merger in the presence of any type of security clearance. In contrast, the present invention requires multiple security clearances, each unique to a security level, wherein these secured extract stores are physically distinct, in physically distinguishable memories. Fahlman also states: "Then, in step 217, the sensitive terms are automatically merged back into the serviced message to create a final message." Col. 5, line 27. No discussion of multi-level security clearance is noted. The same is true regarding merger of security information and non-secured information at col. 6, line 62.

Fahlman does not seem to store remainder or non-secure data in a separate remainder store location apart from secure data. Fahlman (1) identifies secret words, (2) replaces the words with placeholders, then (3) transmits the "sanitized message." Fig. 1, step 107, Fig. 2, step 209, col 2, line 19, col. 2, line 34, col. 2, line 49, col. 3, line 54 (transmission - no storage of non-secure data), col. 4, line 64 (transmission - no separate storage). Fahlman does discuss storing the sanitized message, that is, the non-secure data and the placeholders, but not separate storage of the remainder data. Col. 3, line 63, col. 4, line 47, col. 5, line 1. In the present invention, remainder, non-secret data is stored separately from secret data. See claim 1, "storing" step. Fahlman teaches away from this aspect of the present invention.

Fahlman does not discuss multiple security levels nor multiple storage sites for each security level. Since multiple security levels are not addressed in Fahlman, multiple security clearance codes to obtain access and reconstruct the secured data from multiple extract stores is not discussed. Since Fahlman does not describe any type of password or security clearance procedures, there is no need for a plurality of predetermined security clearances to reconstruct said data from multiple extract stores only in the presence of said predetermined security clearances.

There is no motivation nor suggestion to combine Kirshenbaum's presentment of a security code with Fahlman's single level extraction and storage of secret data. There is no suggestion nor motivation to use security codes to reconstruct "full or partial" data from respective extract stores in the presence of such security codes.

DETAILED ANALYSIS OF Kirshenbaum '298

The technical disclosure of Kirshenbaum '298 is difficult to read and understand. Applicant herein will first reproduce key portions of the Kirshenbaum '298 text without comment, and then provide commentary. Applicant identifies the three different embodiments in Kirshenbaum '298.

"Content manipulator 12 is utilized to create a communication 60 having a plurality of sub-components (62, 64, 66, 68, 70, 72 and 74)." col. 3, line 20, herein "3/20."

Embodiment One

"In the embodiment of FIG. 1, content manipulator 12 is in data communication with a database 14 and is configured to store communication 60 within database 14." 3/33.

> Content manipulator 12 is preferably configured to save a plurality of machine-readable identifier codes associated with communication 60 in database 14, and which identify different sub-components (62, 64, 66, 68, 70, 72 and 74) of communication 60. Database 14 can comprise, for example, a network, such as, for example, the Internet. Database 14 is configured for electronic storage of the sub-

components of communication 60 and the machine-readable identifier codes associated with communication 60.

3/39.

"Printer 16 is utilized to print sub-components 62, 64 and 66 of communication 60." 3/52.

"Processor 32 is configured to identify all sub-components of communication 60 that are stored on database 14, and to retrieve at least one sub-component from database 14 that was not printed as part of document 24." 4/15.

"The sub-components retrieved from database 14 by processor 32 are sent to output device 34 and subsequently displayed as a second document 44." 2/33.

Embodiment Two

"Another particular method of application of the system 10 of FIG. 1 is to protect secure portions of a document while enabling distribution of non-secure portions of the document (the non-secure portions can be separate portions from the secure portions, or can be non-secure variants of the secure portions)." 4/48.

> In an initial step of FIG. 2 (labeled as 100) a data set is created which corresponds to a document. The data set can be created utilizing, for example, the content manipulator 12 described above with reference to FIG. 1. The data set is stored in a database (which can comprise, for example, the database 14 of FIG. 1). The document comprises secure portions and non-secure portions. Such secure portions and non-secure portions can correspond to sub-components 62, 64, 66, 68, 70, 72 and 74 of a document 60 described above with reference to FIG. 1.

5/2.

"The processor is configured to retrieve both secure and non-secure portions of the document, and to send the retrieved portions of the document to the output device. The processor is further configured to recognize if an access code provides clearance to the secure portions of the document." 5/36.

> In particular embodiments, a user may be required to input its individual access code before any portions of a document are sent to the output device. In other embodiments, the user may be required to input its individual access code only before secure portions of the document are sent to the output device. In particular embodiments, the processor is configured to identify a security clearance level of a user and then to enable the user to print portions of a document at any security level below the user's level.

5/48.

Embodiment Three

> Another application of the present invention is described with reference to a flow chart of FIG. 3. At an initial step 200, a data set is provided on the Internet. The data set has an Internet address associated therewith (for purposes of interpreting ... includes any information that locates the data set on the internet, such as, for example, a URL). At a subsequent step 202, the Internet address is displayed on a substrate. The substrate can comprise, for example, paper, ... display screen of a television, ... a movie theater, or ... a billboard. In applications in which a URL address is broadcast by radio or television, the substrate can comprise a piece of paper upon which a user has jotted the address.

6/9.

"At a next step 204, the address is provided to a machine configured to digitize the address, link to the address on the Internet, and download at least portion of the data set." 6/25.

APPLICANT'S ANALYSIS

Applicant respectfully submits that Kirshenbaum '298 (a) poorly defines and uses the terms "document" and "data set" and "database" and hence poorly describes three different systems. By defining database 14 as "comprising" a network and the Internet, Kirshenbaum '298 has created confusion as to how the sub-components are stored. The following passage states that "identifier codes" are used to "identify different sub-components... [62 - 74]" and these identifier codes are "saved" in the database 14, and, by definition, the identifier codes are saved on the Internet.

> Content manipulator 12 is preferably configured to save a plurality of machine-readable identifier codes associated with communication 60 in database 14, and which identify different sub-components (62, 64, 66, 68, 70, 72 and 74) of communication 60. Database 14 can comprise, for example, a network, such as, for example, the Internet. Database 14 is configured for electronic storage of the sub-components of communication 60 and the machine-readable identifier codes associated with communication 60.

3/39.

Therefore, the sub-components 62 - 74 are always in given storage locations on the Internet, and the identifier codes associated with sub-components 62 - 74 are subsequently "saved" on the Internet. There is no EXTRACTION of sub-components 62 - 74. These sub-components are pre-set

and in existence in storage sites in the database/Internet. Kirshenbaum does not disclose a multiple extraction process to different security extract stores.

This interpretation of Kirshenbaum '298 is consistent with the Third Embodiment wherein the data set includes an Internet address which is used by consumer to locate and obtain all or portions of the document.

> Another application of the present invention is described with reference to a flow chart of FIG. 3. At an initial step 200, a data set is provided on the Internet. The data set has an Internet address associated therewith (information that locates the data set on the internet, such as, for example, a URL). At a subsequent step 202, the Internet address is displayed on a substrate. The substrate can comprise, for example, paper, ... display screen of a television, ... a movie theater, or ... a billboard. In applications in which a URL address is broadcast by radio or television, the substrate can comprise a piece of paper upon which a user has jotted the address.

6/9.

"At a next step 204, the address is provided to a machine configured to digitize the address, link to the address on the Internet, and download at least portion of the data set." 6/25. Multiple Internet addresses can be displayed, the user can select one or more and the system downloads selected site data. See 6/36.

However, the Second Embodiment (beginning at 4/48) seems to be completely different than the First and Third Embodiments in that the Second Embodiment discusses retrieval of secure and non-secure data, not the initial storage process of the data set.

> In an initial step of FIG. 2 (labeled as 100) a data set is created which corresponds to a document. The data set can be created utilizing, for example, the content manipulator 12 described above with reference to FIG. 1. The data set is stored in a database (which can comprise, for example, the database 14 of FIG. 1). The document comprises secure portions and non-secure portions. Such secure portions and non-secure portions can correspond to sub-components 62, 64, 66, 68, 70, 72 and 74 of a document 60 described above with reference to FIG. 1.

5/2.

However, the First Embodiment clearly states that the machine-readable identifier codes are "saved" in the database 14 (which maybe on the Internet).

> Content manipulator 12 is preferably configured to save a plurality of machine-readable identifier codes associated with communication 60 in database 14, and which identify different sub-components (62, 64, 66, 68, 70, 72 and 74) of communication 60.

3/39.

In conclusion, Kirshenbaum '298 does not show, teach or suggest that the sub-components 62 - 74 are selectively extracted, based upon some security criteria, then stored in separate memories on the Internet, but rather: "Content manipulator 12 is ... configured to save ... identifier codes associated with communication 60 in database 14, and which [identifier codes] identify different sub-components (62, 64, 66, 68, 70, 72 and 74) of communication 60." 3/39. "Database 14 [in which is saved the identifier codes] can comprise, for example, a network, such as, for example, the Internet." 3/45. Kirshenbaum '298 then goes on the globally state: "Database 14 is configured for electronic storage of the sub-components of communication 60 and the machine-readable identifier codes associated with communication 60." 3/46. This global statement, when understood in context with the specific statement regarding "saving identifier codes in a database on the Internet" is not a clear enough disclosure to show, teach or suggest the specific, claimed steps of the present invention.

> A method of securing data having one or more security sensitive words... with a plurality of <u>physically distinguishable</u> memories respectively designated as a remainder store and a plurality of <u>physically distinct</u> extract stores corresponding to a plurality of security levels for respective subsets of said security sensitive words, characters or icons comprising:
>
> extracting ...
>
> storing said extracted data and said remainder data in said respective extract stores corresponding to an associated security level and said remainder store, respectively; and,
>
> permitting <u>full or partial</u> reconstruction of said data via said extracted data from respective extract stores and remainder data only in the presence of a predetermined security clearance corresponding to said associated security level for said respective extract stores.

To make the argument differently, if the Examiner believes that a database is any organized set of data, and if the person of ordinary skill in the art understands, according to Kirshenbaum '298, that the Internet is included in the "database," then each and every electronic document on any network falls within this definition. The communication 60 and all sub-components 62 - 74 are in existence on the Internet-network. However, just because every electronic document falls within the definition, this does not explain the (a) extraction and storage process into physically distinct extract stores based upon multiple levels of security; and (b) reconstruction based upon multiple security clearances from those multiple secured extract stores.

It is respectfully submitted that the Examiner is unjustifiably extrapolating Kirshenbaum's specific statement regarding "saving identifier codes in a database on the Internet" 3/39, and, based upon hindsight learned only after reading the present invention, assumes that this "saving identifier codes" is the same as extraction and storage of security words or icons into physically distinct extract stores based upon multiple levels of security. [2]

Simply put, Kirshenbaum '298 does not show extraction for multiple levels of security. The Second Embodiment in Kirshenbaum '298 assumes the document, subject to the security system, is already present in database. 5/2 [3]

---

[2] For sake of clarity, Kirshenbaum '298 states: "Content manipulator 12 is preferably configured to save a plurality of machine-readable identifier codes associated with communication 60 in database 14, and which identify different sub-components (62, 64, 66, 68, 70, 72 and 74) of communication 60. Database 14 can comprise, for example, a network, such as, for example, the Internet. Database 14 is configured for electronic storage of the sub-components of communication 60 and the machine-readable identifier codes associated with communication 60." 3/39.

[3] "In an initial step of FIG. 2 (labeled as 100) a data set is created which corresponds to a document. The data set can be created utilizing, for example, the content manipulator 12 described above with reference to FIG. 1. The data set is stored in a database (which can comprise, for example, the database 14 of FIG. 1). The document comprises secure portions and non-secure portions. Such secure portions and non-secure portions can correspond to sub-components 62, 64, 66, 68, 70, 72 and 74 of a document 60 described above with reference to FIG. 1." 5/2

Each of the references, Kirshenbaum '298, and Fahlman '080, describe complete systems and there is no reason to add to or substitute portions of one disclosure with another. For example, there is no reason to combine Kirshenbaum's singular document storage (the database document in existence on the Internet) and reproduction system with Fahlman's secret transmission system. It is improper to take the multiple security level - password system of Kirshenbaum '298 (Second Embodiment), the single extract process of Fahlman '080, and somehow extrapolate into (a) extraction of multiple security sensitive words or icons and (b) separate storage of extracted items into physically distinct memories, and (c) full or partial reconstruction from those separate extract stores in the presence of predetermined respective security clearances.

The Manual of Patent Examining Procedures ("MPEP") explains that it is not obvious to modify a system with a plurality of sensors controlling a plurality of valves to create the inventive and patentable single sensor which controls a plurality of valves:

> In *In re Kotzab,* 217 F.3d 1365, 55 USPQ2d 1313 (Fed.Cir. 2000), the claims were drawn to an injection molding method using a single temperature sensor to control a plurality of flow control valves. The primary reference disclosed a multizone device having multiple sensors, each of which controlled an associated flow control valve, and also taught that one system may be used to control a number of valves. The court found that there was insufficient evidence to show that one system was the same as one sensor. While the control of multiple valves by a single sensor rather than by multiple sensors was a "technologically simple concept," there was no finding "as to the specific, understanding or principle within the knowledge of the skilled artisan" that would have provided the motivation to use a single sensor as the system to control more than one valve. 217 F.3d at 1371, 55 USPQ2d at 1318.

MPEP § 2143.01.

In the present invention, Applicant has developed a system designed to safeguard a plurality of security levels with specific dispersal and retrieval techniques. The single data storage systems of the prior art Fahlman '080 and Kirshenbaum '298 cannot be extended to the claimed multi-level security storage systems.

> The mere fact that references can be combined or modified does not render the resultant combination obvious unless the prior art also suggests the desirability

> of the combination. *In re Mills*, 916 F.2d 680, 16 USPQ2d 1430 (Fed. Cir. 1990) (Claims were directed to an apparatus for producing an aerated cementitious composition by drawing air into the cementitious composition by driving the output pump at a capacity greater than the feed rate. The prior art reference taught that the feed means can be run at a variable speed, however the court found that this does not require that the output pump be run at the claimed speed so that air is drawn into the mixing chamber and is entrained in the ingredients during operation. Although a prior art device "may be capable of being modified to run the way the apparatus is claimed, there must be a suggestion or motivation in the reference to do so." 916 F.2d at 682, 16 USPQ2d at 1432.).

MPEP § 2143.01.

In the prior art cited herein, there is no suggestion of the desirability of multi-level secured data storage, extraction for each security level, and full or partial reconstruction using multiple extract stores requiring multiple security codes for access. Further, the examiner's suggestion -- that one can take certain aspects of Kirshenbaum '298 and mix it with other aspects of Fahlman '080 -- "modifies" that reference to an unacceptable degree. "If proposed modification would render the prior art invention being modified unsatisfactory for its intended purpose, then there is no suggestion or motivation to make the proposed modification. *In re Gordon*, 733 F.2d 900, 221 USPQ 1125 (Fed. Cir. 1984)." MPEP § 2143.01. This change to Fahlman '080 modifies the operation of the reference improperly. "If the proposed modification or combination of the prior art would change the principle of operation of the prior art invention being modified, then the teachings of the references are not sufficient to render the claims prima facie obvious. *In re Ratti*, 270 F.2d 810, 123 USPQ 349 (CCPA 1959)." MPEP § 2143.01.

With respect to Gupta '538 and Hoyle '290, those references are cited by the examiner as secondary references. The secondary references do not cure the defects described above with respect to the primary references. With respect to filter Narasimhan '165, that reference does not cure the defects noted above with respect to the primary references.

Applicant respectfully requests that the examiner withdraw the rejections and approve the patentability of claims 1 - 27.

Respectfully submitted,

By _____

Robert C. Kain, Jr.
Reg. No. 30,648
Fleit, Kain, Gibbons, Gutman, Bongini & Bianco, P.L.
750 Southeast Third Avenue, Suite 100
Fort Lauderdale, FL 33316-1153
Telephone: 954-768-9002
Facsimile: 954-768-0158

Certificate of Mailing

I hereby certify that this correspondence is being ~~deposited with the U.S. Postal Service as First Class Mail in an envelope addressed~~ submitted by electronic filing to Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450 on June 20, 2006.

_____

Robert C. Kain, Jr.
Reg. No. 30,648

ADDENDUM

90. (parent case 09/916,397 - one of several allowed claims) A method for securing data in a computer network with one or more security sensitive words, characters or icons and a plurality of security levels each with a respective security clearance, subsets of said security sensitive words, characters or icons being correlated with respective ones of said plurality of security levels, said computer network having a plurality of computers interconnected together, each of said plurality of computers having a memory therein, one of said plurality of computers designated as a data input computer, said plurality of computers including a first computer designated as a remainder store and a further plurality of computers designated as extract stores for respective ones of said plurality of security levels, comprising:

extracting said security sensitive words, characters or icons for said plurality of security levels from said data to obtain extracted data for respective security levels and remainder data;

storing said extracted data in extract stores corresponding to respective security levels and said remainder data in said remainder store;

presenting a plurality of predetermined security clearances to obtain access to respective ones of said extract stores; and,

permitting full or partial reconstruction of said data via said extracted data and remainder data only in the presence of respective ones of said predetermined security clearances after presentment thereof.

\\Tiger\data share\RCK\CLIENTS\Redlich\Patents\6851-02cip3-amdt-2d-060906.wpd

DECLARATION AND POWER OF ATTORNEY FOR PATENT APPLICATION

As a below named inventor, I hereby declare that:

My residence, post office address and citizenship are as stated below next to my name,

I believe I am the original, first and sole inventor (if only one name is listed below) or an original, first and joint inventor (if plural names are listed below) of the subject matter which is claimed and for which a patent is sought on the invention entitled Data Security System and Method With Parsing and Dispersion Techniques the specification of which

(check one) ☐ is attached hereto.

☒ was filed on May 23, 2002

as Application Serial No. 10/155,192

and was amended on February 13, 2006 (if applicable).

I hereby state that I have reviewed and understand the contents of the above identified specification, including the claims, as amended by any amendment referred to above.

I acknowledge the duty to disclose information which is material to the patentability of this application in accordance with Title 37, Code of Federal Regulations, §1.56(a).

I hereby claim foreign priority benefits under Title 35, United States Code, §119 (a-d) or §365(b) of any foreign application(s) for patent or inventor's certificate, or §365(a) of any PCT international application which designated at least one country other than the United States of America, listed below and have also identified below any foreign application for patent or inventor's certificate, or any PCT international application having a filing date before that of the application on which priority is claimed:

Prior Foreign Application(s)			Priority Claimed	
(Number)	(Country)	(Day/Month/Year Filed)	☐ Yes	☐ No
(Number)	(Country)	(Day/Month/Year Filed)	☐ Yes	☐ No

I hereby claim the benefit under Title 35, United States Code, §120 of any United States application(s) listed below and, insofar as the subject matter of each of the claims of this application is not disclosed in the prior United States application in the manner provided by the first paragraph of Title 35, United States Code, §112, I acknowledge the duty to disclose material information as defined in Title 37, Code of Federal Regulations, §1.56(a) which occurred between the filing date of the prior application and the national or PCT international filing date of this application:

09/916,397 (Application Serial No.)	07/27/01 (Filing Date)	pending (Status) (patented, pending, abandoned)
10/008,209 (Application Serial No.)	12/6/01 (Filing Date)	pending (Status)
10/008,218 (Application Serial No.)	12/6/01 (Filing Date)	pending (Status) (patented, pending, abandoned)

I hereby claim the benefit under Title 35, United States Code, §119(e) of any United States provisional application(s) listed below:

60/260,398, filed Jan. 9, 2001 (Provisional Application No.)
60/287,813, filed May 2, 2001 (Provisional Application No.)
60/267,944, filed February 12, 2001 (Provisional Application No.)

60/247,242, filed November 12, 2000 (Provisional Application No.)
60/247,232, filed November 13, 2000 (Provisional Application No.)

I hereby declare that all statements made herein of my own knowledge are true and that all statements made on information and belief are believed to be true; and further that these statements were made with the knowledge that willful false statements and the like so made are punishable by fine or imprisonment, or both, under 18 U.S.C. 1001, and that such willful false statements may jeopardize the validity of the application or any patent issued thereon.
POWER OF ATTORNEY: As a named inventor, I hereby appoint the following attorney(s) and/or agents to prosecute this application and transact all business in the Patent and Trademark Office connected therewith.
Robert C. Kain, Jr., Reg. No. 30,648; Martin Fleit, Reg. No. 16,900; Jon A. Gibbons, Reg. No. 37,333; Jose Gutman, Reg. No. 35,171; and Steve Bongini, Reg. No. 40,917; Paul D. Bianco, Reg. No. 43,500.

Send Correspondence to: Robert C. Kain, Jr., **Customer Number 004897**
Fleit, Kain, Gibbons, Gutman & Bongini, P.L.
750 Southeast Third Avenue, Suite 100
Ft. Lauderdale, Florida 33316-1153
Direct Telephone Calls to: Robert C. Kain, Jr., (954) 768-9002

Ron M. Redlich
Full name of sole or first inventor

Inventor's signature — Date: 06-1-2006

Miami Beach, FL
Residence

U.S.
Citizenship

4201 Collins Avenue, Apt. 2103
Post Office Address

Miami Beach, FL 33140

Martin A. Nemzow
Full name of second joint inventor, if any

Inventor's signature — Date: 06-11-2006

Miami Beach, FL
Residence

U.S.
Citizenship

2915 Flamingo Drive
Post Office Address

Miami Beach, FL 33140

PTO/SB/08A (07-05)
Approved for use through 07/31/2006. OMB 0651-0031
U.S. Patent and Trademark Office; U.S. DEPARTMENT OF COMMERCE
Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it contains a valid OMB control number.

Substitute for form 1449/PTO

INFORMATION DISCLOSURE STATEMENT BY APPLICANT

(Use as many sheets as necessary)

Sheet 1 of 1

Complete if Known	
Application Number	10/155,192
Filing Date	May 23, 2002
First Named Inventor	Ron M. Redlich, et al.
Art Unit	2143
Examiner Name	Chung
Attorney Docket Number	6851-02-cip3

U. S. PATENT DOCUMENTS

Examiner Initials*	Cite No.[1]	Document Number Number-Kind Code[2] (if known)	Publication Date MM-DD-YYYY	Name of Patentee or Applicant of Cited Document	Pages, Columns, Lines, Where Relevant Passages or Relevant Figures Appear
		US- 6,253,203	06-26-2001	O'Flaherty et al.	
		US-			
		US-			
		US-			
		US-			
		US-			
		US-			
		US-			
		US-			
		US-			
		US-			
		US-			
		US-			
		US-			
		US-			
		US-			
		US-			
		US-			
		US-			
		US-			

FOREIGN PATENT DOCUMENTS

Examiner Initials*	Cite No.[1]	Foreign Patent Document Country Code[3]-Number[4]-Kind Code[5] (if known)	Publication Date MM-DD-YYYY	Name of Patentee or Applicant of Cited Document	Pages, Columns, Lines, Where Relevant Passages Or Relevant Figures Appear	T[6]

Examiner Signature		Date Considered	

*EXAMINER: Initial if reference considered, whether or not citation is in conformance with MPEP 609. Draw line through citation if not in conformance and not considered. Include copy of this form with next communication to applicant. [1] Applicant's unique citation designation number (optional). [2] See Kinds Codes of USPTO Patent Documents at www.uspto.gov or MPEP 901.04. [3] Enter Office that issued the document, by the two-letter code (WIPO Standard ST.3). [4] For Japanese patent documents, the indication of the year of the reign of the Emperor must precede the serial number of the patent document. [5] Kind of document by the appropriate symbols as indicated on the document under WIPO Standard ST.16 if possible. [6] Applicant is to place a check mark here if English language Translation is attached.

This collection of information is required by 37 CFR 1.97 and 1.98. The information is required to obtain or retain a benefit by the public which is to file (and by the USPTO to process) an application. Confidentiality is governed by 35 U.S.C. 122 and 37 CFR 1.14. This collection is estimated to take 2 hours to complete, including gathering, preparing, and submitting the completed application form to the USPTO. Time will vary depending upon the individual case. Any comments on the amount of time you require to complete this form and/or suggestions for reducing this burden, should be sent to the Chief Information Officer, U.S. Patent and Trademark Office, P.O. Box 1450, Alexandria, VA 22313-1450. DO NOT SEND FEES OR COMPLETED FORMS TO THIS ADDRESS. **SEND TO: Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450.**

If you need assistance in completing the form, call 1-800-PTO-9199 (1-800-786-9199) and select option 2.

Electronic Acknowledgement Receipt

EFS ID:	1084474
Application Number:	10155192
Confirmation Number:	9510
Title of Invention:	Data security system and method with parsing and dispersion techniques
First Named Inventor:	Ron M. Redlich
Customer Number:	4897
Filer:	Robert C. Kain
Filer Authorized By:	
Attorney Docket Number:	6851-2-CIP-3
Receipt Date:	20-JUN-2006
Filing Date:	23-MAY-2002
Time Stamp:	11:31:30
Application Type:	Utility
International Application Number:	

Payment information:

Submitted with Payment	no

File Listing:

Document Number	Document Description	File Name	File Size(Bytes)	Multi Part	Pages
1	Transmittal letter	6851-02-cip3-amdt-2d-060906-trans.pdf	279014	no	1

Warnings:					
Information:					
2	Amendment After Final	6851-02-cip3-amdt-2d-06090 6-spec.pdf	259615	no	1
Warnings:					
Information:					
3	Amendment After Final	6851-02-cip3-amdt-2d-06090 6-claims.pdf	3221099	no	6
Warnings:					
Information:					
4	Applicant Arguments/Remarks Made in an Amendment	6851-02-cip3-amdt-2d-06090 6-remarks.pdf	9951553	no	16
Warnings:					
Information:					
5	Oath or Declaration filed	6851-02cip3-Dec061106.pdf	93315	no	2
Warnings:					
Information:					
6	Information Disclosure Statement (IDS) Filed	6851-02-cip3-amdt-2d-06090 6-IDS.pdf	60419	no	1
Warnings:					
Information:					
This is not an USPTO supplied IDS fillable form					
			Total Files Size (in bytes):	13865015	

This Acknowledgement Receipt evidences receipt on the noted date by the USPTO of the indicated documents, characterized by the applicant, and including page counts, where applicable. It serves as evidence of receipt similar to a Post Card, as described in MPEP 503.

New Applications Under 35 U.S.C. 111
If a new application is being filed and the application includes the necessary components for a filing date (see 37 CFR 1.53(b)-(d) and MPEP 506), a Filing Receipt (37 CFR 1.54) will be issued in due course and the date shown on this Acknowledgement Receipt will establish the filing date of the application.

National Stage of an International Application under 35 U.S.C. 371
If a timely submission to enter the national stage of an international application is compliant with the conditions of 35 U.S.C. 371 and other applicable requirements a Form PCT/DO/EO/903 indicating acceptance of the application as a national stage submission under 35 U.S.C. 371 will be issued in addition to the Filing Receipt, in due course.

PATENTS

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re Application of Ron M. Redlich, et al.

Serial No. 10/155,192 Examiner Chung

Filed: May 23, 2002 Group 2143

For: Data Security System and Method With Parsing and Dispersion Techniques

AMENDMENT AFTER FINAL OFFICE ACTION

Mail Stop AF
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Sir:

This amendment responds to the final Office Action dated June 1, 2006. If a fee is necessary, please charge Deposit Account no. 03-1231. Applicant submits herewith a supplemental declaration to correct the claim for priority.

[Page left blank]

PATENT APPLICATION FEE DETERMINATION RECORD
Effective October 1, 2001

Application or Docket Number: 10/155,192
~~685-2 CIP3~~

CLAIMS AS FILED - PART I

	(Column 1)	(Column 2)
TOTAL CLAIMS	27	
FOR 05/23/02	NUMBER FILED	NUMBER EXTRA
TOTAL CHARGEABLE CLAIMS	27 minus 20=	* 7
INDEPENDENT CLAIMS	4 minus 3 =	* 1
MULTIPLE DEPENDENT CLAIM PRESENT	☐	

* If the difference in column 1 is less than zero, enter "0" in column 2

SMALL ENTITY TYPE ☐	RATE	FEE	OR	OTHER THAN SMALL ENTITY	RATE	FEE
	BASIC FEE	370.00	OR		BASIC FEE	740.00
	X$ 9=	63	OR		X$18=	
	X42=	42	OR		X84=	
	+140=		OR		+280=	
	TOTAL	475	OR		TOTAL	

02/16/06 **CLAIMS AS AMENDED - PART II**

AMENDMENT A	(Column 1) CLAIMS REMAINING AFTER AMENDMENT		(Column 2) HIGHEST NUMBER PREVIOUSLY PAID FOR	(Column 3) PRESENT EXTRA
Total	* 27	Minus	** 27	=
Independent	* 4	Minus	*** 4	=
FIRST PRESENTATION OF MULTIPLE DEPENDENT CLAIM				☐

SMALL ENTITY	RATE	ADDITIONAL FEE	OR	OTHER THAN SMALL ENTITY	RATE	ADDITIONAL FEE
	X$ 9=		OR		X$18=	
	X42=		OR		X84=	
	+140=		OR		+280=	
	TOTAL ADDIT. FEE		OR		TOTAL ADDIT. FEE	

06/20/06

AMENDMENT B	(Column 1) CLAIMS REMAINING AFTER AMENDMENT		(Column 2) HIGHEST NUMBER PREVIOUSLY PAID FOR	(Column 3) PRESENT EXTRA
Total	* 27	Minus	** 27	= 0
Independent	* 4	Minus	*** 4	= 0
FIRST PRESENTATION OF MULTIPLE DEPENDENT CLAIM				☐

SMALL ENTITY	RATE	ADDITIONAL FEE	OR	OTHER THAN SMALL ENTITY	RATE	ADDITIONAL FEE
	X$ 9=		OR		X$18=	
	X42=		OR		X84=	
	+140=		OR		+280=	
	TOTAL ADDIT. FEE	0	OR		TOTAL ADDIT. FEE	

Best Available Copy

AMENDMENT C	(Column 1) CLAIMS REMAINING AFTER AMENDMENT		(Column 2) HIGHEST NUMBER PREVIOUSLY PAID FOR	(Column 3) PRESENT EXTRA
Total	*	Minus	**	=
Independent	*	Minus	***	=
FIRST PRESENTATION OF MULTIPLE DEPENDENT CLAIM				☐

SMALL ENTITY	RATE	ADDITIONAL FEE	OR	OTHER THAN SMALL ENTITY	RATE	ADDITIONAL FEE
	X$ 9=		OR		X$18=	
	X42=		OR		X84=	
	+140=		OR		+280=	
	TOTAL ADDIT. FEE		OR		TOTAL ADDIT. FEE	

* If the entry in column 1 is less than the entry in column 2, write "0" in column 3.
** If the "Highest Number Previously Paid For" IN THIS SPACE is less than 20, enter "20."
*** If the "Highest Number Previously Paid For" IN THIS SPACE is less than 3, enter "3."
The "Highest Number Previously Paid For" (Total or Independent) is the highest number found in the appropriate box in column 1.

Document code: WFEE

United States Patent and Trademark Office
Sales Receipt for Accounting Date: 07/17/2006

TDADE1 SALE #00000001 Mailroom Dt: 06/20/2006 031231 10155192
01 FC:1806 180.00 DA

PATENTS

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re Application of Ron M. Redlich, et al.

Serial No. 10/155,192 Examiner Chung

Filed: May 23, 2002 Group 2143

For: Data Security System and Method With Parsing and Dispersion Techniques

SUPPLEMENTAL REMARKS FOR AMENDMENT AFTER FINAL OFFICE ACTION

Mail Stop AF
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Sir:

These are Supplemental Remarks for the Amendment filed June 20, 2006, responding to the final Office Action dated June 1, 2006. Applicant submits herewith a Terminal Disclaimer. The $65.00 fee for the Terminal Disclaimer is charged to Deposit Account no. 03-1231. If a fee is inadequate, please debit or credit Deposit Account no. 03-1231.

1. Application Papers enclosed:

   a. Transmittal Supplemental Remarks for Amendment after Final Office Action (1pg).

   b. Remarks (3 pgs.)

2. Terminal Disclaimer (2 pgs.)

[Page left blank]

REMARKS

These are supplemental remarks to the amendment of June 20, 2006 responsive to the final Office Action dated June 1, 2006.

In the Office Action dated August 11, 2005, the examiner issued a double patenting rejection based upon Serial Nos. 09/916,397 (now allowed); 10/008,209; 10/008,218; 10/155,525; 10/277,196; and 10/390,807. Applicant submits, under protest, a Terminal Disclaimer for the allowed parent case, Serial No. 09/916,397 and requests that the Examiner withdraw the double patenting rejection based upon Serial Nos. 10/008,209; 10/008,218; 10/155,525; 10/277,196; and 10/390,807 since those cases will not issue before the present case. Further, the enclosed Terminal Disclaimer linked to the Parent Case Serial Nos. 09/916,397 (now allowed) will limit the term of the present case to be coextensive with the Parent Case and therefore the present case could never be longer in duration compared to the "sibling" cases, 10/008,209; 10/008,218; 10/155,525; 10/277,196; and 10/390,807.

> If a "provisional" nonstatutory obviousness-type double patenting (ODP) rejection is the only rejection remaining in the earlier filed of the two pending applications, while the later-filed application is rejectable on other grounds, the examiner should withdraw that rejection and permit the earlier-filed application to issue as a patent without a terminal disclaimer.

MPEP 804 (I)(B)(1), p. 800-17.

> The filing of a terminal disclaimer to obviate a rejection based on nonstatutory double patenting is not an admission of the propriety of the rejection. *Quad Environmental Technologies Corp. v. Union Sanitary District*, 946 F.2d 870, 20 USPQ2d 1392 (Fed. Cir. 1991). The court indicated that the "filing of a terminal disclaimer simply serves the statutory function of removing the rejection of double patenting, and raises neither a presumption nor estoppel on the merits of the rejection."

MPEP 804.02(II), p. 800-32.

Applicant is submitting this terminal disclaimer for the parent case in order to expedite prosecution. Applicant disagrees that the Examiner adequately explained the link or combination of the present claims with those claims in one or more of Serial Nos. 10/008,209; 10/008,218; 10/155,525; 10/277,196; and 10/390,807 with the cited prior art.

> Since the analysis employed in an obviousness-type double patenting determination parallels the guidelines for a 35 U.S.C. 103(a) rejection, the factual inquiries set forth in *Graham v. John Deere Co.,* 383 U.S. 1, 148 USPQ 459 (1966), that are applied for establishing a background for determining obviousness under 35 U.S.C. 103 are employed when making an obvious-type double patenting analysis. These factual inquiries are summarized as follows: (A) Determine the scope and content of a patent claim relative to a claim in the application at issue; (B) Determine the differences between the scope and content of the patent claim as determined in (A) and the claim in the application at issue; (C) Determine the level of ordinary skill in the pertinent art; and (D) Evaluate any objective indicia of nonobviousness.
>
> The conclusion of obviousness-type double patenting is made in light of these factual determinations. Any obviousness-type double patenting rejection should make clear: (A) The differences between the inventions defined by the conflicting claims — a claim in the patent compared to a claim in the application; and (B) The reasons why a person of ordinary skill in the art would conclude that the invention defined in the claim at issue would have been< an obvious variation of the invention defined in a claim in the patent.

MPEP § 804(II)(B)(1) p. 800-21

It is respectfully submitted that the Examiner accept the enclosed Terminal Disclaimer and withdraw the double patenting rejection with respect to Serial Nos. 10/008,209; 10/008,218; 10/155,525; 10/277,196; and 10/390,807.

[page left blank]

Applicant respectfully requests that the examiner withdraw the rejections and approve the patentability of claims 1 - 27.

Respectfully submitted,

By /Robert Kain/

Robert C. Kain, Jr.
Reg. No. 30,648
Fleit, Kain, Gibbons, Gutman, Bongini & Bianco, P.L.
750 Southeast Third Avenue, Suite 100
Fort Lauderdale, FL 33316-1153
Telephone: 954-768-9002
Facsimile: 954-768-0158

Certificate of Mailing

I hereby certify that this correspondence is being filed via efiling with the Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450 on June 28, 2006.

/Robert Kain/
Robert C. Kain, Jr.
Reg. No. 30,648

\\TIGER\Data Share\RCK\CLIENTS\Redlich\Patents\6851-02cip3-amdt-3d-062806.wpd

PATENTS

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re Application of Ron M. Redlich, et al.

Serial No. 10/155,192 Examiner Chung

Filed: May 23, 2002 Group 2143

For: Data Security System and Method With Parsing and Dispersion Techniques

TERMINAL DISCLAIMER

Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Sir:

Petitioner, DigitalDoors, Inc., a Florida corporation, having its principal place of business at 4201 Collins Avenue, Suite 2103, Miami Beach, FL 33140, represents that it is the owner by assignment of all rights to application Serial No. 09916397, filed on 07/27/2001 and the captioned case, Serial No. 10155192, filed on 05/23/2002. The captioned application was assigned to DigitalDoors, Inc. and the assignment was filed with the original application. The evidentiary document accompanying or referred to in the instant terminal disclaimer has been reviewed by the undersigned and it is certified that to the best of assignee, DigitalDoors, Inc.'s knowledge and belief, title is in the assignee.

Petitioner, DigitalDoors, Inc., hereby disclaims the terminal part of any patent granted on the above identified application, which would extend beyond the expiration date of any patent granted on application Serial No. 09916397, filed on 07/27/2001, and, except as provided below, hereby agrees that any patent so granted on the above identified application shall be enforceable only for and during such period and that both patents will remain commonly owned.

Petitioner, DigitalDoors, Inc., is a small entity.

Petitioner does not disclaim any terminal part of any patent granted on the above identified application prior to the expiration date of the full statutory term of any patent granted on application Serial No. 09916397, filed on 07/27/2001 in the event that the latter patent: expires for failure to pay a maintenance fee, is held unenforceable, is found invalid, is statutorily disclaimed in whole or terminally disclaimed under 37 C.F.R. §1.321(a), has all claims canceled by a reexamination certificate, is reissued, or is otherwise terminated prior to the expiration of its statutory term as presently shortened by any terminal disclaimer, except for the separation of legal title stated above.

Petitioner expressly reserves and does not waive the right to extend the term of any patent granted on application Serial No. 09916397, filed on 07/27/2001 or any patent granted on the above identified application under any statute which permits such an extension of term.

I hereby declare that all statements made herein of our own knowledge are true, and that all statements made on information and belief are believed to be true; and further, that these statements were made with the knowledge that willful false statements, and the like so made, are punishable by fine or imprisonment, or both, under Section 1001 of Title 18 of the United States Code, and that such willful false statements may jeopardize the validity of the application or any patent issuing thereon.

DigitalDoors, Inc.

Dated: __June 28, 2006___

By: __/Robert Kain/_____
Robert C. Kain, Jr.
Reg. No. 30,648
Fleit, Kain, Gibbons, Gutman, Bongini & Bianco, P.L.
750 Southeast Third Avenue, Suite 100
Ft. Lauderdale, Florida 33316-1153
(954) 768-9002
Attorney of Record for Petitioner

\\TIGER\Data Share\RCK\CLIENTS\Redlich\6851-02-cip3-Terminal-discl.wpd

Electronic Patent Application Fee Transmittal

Application Number:	10155192
Filing Date:	23-May-2002
Title of Invention:	Data security system and method with parsing and dispersion techniques
First Named Inventor:	Ron M. Redlich
Filer:	Robert C. Kain
Attorney Docket Number:	6851-2-CIP-3

Filed as Small Entity

Utility Filing Fees

Description	Fee Code	Quantity	Amount	Sub-Total in USD($)
Basic Filing:				
Pages:				
Claims:				
Miscellaneous-Filing:				
Petition:				
Patent-Appeals-and-Interference:				
Post-Allowance-and-Post-Issuance:				
Statutory disclaimer	2814	1	65	65
Extension-of-Time:				

Description	Fee Code	Quantity	Amount	Sub-Total in USD($)
Miscellaneous:				
			Total in USD ($)	**65**

Electronic Acknowledgement Receipt

EFS ID:	1094924
Application Number:	10155192
Confirmation Number:	9510
Title of Invention:	Data security system and method with parsing and dispersion techniques
First Named Inventor:	Ron M. Redlich
Customer Number:	4897
Filer:	Robert C. Kain
Filer Authorized By:	
Attorney Docket Number:	6851-2-CIP-3
Receipt Date:	28-JUN-2006
Filing Date:	23-MAY-2002
Time Stamp:	10:22:28
Application Type:	Utility
International Application Number:	

Payment information:

Submitted with Payment	yes
Payment was successfully received in RAM	$65
RAM confirmation Number	708
Deposit Account	031231
The Director of the USPTO is hereby authorized to charge indicated fees and credit any overpayment as follows: Charge any Additional Fees required under 37 C.F.R. Section 1.16 and 1.17	

File Listing:

Document Number	Document Description	File Name	File Size(Bytes)	Multi Part	Pages
1	Amendment After Final	6851-02cip3-amdt-3d-062806-trans.pdf	18347	no	1
Warnings:					
Information:					
2	Applicant Arguments/Remarks Made in an Amendment	6851-02cip3-amdt-3d-062806-rem.pdf	32877	no	3
Warnings:					
Information:					
3	Miscellaneous Incoming Letter	6851-02-cip3-terminal-disclaimer.pdf	78168	no	2
Warnings:					
Information:					
4	Fee Worksheet (PTO-875)	fee-info.pdf	8149	no	2
Warnings:					
Information:					
		Total Files Size (in bytes):	137541		

This Acknowledgement Receipt evidences receipt on the noted date by the USPTO of the indicated documents, characterized by the applicant, and including page counts, where applicable. It serves as evidence of receipt similar to a Post Card, as described in MPEP 503.

New Applications Under 35 U.S.C. 111
If a new application is being filed and the application includes the necessary components for a filing date (see 37 CFR 1.53(b)-(d) and MPEP 506), a Filing Receipt (37 CFR 1.54) will be issued in due course and the date shown on this Acknowledgement Receipt will establish the filing date of the application.

National Stage of an International Application under 35 U.S.C. 371
If a timely submission to enter the national stage of an international application is compliant with the conditions of 35 U.S.C. 371 and other applicable requirements a Form PCT/DO/EO/903 indicating acceptance of the application as a national stage submission under 35 U.S.C. 371 will be issued in addition to the Filing Receipt, in due course.

PATENTS

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re Application of Ron M. Redlich, et al

Serial No. 10/155,192
Confirmation No. 9510 Examiner Chung

Filed: May 23, 2002 Group 2143

For: Data Security System and Method With Parsing and Dispersion Techniques

SUBMISSION OF REPLACEMENT DRAWINGS

Mail Stop PGPUB Drawings
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Sir:

Enclosed please find seventeen (17) sheet(s) of replacement drawings for the captioned case.

Respectfully submitted,

By _____ July 14, 2006

Robert C. Kain, Jr.
Reg. No. 30,648
Fleit, Kain, Gibbons, Gutman, Bongini & Bianco, P.L.
750 Southeast Third Avenue, Suite 100
Ft. Lauderdale, FL 33316-1153
Telephone: 954-768-9002
Facsimile: 954-768-0158

\\Tiger\data share\RCK\CLIENTS\Redlich\6851-02cip3-Submit-Formal-Dwgs-071406.wpd

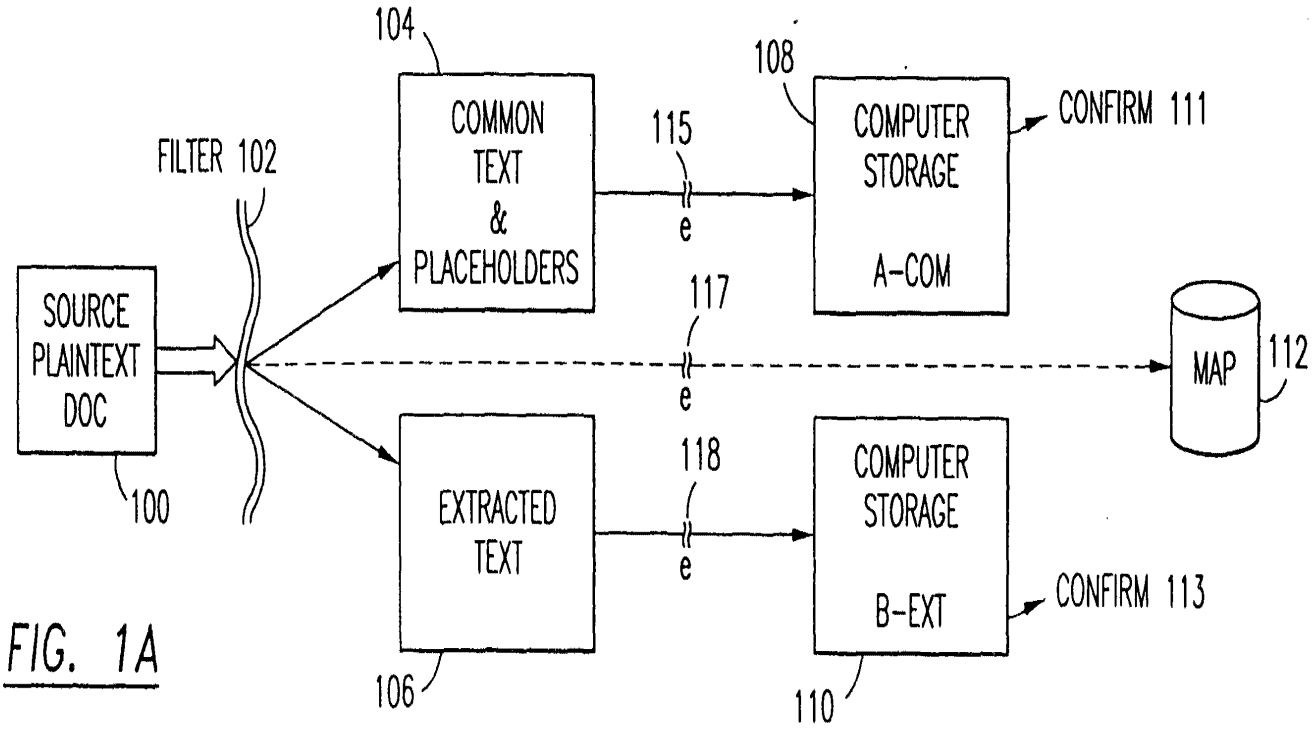


FIG. 1A

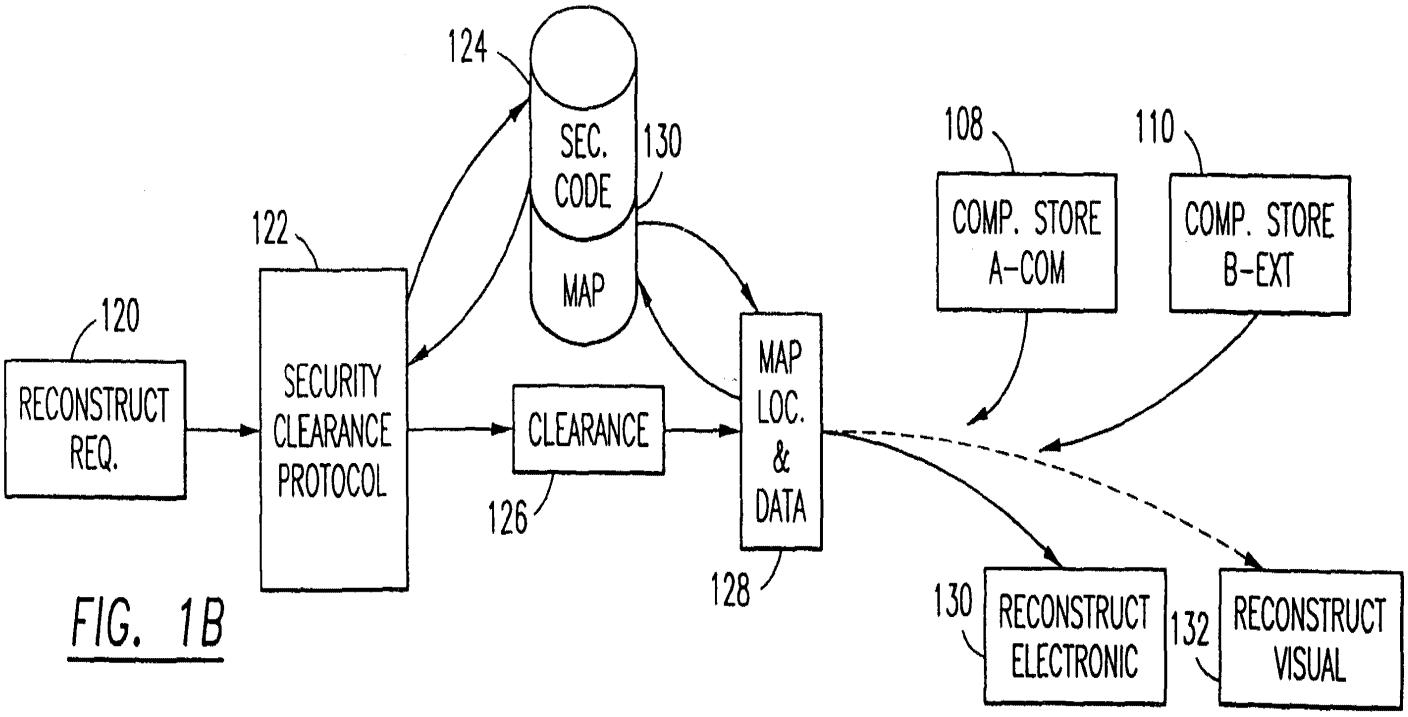


FIG. 1B

Replacement Drawing

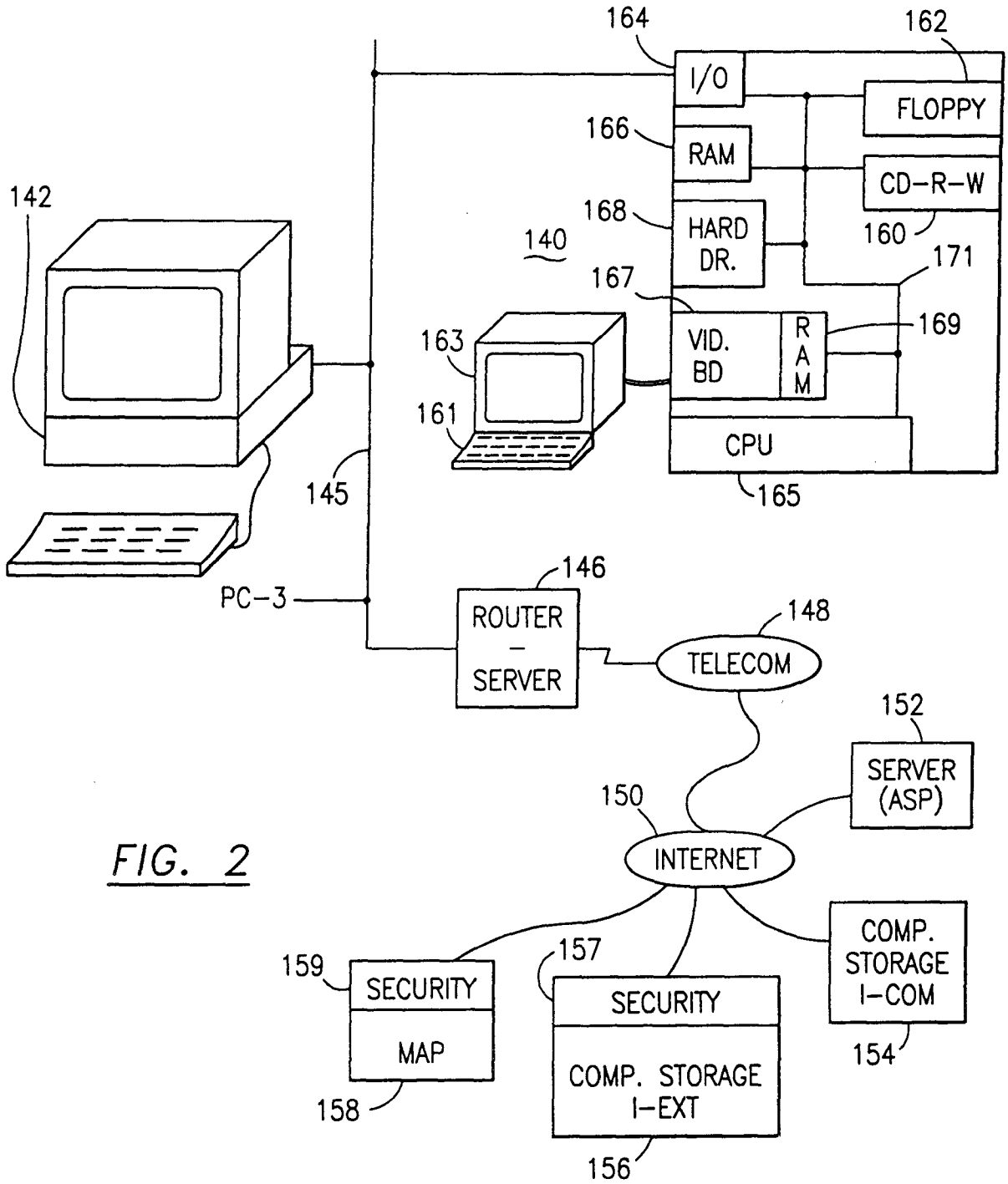
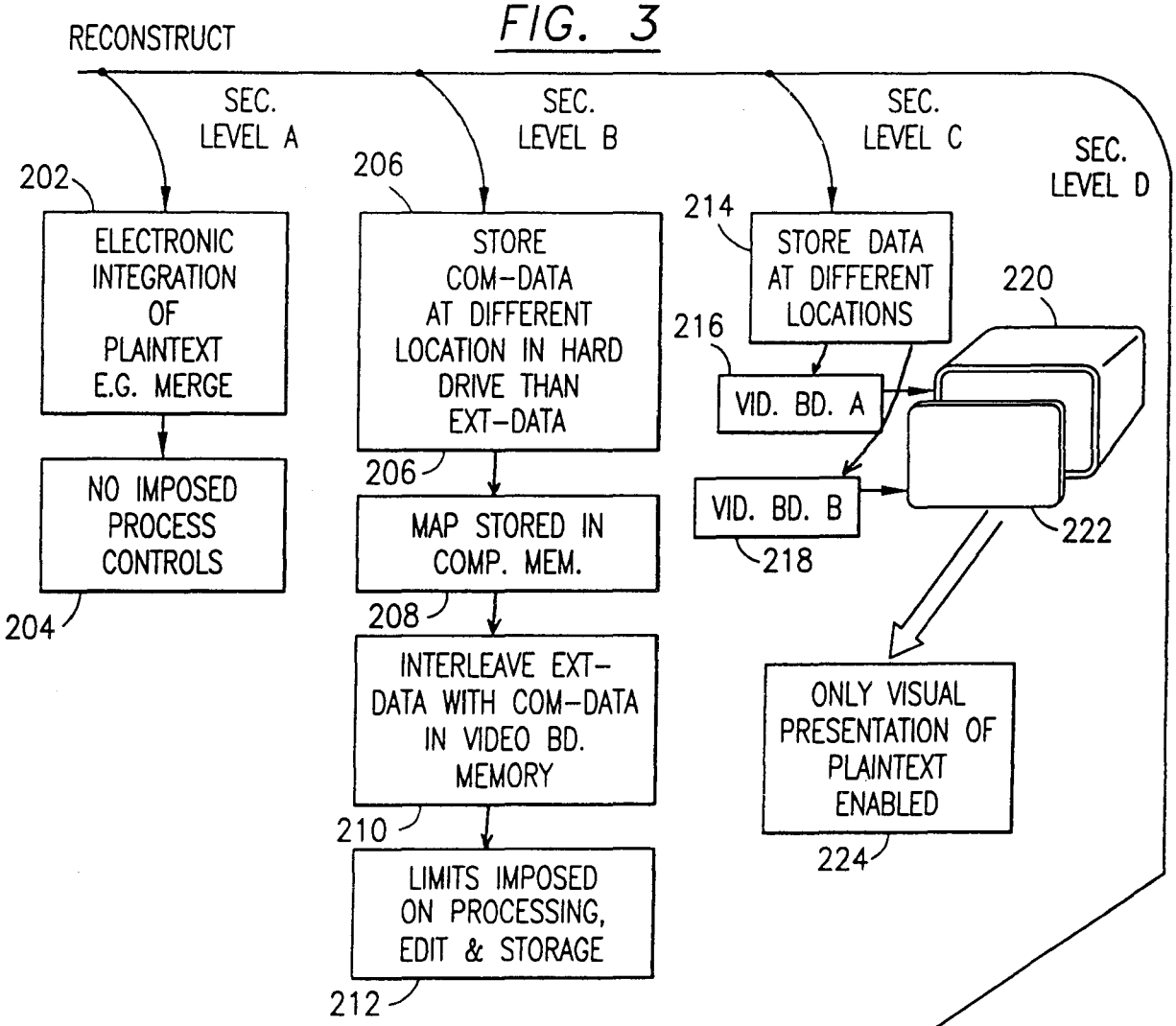

164
162
I/O
FLOPPY
166
RAM
CD-R-W
142
168
HARD DR.
140
160
171
167
VID. BD
R A M
169
163
161
CPU
145
165
146
PC-3
ROUTER – SERVER
148
TELECOM
152
SERVER (ASP)
150
INTERNET
COMP. STORAGE I-COM
159
SECURITY
MAP
158
157
SECURITY
COMP. STORAGE I-EXT
154
156

FIG. 2

FIG. 3


RECONSTRUCT
SEC. LEVEL A
SEC. LEVEL B
SEC. LEVEL C
SEC. LEVEL D
202
ELECTRONIC INTEGRATION OF PLAINTEXT E.G. MERGE
NO IMPOSED PROCESS CONTROLS
204
206
STORE COM-DATA AT DIFFERENT LOCATION IN HARD DRIVE THAN EXT-DATA
206
MAP STORED IN COMP. MEM.
208
INTERLEAVE EXT-DATA WITH COM-DATA IN VIDEO BD. MEMORY
210
LIMITS IMPOSED ON PROCESSING, EDIT & STORAGE
212
214
STORE DATA AT DIFFERENT LOCATIONS
216
VID. BD. A
VID. BD. B
218
220
222
ONLY VISUAL PRESENTATION OF PLAINTEXT ENABLED
224
228
PARSE EXT-DATA PER SECURITY CLEARANCE
226
REPEAT PROCESS A, B OR C WITH ONLY PORTIONS OF EXT-DATA PRESENTED TO USER BASED ON SEC. CLEARANCE

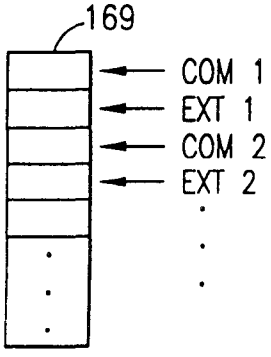

169
COM 1
EXT 1
COM 2
EXT 2

FIG. 3A

FIG. 4

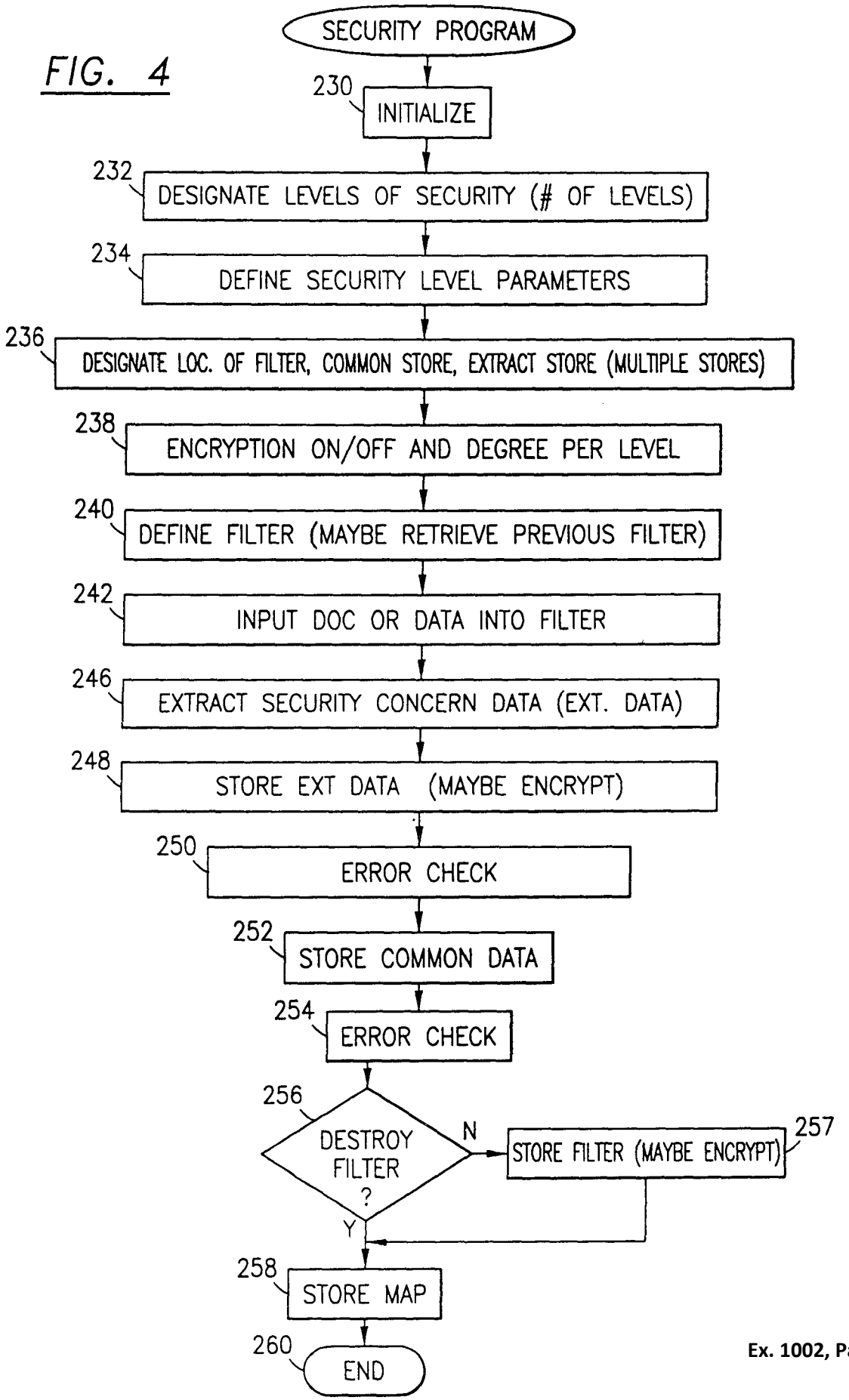
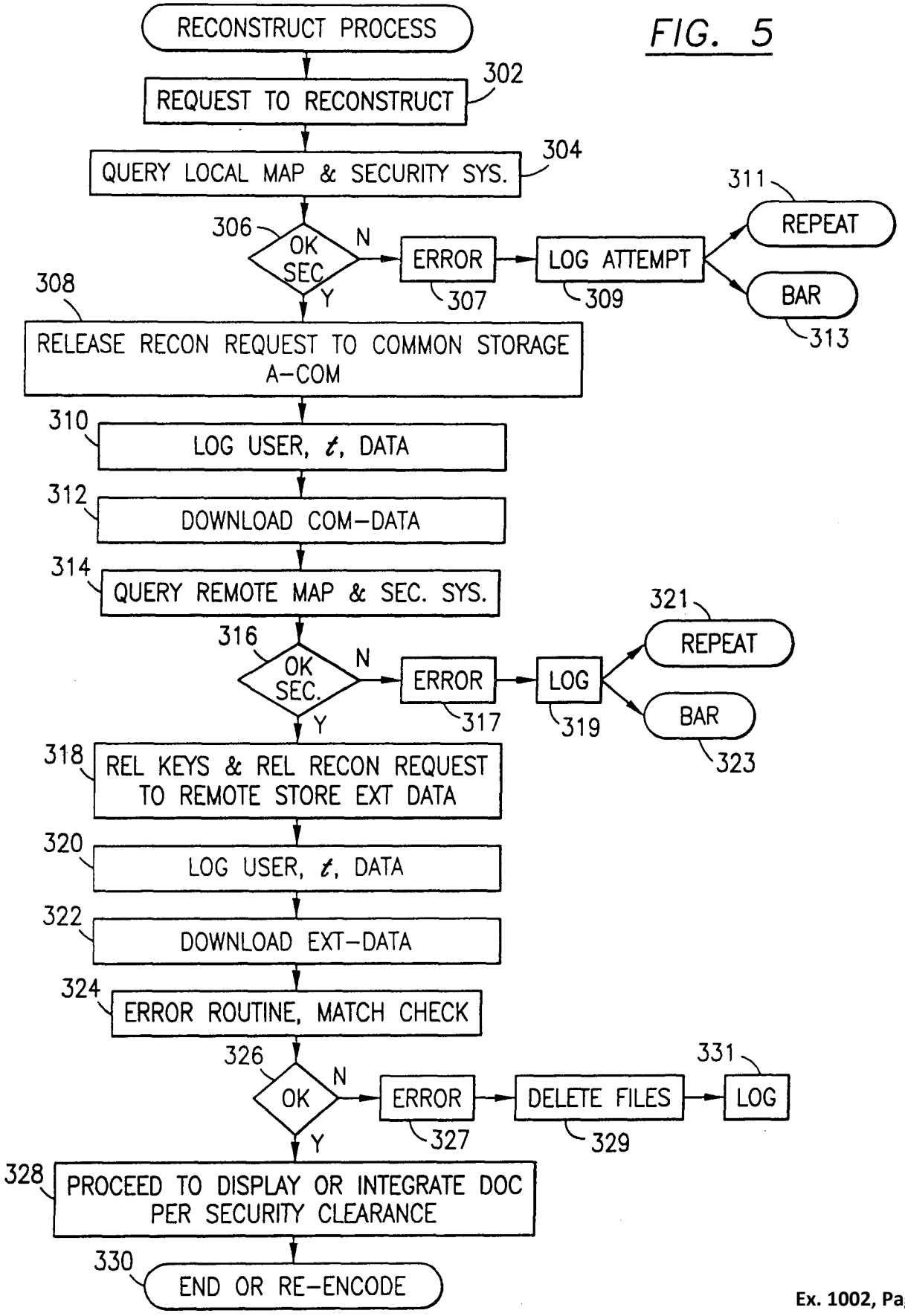

SECURITY PROGRAM
230
INITIALIZE
232
DESIGNATE LEVELS OF SECURITY (# OF LEVELS)
234
DEFINE SECURITY LEVEL PARAMETERS
236
DESIGNATE LOC. OF FILTER, COMMON STORE, EXTRACT STORE (MULTIPLE STORES)
238
ENCRYPTION ON/OFF AND DEGREE PER LEVEL
240
DEFINE FILTER (MAYBE RETRIEVE PREVIOUS FILTER)
242
INPUT DOC OR DATA INTO FILTER
246
EXTRACT SECURITY CONCERN DATA (EXT. DATA)
248
STORE EXT DATA (MAYBE ENCRYPT)
250
ERROR CHECK
252
STORE COMMON DATA
254
ERROR CHECK
256
DESTROY FILTER ?
N
257
STORE FILTER (MAYBE ENCRYPT)
Y
258
STORE MAP
260
END

FIG. 5


RECONSTRUCT PROCESS
302
REQUEST TO RECONSTRUCT
304
QUERY LOCAL MAP & SECURITY SYS.
306
OK SEC
N
Y
ERROR
307
LOG ATTEMPT
309
311
REPEAT
BAR
313
308
RELEASE RECON REQUEST TO COMMON STORAGE A-COM
310
LOG USER, t, DATA
312
DOWNLOAD COM-DATA
314
QUERY REMOTE MAP & SEC. SYS.
316
OK SEC.
N
Y
ERROR
317
LOG
319
321
REPEAT
BAR
323
318
REL KEYS & REL RECON REQUEST TO REMOTE STORE EXT DATA
320
LOG USER, t, DATA
322
DOWNLOAD EXT-DATA
324
ERROR ROUTINE, MATCH CHECK
326
OK
N
Y
ERROR
327
DELETE FILES
329
331
LOG
328
PROCEED TO DISPLAY OR INTEGRATE DOC PER SECURITY CLEARANCE
330
END OR RE-ENCODE

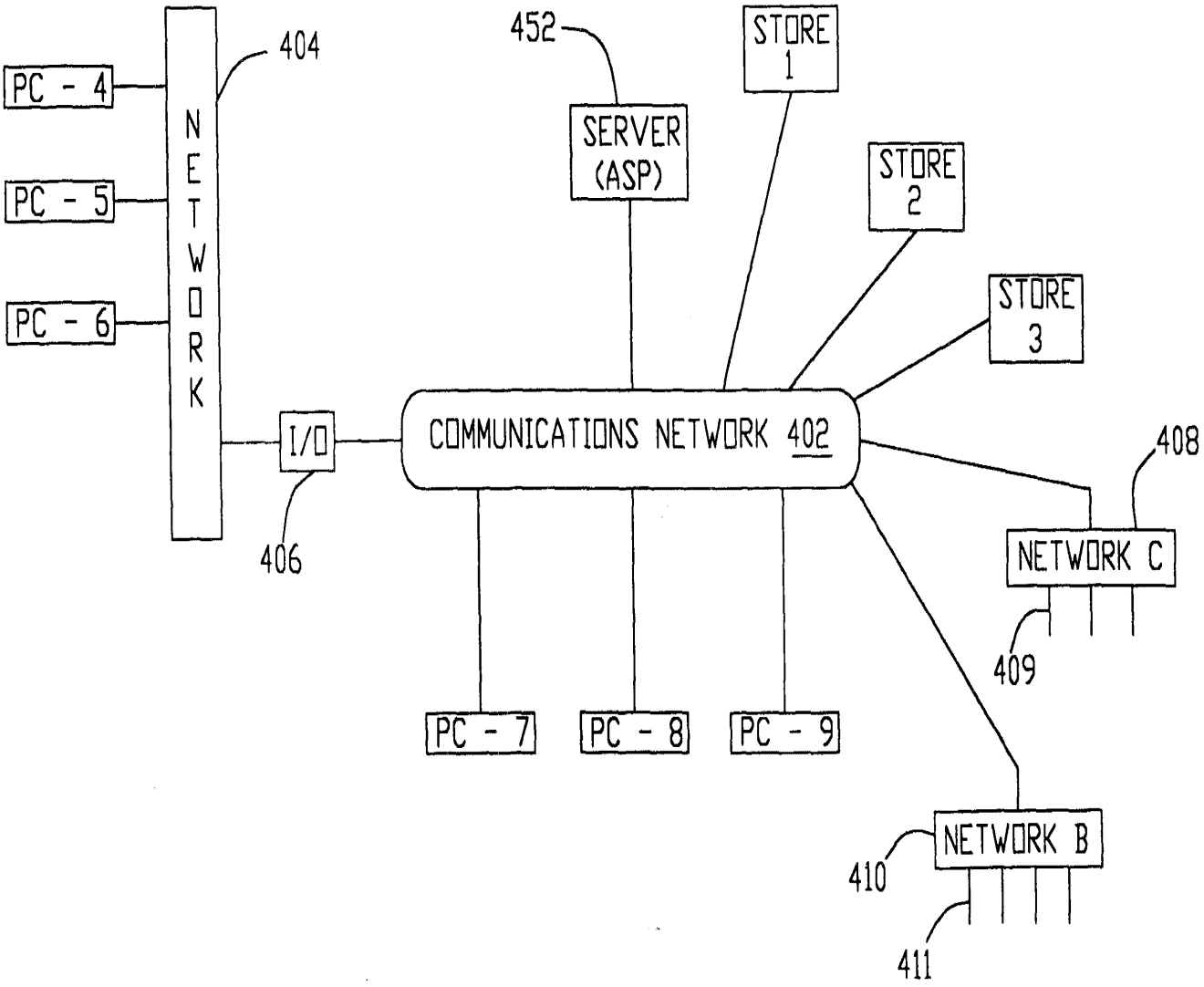


FIG.6

Replacement Drawing

FIG. 7

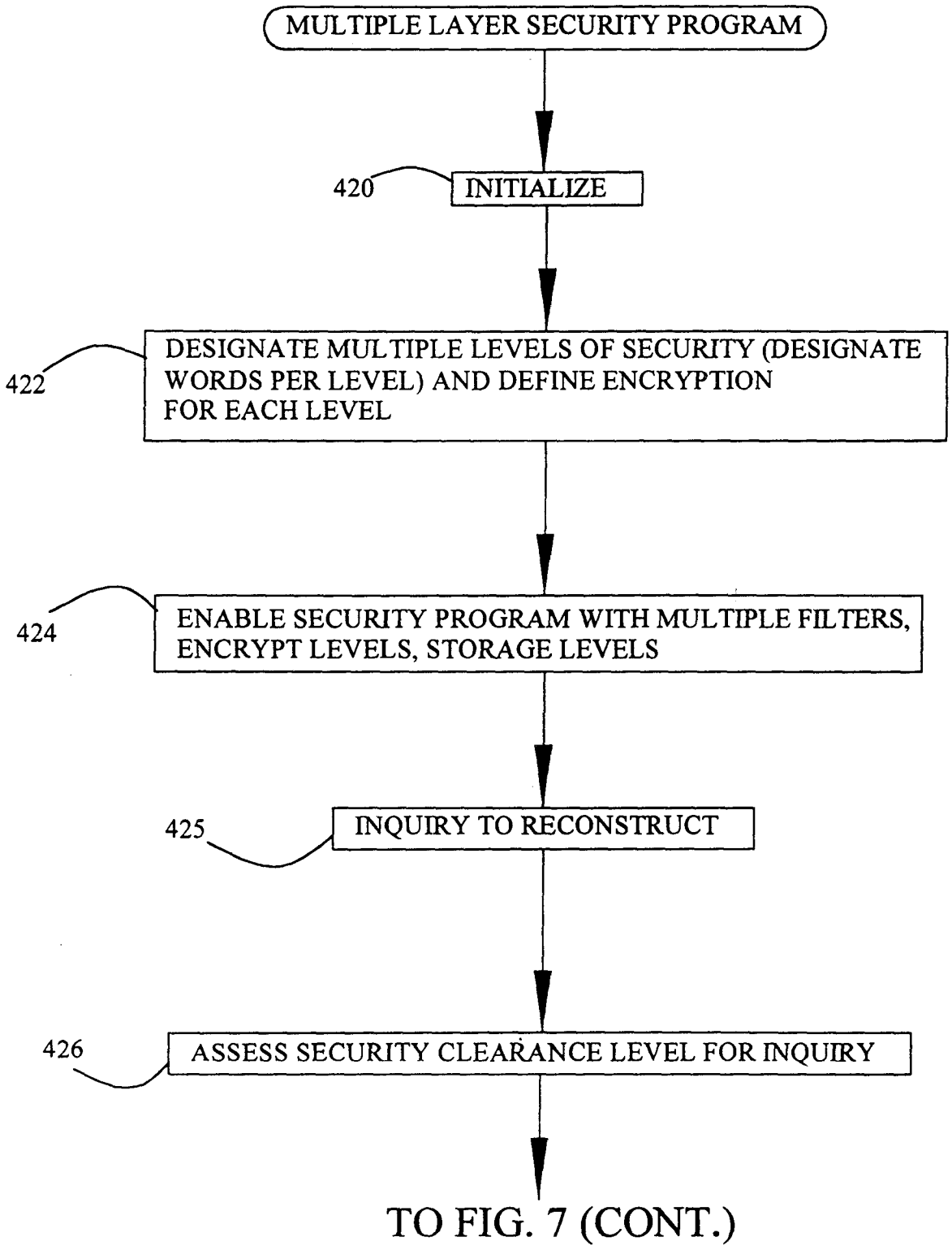

MULTIPLE LAYER SECURITY PROGRAM
420
INITIALIZE
422
DESIGNATE MULTIPLE LEVELS OF SECURITY (DESIGNATE WORDS PER LEVEL) AND DEFINE ENCRYPTION FOR EACH LEVEL
424
ENABLE SECURITY PROGRAM WITH MULTIPLE FILTERS, ENCRYPT LEVELS, STORAGE LEVELS
425
INQUIRY TO RECONSTRUCT
426
ASSESS SECURITY CLEARANCE LEVEL FOR INQUIRY
TO FIG. 7 (CONT.)

FIG. 7 (CONT.)

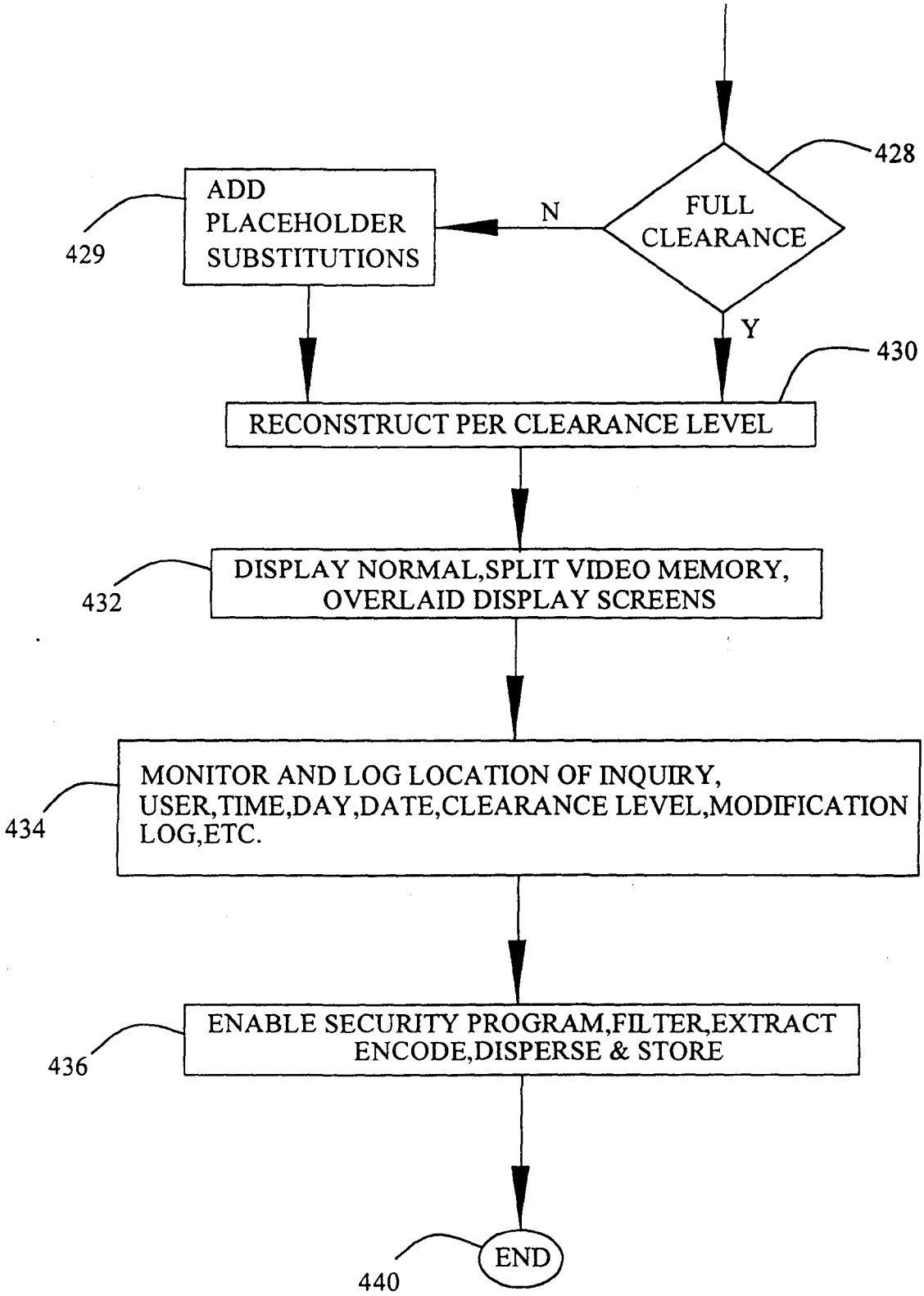

428
FULL CLEARANCE
N
ADD PLACEHOLDER SUBSTITUTIONS
429
Y
430
RECONSTRUCT PER CLEARANCE LEVEL
DISPLAY NORMAL,SPLIT VIDEO MEMORY, OVERLAID DISPLAY SCREENS
432
MONITOR AND LOG LOCATION OF INQUIRY, USER,TIME,DAY,DATE,CLEARANCE LEVEL,MODIFICATION LOG,ETC.
434
ENABLE SECURITY PROGRAM,FILTER,EXTRACT ENCODE,DISPERSE & STORE
436
END
440

FIG. 8

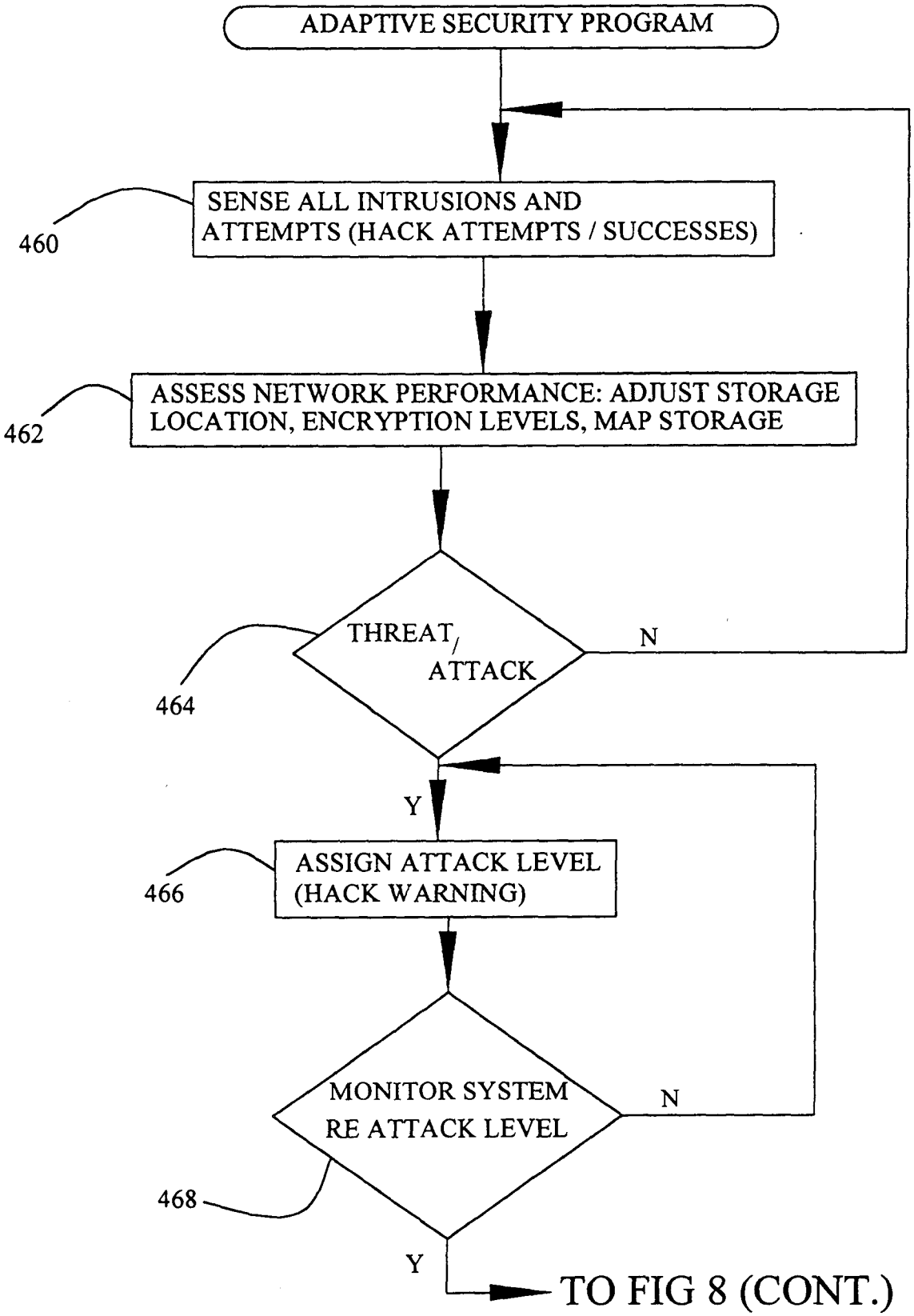

ADAPTIVE SECURITY PROGRAM
SENSE ALL INTRUSIONS AND ATTEMPTS (HACK ATTEMPTS / SUCCESSES)
460
ASSESS NETWORK PERFORMANCE: ADJUST STORAGE LOCATION, ENCRYPTION LEVELS, MAP STORAGE
462
THREAT/ ATTACK
464
N
Y
ASSIGN ATTACK LEVEL (HACK WARNING)
466
MONITOR SYSTEM RE ATTACK LEVEL
468
N
Y
TO FIG 8 (CONT.)

FIG. 8 (CONT.)

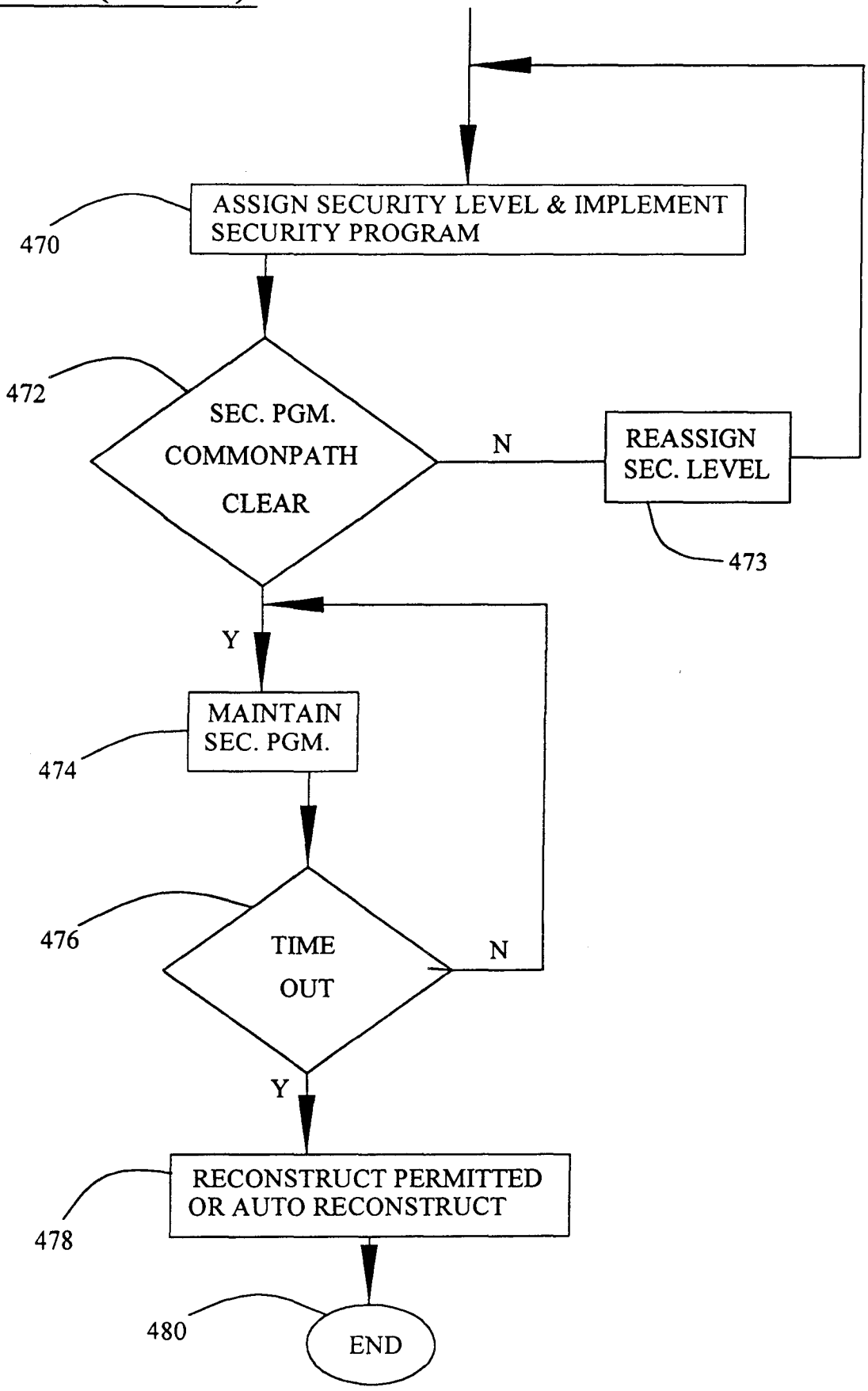

ASSIGN SECURITY LEVEL & IMPLEMENT SECURITY PROGRAM
470
SEC. PGM. COMMONPATH CLEAR
472
N
REASSIGN SEC. LEVEL
473
Y
MAINTAIN SEC. PGM.
474
TIME OUT
476
N
Y
RECONSTRUCT PERMITTED OR AUTO RECONSTRUCT
478
END
480

FIG. 9

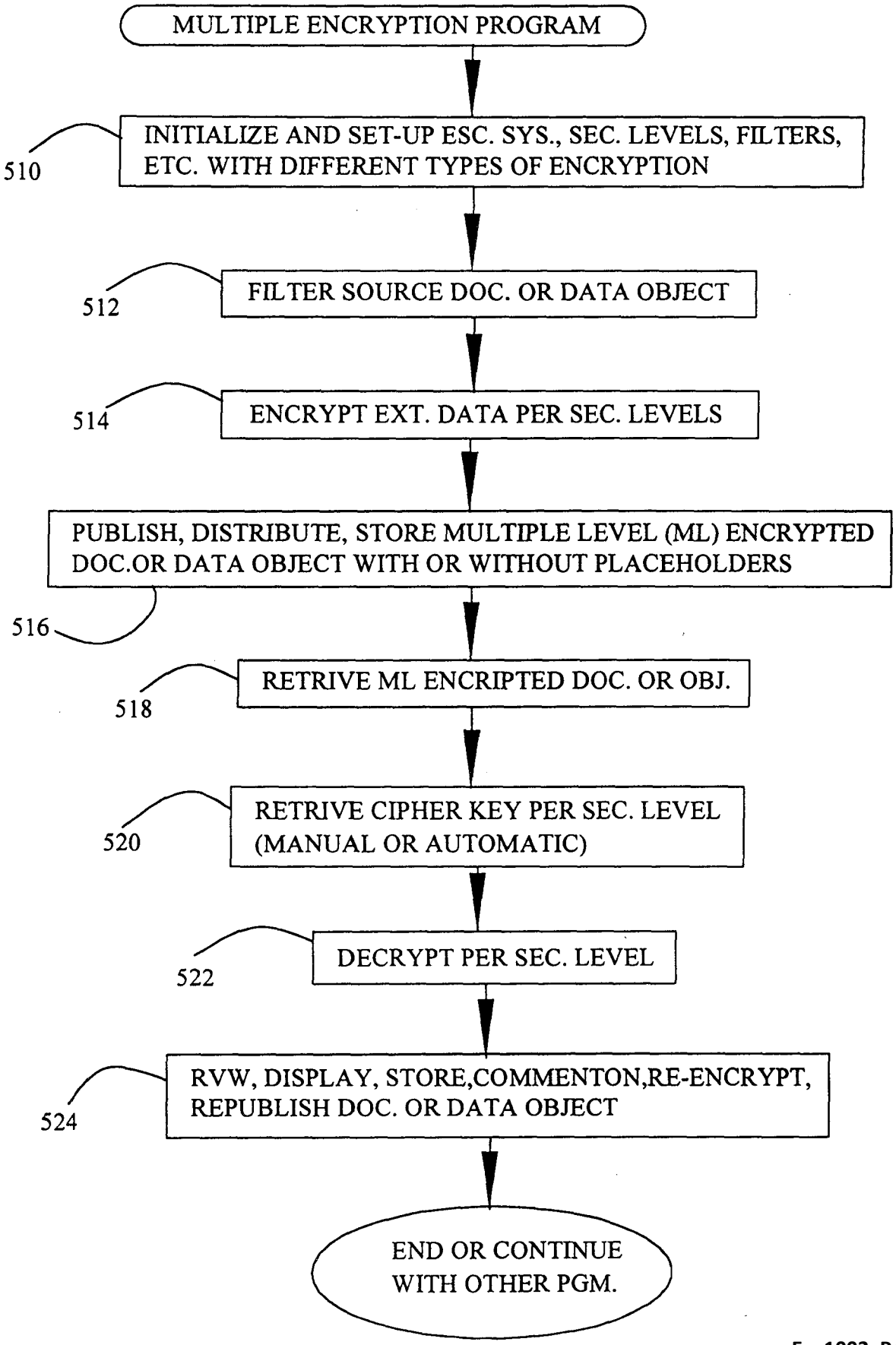

MULTIPLE ENCRYPTION PROGRAM
510
INITIALIZE AND SET-UP ESC. SYS., SEC. LEVELS, FILTERS, ETC. WITH DIFFERENT TYPES OF ENCRYPTION
512
FILTER SOURCE DOC. OR DATA OBJECT
514
ENCRYPT EXT. DATA PER SEC. LEVELS
516
PUBLISH, DISTRIBUTE, STORE MULTIPLE LEVEL (ML) ENCRYPTED DOC.OR DATA OBJECT WITH OR WITHOUT PLACEHOLDERS
518
RETRIVE ML ENCRIPTED DOC. OR OBJ.
520
RETRIVE CIPHER KEY PER SEC. LEVEL (MANUAL OR AUTOMATIC)
522
DECRYPT PER SEC. LEVEL
524
RVW, DISPLAY, STORE,COMMENTON,RE-ENCRYPT, REPUBLISH DOC. OR DATA OBJECT
END OR CONTINUE WITH OTHER PGM.

FIG. 10

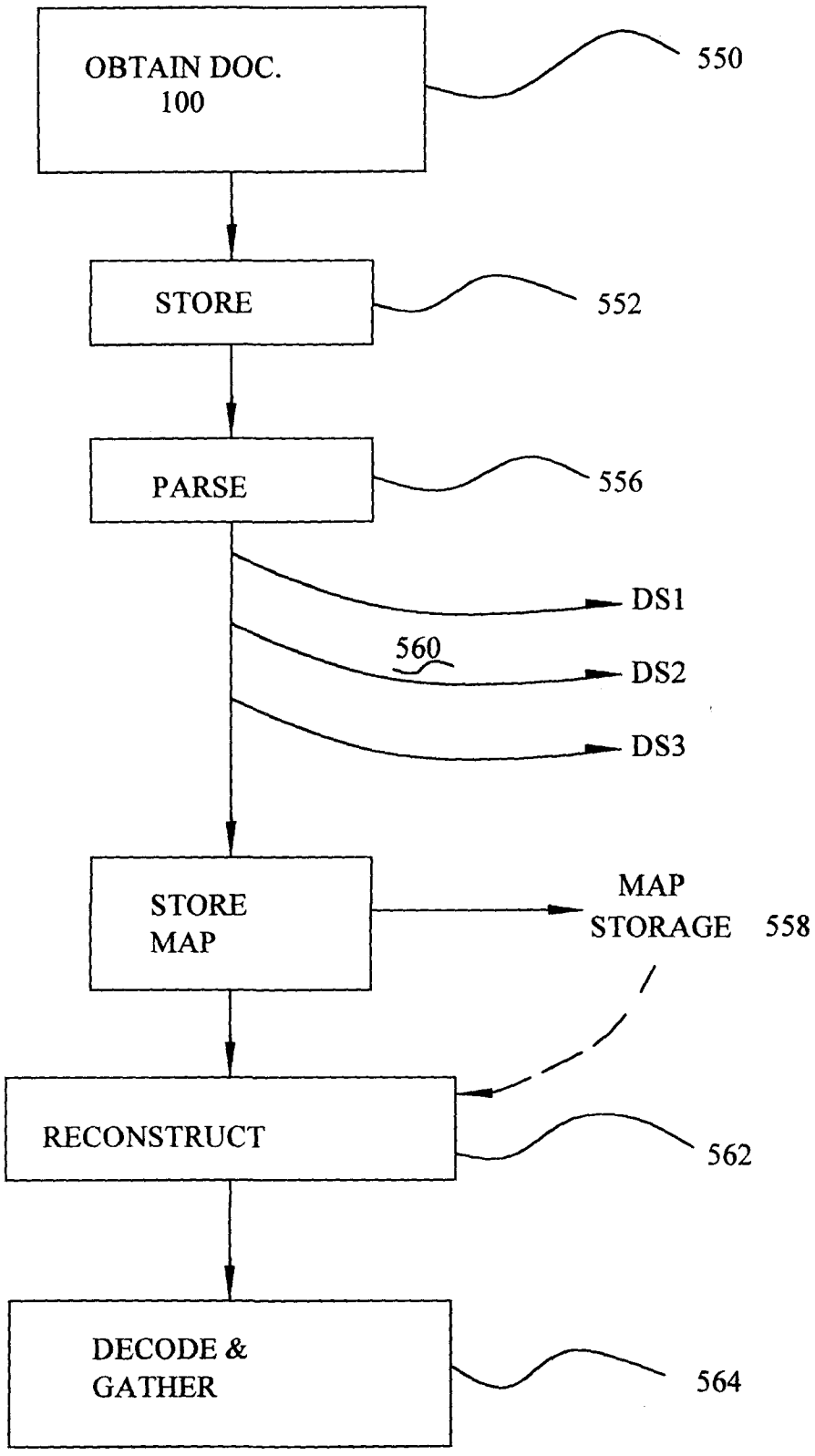

OBTAIN DOC.
100
550
STORE
552
PARSE
556
DS1
560
DS2
DS3
STORE
MAP
MAP
STORAGE
558
RECONSTRUCT
562
DECODE &
GATHER
564

FIG.11A

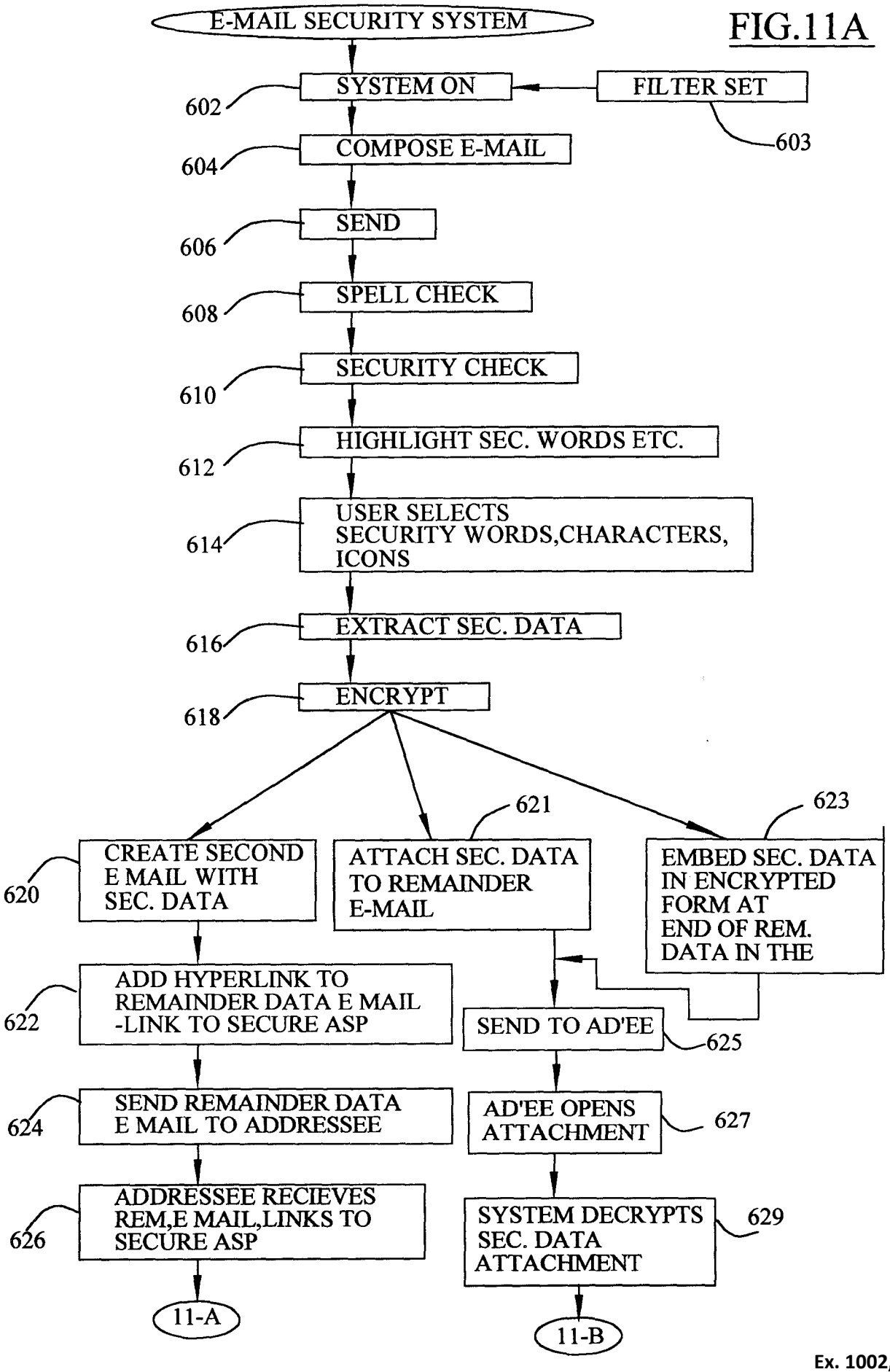

E-MAIL SECURITY SYSTEM
SYSTEM ON
FILTER SET
602
603
COMPOSE E-MAIL
604
SEND
606
SPELL CHECK
608
SECURITY CHECK
610
HIGHLIGHT SEC. WORDS ETC.
612
USER SELECTS SECURITY WORDS,CHARACTERS, ICONS
614
EXTRACT SEC. DATA
616
ENCRYPT
618
621
623
CREATE SECOND E MAIL WITH SEC. DATA
620
ATTACH SEC. DATA TO REMAINDER E-MAIL
EMBED SEC. DATA IN ENCRYPTED FORM AT END OF REM. DATA IN THE
ADD HYPERLINK TO REMAINDER DATA E MAIL -LINK TO SECURE ASP
622
SEND TO AD'EE
625
SEND REMAINDER DATA E MAIL TO ADDRESSEE
624
AD'EE OPENS ATTACHMENT
627
ADDRESSEE RECIEVES REM,E MAIL,LINKS TO SECURE ASP
626
SYSTEM DECRYPTS SEC. DATA ATTACHMENT
629
11-A
11-B

FIG.11B

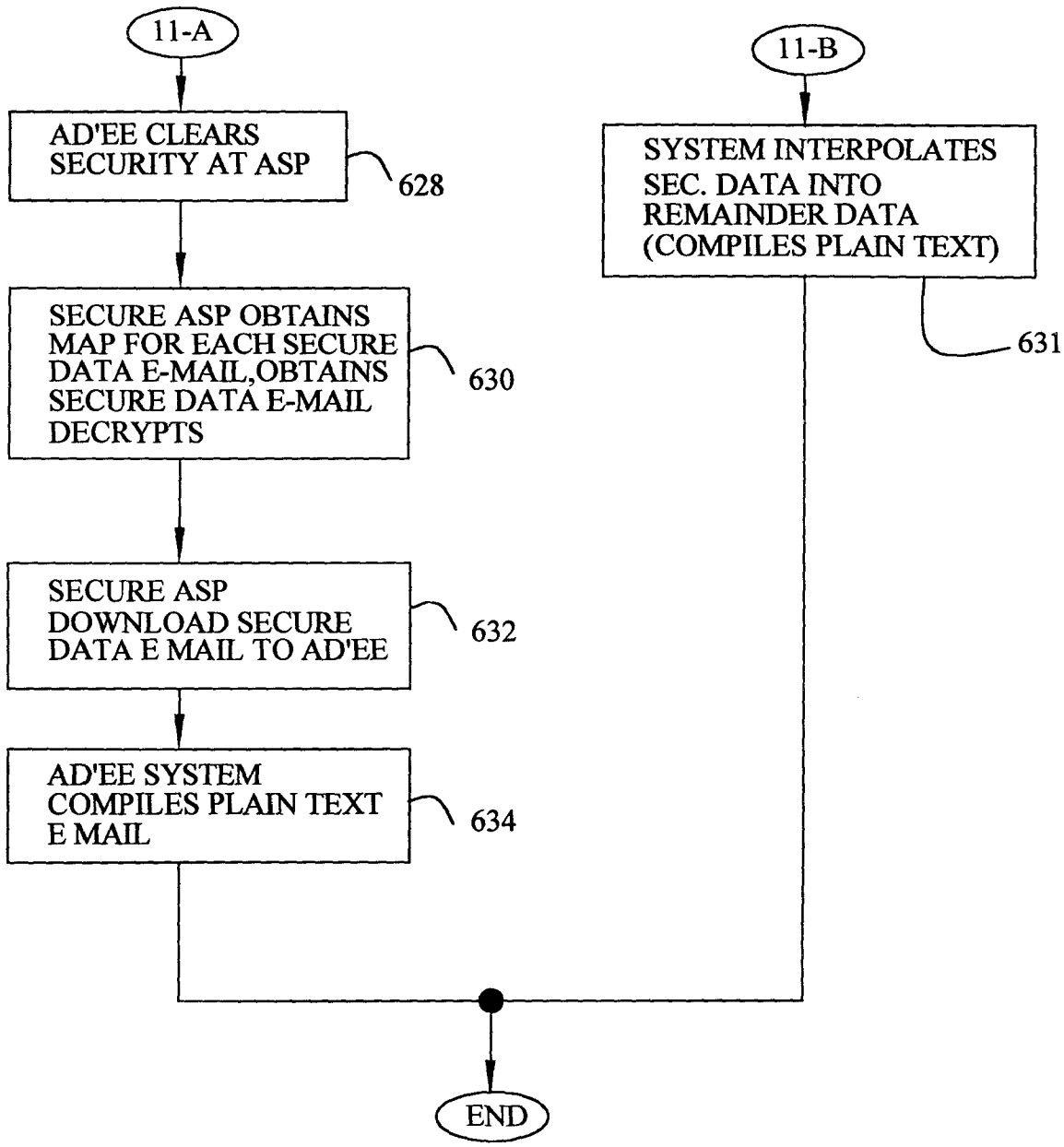

11-A
AD'EE CLEARS SECURITY AT ASP
628
SECURE ASP OBTAINS MAP FOR EACH SECURE DATA E-MAIL,OBTAINS SECURE DATA E-MAIL DECRYPTS
630
SECURE ASP DOWNLOAD SECURE DATA E MAIL TO AD'EE
632
AD'EE SYSTEM COMPILES PLAIN TEXT E MAIL
634
11-B
SYSTEM INTERPOLATES SEC. DATA INTO REMAINDER DATA (COMPILES PLAIN TEXT)
631
END

FIG.12A

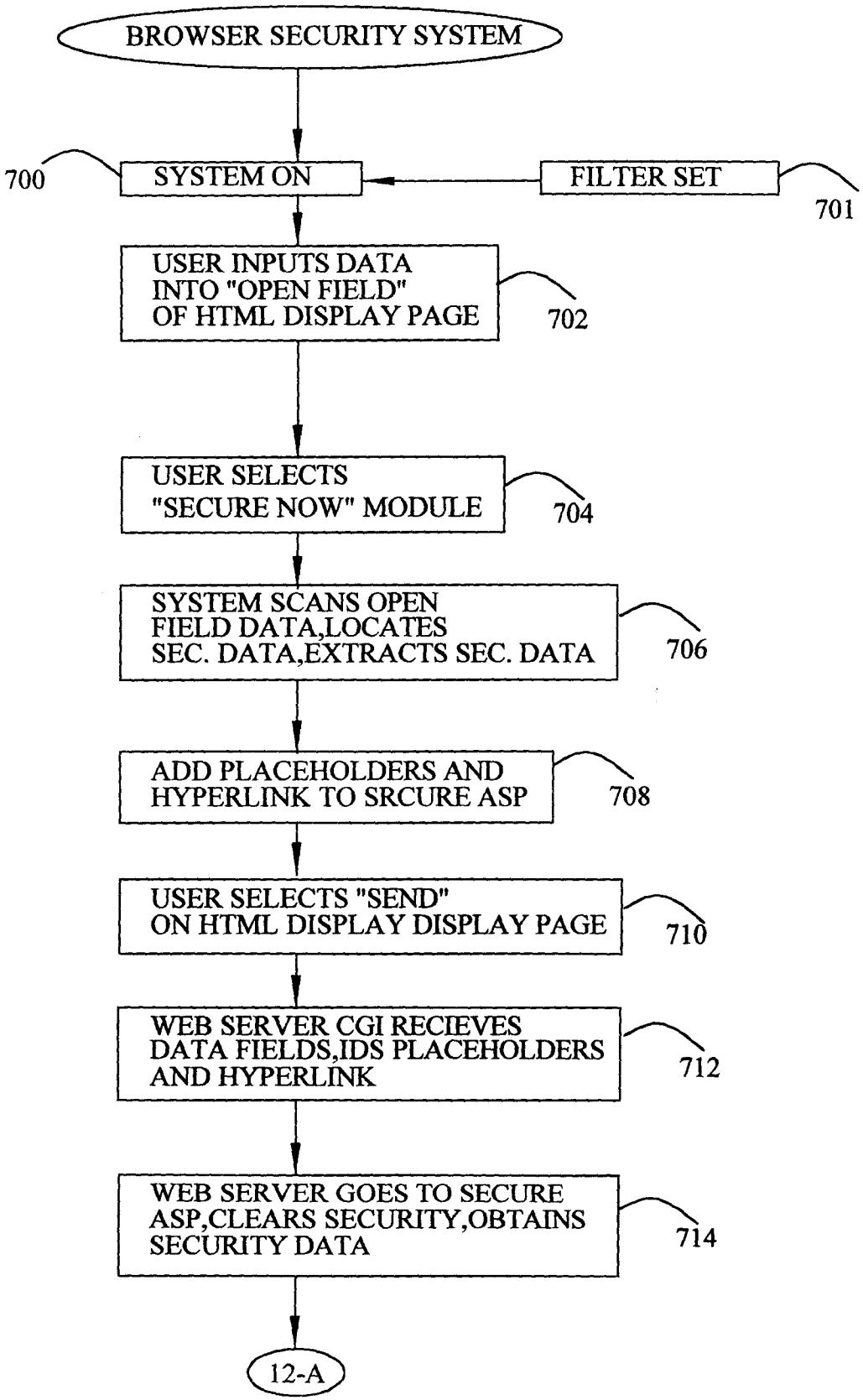

BROWSER SECURITY SYSTEM
700
SYSTEM ON
FILTER SET
701
USER INPUTS DATA INTO "OPEN FIELD" OF HTML DISPLAY PAGE
702
USER SELECTS "SECURE NOW" MODULE
704
SYSTEM SCANS OPEN FIELD DATA,LOCATES SEC. DATA,EXTRACTS SEC. DATA
706
ADD PLACEHOLDERS AND HYPERLINK TO SRCURE ASP
708
USER SELECTS "SEND" ON HTML DISPLAY DISPLAY PAGE
710
WEB SERVER CGI RECIEVES DATA FIELDS,IDS PLACEHOLDERS AND HYPERLINK
712
WEB SERVER GOES TO SECURE ASP,CLEARS SECURITY,OBTAINS SECURITY DATA
714
12-A

FIG.12B

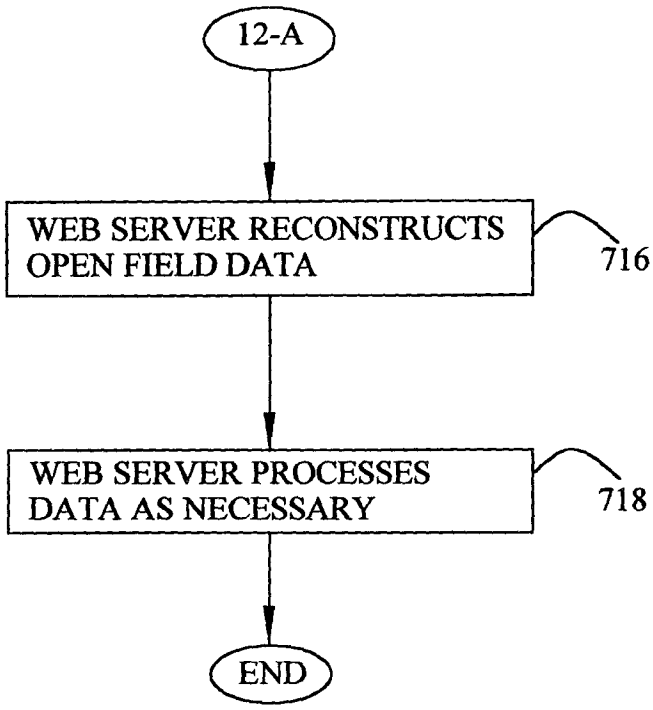
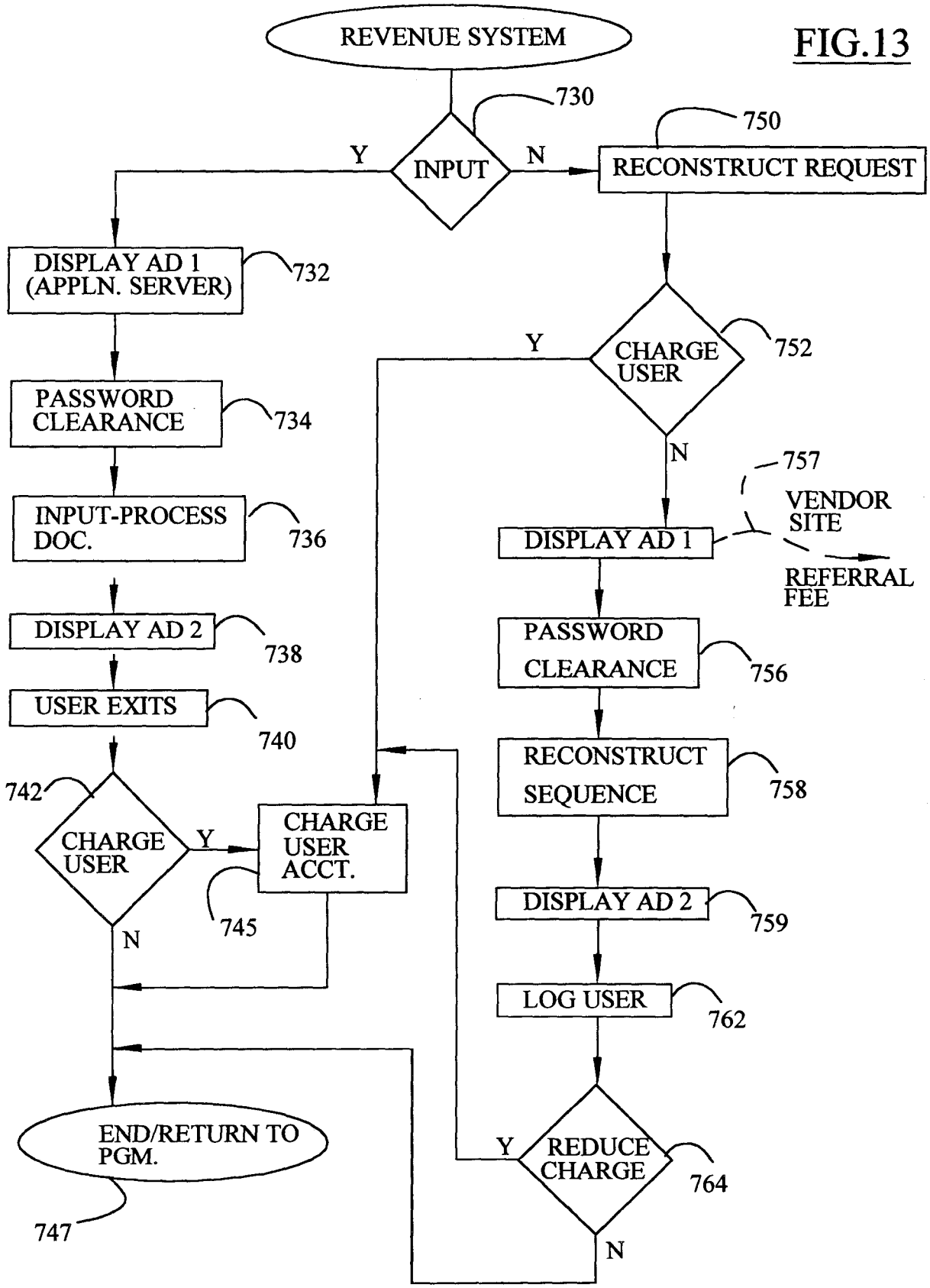

12-A
WEB SERVER RECONSTRUCTS OPEN FIELD DATA
716
WEB SERVER PROCESSES DATA AS NECESSARY
718
END

FIG.13


REVENUE SYSTEM
730
INPUT
Y
N
750
RECONSTRUCT REQUEST
DISPLAY AD 1 (APPLN. SERVER)
732
PASSWORD CLEARANCE
734
INPUT-PROCESS DOC.
736
DISPLAY AD 2
738
USER EXITS
740
742
CHARGE USER
Y
CHARGE USER ACCT.
745
N
END/RETURN TO PGM.
747
Y
CHARGE USER
752
N
757
VENDOR SITE
DISPLAY AD 1
REFERRAL FEE
PASSWORD CLEARANCE
756
RECONSTRUCT SEQUENCE
758
DISPLAY AD 2
759
LOG USER
762
Y
REDUCE CHARGE
764
N

Electronic Acknowledgement Receipt

EFS ID:	1113917
Application Number:	10155192
Confirmation Number:	9510
Title of Invention:	Data security system and method with parsing and dispersion techniques
First Named Inventor:	Ron M. Redlich
Customer Number:	4897
Filer:	Robert C. Kain
Filer Authorized By:	
Attorney Docket Number:	6851-2-CIP-3
Receipt Date:	14-JUL-2006
Filing Date:	23-MAY-2002
Time Stamp:	16:05:34
Application Type:	Utility
International Application Number:	

Payment information:

Submitted with Payment	no

File Listing:

Document Number	Document Description	File Name	File Size(Bytes)	Multi Part	Pages
1	Miscellaneous Incoming Letter	6851-02cip3-Submit-Formal-Dwgs-071406.pdf	22501	no	1

Warnings:					
Information:					
2	Drawings	6851-02cip3-drawings-071406.pdf	350764	no	17
Warnings:					
The page size in the PDF is too large. The pages should be 8.5 x 11 or A4. If this PDF is submitted, the pages will be resized upon entry into the Image File Wrapper and may affect subsequent processing					
Information:					
			Total Files Size (in bytes):	373265	

This Acknowledgement Receipt evidences receipt on the noted date by the USPTO of the indicated documents, characterized by the applicant, and including page counts, where applicable. It serves as evidence of receipt similar to a Post Card, as described in MPEP 503.

New Applications Under 35 U.S.C. 111

If a new application is being filed and the application includes the necessary components for a filing date (see 37 CFR 1.53(b)-(d) and MPEP 506), a Filing Receipt (37 CFR 1.54) will be issued in due course and the date shown on this Acknowledgement Receipt will establish the filing date of the application.

National Stage of an International Application under 35 U.S.C. 371

If a timely submission to enter the national stage of an international application is compliant with the conditions of 35 U.S.C. 371 and other applicable requirements a Form PCT/DO/EO/903 indicating acceptance of the application as a national stage submission under 35 U.S.C. 371 will be issued in addition to the Filing Receipt, in due course.

Application Number	Application/Control No.	Applicant(s)/Patent under Reexamination	
	10/155,192	REDLICH ET AL.	

Document Code - DISQ	Internal Document – DO NOT MAIL

TERMINAL DISCLAIMER	☐ APPROVED	☒ DISAPPROVED
Date Filed : 06/28/2006	This patent is subject to a Terminal Disclaimer	

Approved/Disapproved by:
dah

U.S. Patent and Trademark Office

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.
10/155,192	05/23/2002	Ron M. Redlich	6851-2-CIP-3	9510

4897 7590 07/31/2006

ROBERT C. KAIN, JR.
750 SOUTHEAST THIRD AVENUE
SUITE 100
FT LAUDERDALE, FL 333161153

EXAMINER	
DALENCOURT, YVES	

ART UNIT	PAPER NUMBER
2157	

DATE MAILED: 07/31/2006

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

Advisory Action Before the Filing of an Appeal Brief	**Application No.** 10/155,192	**Applicant(s)** REDLICH ET AL.	
	Examiner Yves Dalencourt	**Art Unit** 2157	

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

THE REPLY FILED 14 July 2006 FAILS TO PLACE THIS APPLICATION IN CONDITION FOR ALLOWANCE.

1. ☒ The reply was filed after a final rejection, but prior to or on the same day as filing a Notice of Appeal. To avoid abandonment of this application, applicant must timely file one of the following replies: (1) an amendment, affidavit, or other evidence, which places the application in condition for allowance; (2) a Notice of Appeal (with appeal fee) in compliance with 37 CFR 41.31; or (3) a Request for Continued Examination (RCE) in compliance with 37 CFR 1.114. The reply must be filed within one of the following time periods:
   a) ☒ The period for reply expires 3 months from the mailing date of the final rejection.
   b) ☐ The period for reply expires on: (1) the mailing date of this Advisory Action, or (2) the date set forth in the final rejection, whichever is later. In no event, however, will the statutory period for reply expire later than SIX MONTHS from the mailing date of the final rejection.

   Examiner Note: If box 1 is checked, check either box (a) or (b). ONLY CHECK BOX (b) WHEN THE FIRST REPLY WAS FILED WITHIN TWO MONTHS OF THE FINAL REJECTION. See MPEP 706.07(f).

Extensions of time may be obtained under 37 CFR 1.136(a). The date on which the petition under 37 CFR 1.136(a) and the appropriate extension fee have been filed is the date for purposes of determining the period of extension and the corresponding amount of the fee. The appropriate extension fee under 37 CFR 1.17(a) is calculated from: (1) the expiration date of the shortened statutory period for reply originally set in the final Office action; or (2) as set forth in (b) above, if checked. Any reply received by the Office later than three months after the mailing date of the final rejection, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

NOTICE OF APPEAL

2. ☐ The Notice of Appeal was filed on _____. A brief in compliance with 37 CFR 41.37 must be filed within two months of the date of filing the Notice of Appeal (37 CFR 41.37(a)), or any extension thereof (37 CFR 41.37(e)), to avoid dismissal of the appeal. Since a Notice of Appeal has been filed, any reply must be filed within the time period set forth in 37 CFR 41.37(a).

AMENDMENTS

3. ☒ The proposed amendment(s) filed after a final rejection, but prior to the date of filing a brief, will not be entered because
   (a) ☒ They raise new issues that would require further consideration and/or search (see NOTE below);
   (b) ☐ They raise the issue of new matter (see NOTE below);
   (c) ☐ They are not deemed to place the application in better form for appeal by materially reducing or simplifying the issues for appeal; and/or
   (d) ☐ They present additional claims without canceling a corresponding number of finally rejected claims.

   NOTE: *See Continuation Sheet*. (See 37 CFR 1.116 and 41.33(a)).
4. ☐ The amendments are not in compliance with 37 CFR 1.121. See attached Notice of Non-Compliant Amendment (PTOL-324).
5. ☐ Applicant's reply has overcome the following rejection(s): _____.
6. ☐ Newly proposed or amended claim(s) _____ would be allowable if submitted in a separate, timely filed amendment canceling the non-allowable claim(s).
7. ☒ For purposes of appeal, the proposed amendment(s): a) ☒ will not be entered, or b) ☐ will be entered and an explanation of how the new or amended claims would be rejected is provided below or appended.
   The status of the claim(s) is (or will be) as follows:
   Claim(s) allowed: *none*.
   Claim(s) objected to: *none*.
   Claim(s) rejected: *1-27*.
   Claim(s) withdrawn from consideration: _____.

AFFIDAVIT OR OTHER EVIDENCE

8. ☐ The affidavit or other evidence filed after a final action, but before or on the date of filing a Notice of Appeal will not be entered because applicant failed to provide a showing of good and sufficient reasons why the affidavit or other evidence is necessary and was not earlier presented. See 37 CFR 1.116(e).
9. ☐ The affidavit or other evidence filed after the date of filing a Notice of Appeal, but prior to the date of filing a brief, will not be entered because the affidavit or other evidence failed to overcome all rejections under appeal and/or appellant fails to provide a showing a good and sufficient reasons why it is necessary and was not earlier presented. See 37 CFR 41.33(d)(1).
10. ☐ The affidavit or other evidence is entered. An explanation of the status of the claims after entry is below or attached.

REQUEST FOR RECONSIDERATION/OTHER

11. ☐ The request for reconsideration has been considered but does NOT place the application in condition for allowance because: _____.
12. ☐ Note the attached Information Disclosure Statement(s). (PTO/SB/08 or PTO-1449) Paper No(s). _____
13. ☐ Other: _____.

Dalencourt
Primary Examiner

Continuation of 3. NOTE: the added limitations of "physically distinguishable"..... and "physically distinct" ..., and "permitting full or partial .... " in claims 1, 7, 16, and 22 require further search and consideration .

PATENTS

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re Application of Ron M. Redlich, et al.

Serial No. 10/155,192 Examiner Dalencourt
Confirm no. 9510
Filed: May 23, 2002 Group 2157

For: Data Security System and Method With Parsing and Dispersion Techniques

REQUEST FOR CONTINUED EXAMINATION (RCE)

Commissioner for Patents
Washington, D.C. 20231

Sir:

This is a request for continued examination (RCE) under 37 C.F.R. § 1.114 for the captioned patent application. On June 20, 2006, June 28 and July 14, Applicant submitted papers responding to the outstanding Office Action. Rule 1.114 indicates that an RCE and a previous submission will be considered after the RCE is filed. 37 C.F.R. § 1.114(d). Applicant requests that the final office action be withdrawn and the claims approved. Please debit Deposit Account No. 03-1231 for the $375.00 fee. Enclosed are:

1. Request for Continued Examination (RCE) (1 pg.)

Fleit, Kain, Gibbons, Gutman, Bongini & Bianco, P.L.

By___/Robert Kain/_____
Robert C. Kain, Jr.
Reg. No. 30,648
750 Southeast Third Avenue, Suite 100
Ft. Lauderdale, Florida 33316-1153
Telephone: (954) 768-9002
Facsimile: (954)768-0158

Certificate of eFiling

I hereby certify that this correspondence is being filed via efiling with the Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450 on August 7, 2006.

___/Robert Kain/_____
Robert C. Kain, Jr., Reg. No. 30,648

\\TIGER\Data Share\RCK\CLIENTS\Redlich\6851-02cip3-RCE-080706.wpd

Electronic Patent Application Fee Transmittal

Application Number:	10155192
Filing Date:	23-May-2002
Title of Invention:	Data security system and method with parsing and dispersion techniques
First Named Inventor:	Ron M. Redlich
Filer:	Robert C. Kain
Attorney Docket Number:	6851-2-CIP-3

Filed as Small Entity

Utility Filing Fees

Description	Fee Code	Quantity	Amount	Sub-Total in USD($)
Basic Filing:				
Pages:				
Claims:				
Miscellaneous-Filing:				
Petition:				
Patent-Appeals-and-Interference:				
Post-Allowance-and-Post-Issuance:				
Extension-of-Time:				

Description	Fee Code	Quantity	Amount	Sub-Total in USD($)
Miscellaneous:				
Request for continued examination	2801	1	395	395
		Total in USD ($)		**395**

Electronic Acknowledgement Receipt

EFS ID:	1143455
Application Number:	10155192
Confirmation Number:	9510
Title of Invention:	Data security system and method with parsing and dispersion techniques
First Named Inventor:	Ron M. Redlich
Customer Number:	4897
Filer:	Robert C. Kain
Filer Authorized By:	
Attorney Docket Number:	6851-2-CIP-3
Receipt Date:	07-AUG-2006
Filing Date:	23-MAY-2002
Time Stamp:	14:34:16
Application Type:	Utility
International Application Number:	

Payment information:

Submitted with Payment	yes
Payment was successfully received in RAM	$395
RAM confirmation Number	1017
Deposit Account	031231
The Director of the USPTO is hereby authorized to charge indicated fees and credit any overpayment as follows: Charge any Additional Fees required under 37 C.F.R. Section 1.16 and 1.17	

File Listing:

Document Number	Document Description	File Name	File Size(Bytes)	Multi Part	Pages
1	Request for Continued Examination (RCE)	6851-02cip3-RCE-080706.pdf	22725	no	1
Warnings:					
Information:					
2	Fee Worksheet (PTO-875)	fee-info.pdf	8172	no	2
Warnings:					
Information:					
		Total Files Size (in bytes):	30897		

This Acknowledgement Receipt evidences receipt on the noted date by the USPTO of the indicated documents, characterized by the applicant, and including page counts, where applicable. It serves as evidence of receipt similar to a Post Card, as described in MPEP 503.

New Applications Under 35 U.S.C. 111
If a new application is being filed and the application includes the necessary components for a filing date (see 37 CFR 1.53(b)-(d) and MPEP 506), a Filing Receipt (37 CFR 1.54) will be issued in due course and the date shown on this Acknowledgement Receipt will establish the filing date of the application.

National Stage of an International Application under 35 U.S.C. 371
If a timely submission to enter the national stage of an international application is compliant with the conditions of 35 U.S.C. 371 and other applicable requirements a Form PCT/DO/EO/903 indicating acceptance of the application as a national stage submission under 35 U.S.C. 371 will be issued in addition to the Filing Receipt, in due course.

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.
10/155,192	05/23/2002	Ron M. Redlich	6851-2-CIP-3	9510

4897 7590 10/12/2006

ROBERT C. KAIN, JR.
750 SOUTHEAST THIRD AVENUE
SUITE 100
FT LAUDERDALE, FL 333161153

EXAMINER	
ART UNIT	PAPER NUMBER

DATE MAILED: 10/12/2006

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

Notice of Non-Compliant Amendment (37 CFR 1.121) for Applications Under Accelerated Examination	Application No. 10/155,192	Applicant(s) REDLICH ET AL.	
	Examiner Yves Dalencourt	Art Unit 2157	

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

Since this application has been granted special status under the accelerated examination program, NO extensions of time under 37 CFR 1.136(a) will be permitted.

The amendment document filed on *07 August 2006* is considered non-compliant because it has failed to meet the requirements of 37 CFR 1.121 or 1.4. In order for the amendment document to be compliant, correction of the following item(s) is required.

THE FOLLOWING MARKED (X) ITEM(S) CAUSE THE AMENDMENT DOCUMENT TO BE NON-COMPLIANT:

- ☐ 1. Amendments to the specification:
  - ☐ A. Amended paragraph(s) do not include markings.
  - ☐ B. New paragraph(s) should not be underlined.
  - ☐ C. Other _____.
- ☐ 2. Abstract:
  - ☐ A. Not presented on a separate sheet. 37 CFR 1.72.
  - ☐ B. Other _____.
- ☐ 3. Amendments to the drawings:
  - ☐ A. The drawings are not properly identified in the top margin as "Replacement Sheet," "New Sheet," or "Annotated Sheet" as required by 37 CFR 1.121(d).
  - ☐ B. The practice of submitting proposed drawing correction has been eliminated. Replacement drawings showing amended figures, without markings, in compliance with 37 CFR 1.84 are required.
  - ☐ C. Other _____.
- ☒ 4. Amendments to the claims:
  - ☐ A. A complete listing of all of the claims is not present.
  - ☐ B. The listing of claims does not include the text of all pending claims (including withdrawn claims)
  - ☒ C. Each claim has not been provided with the proper status identifier, and as such, the individual status of each claim cannot be identified. Note: the status of every claim must be indicated after its claim number by using one of the following status identifiers: (Original), (Currently amended), (Canceled), (Previously presented), (New), (Not entered), (Withdrawn) and (Withdrawn-currently amended).
  - ☐ D. The claims of this amendment paper have not been presented in ascending numerical order.
  - ☒ E. Other: *See Continuation Sheet*.
- ☐ 5. Other (e.g., the amendment is unsigned or not signed in accordance with 37 CFR 1.4):

  _____.

YVES DALENCOURT
PRIMARY EXAMINER
TECHNOLOGY CENTER 2100

For further explanation of the amendment format required by 37 CFR 1.121, see MPEP § 714.

TIME PERIODS FOR FILING A REPLY TO THIS NOTICE:
Applicant is given **no new time period** if the non-compliant amendment is an after-final amendment or an amendment filed after allowance, or a drawing submission (only). If applicant wishes to resubmit the non-compliant after-final amendment with corrections, the **entire corrected amendment** must be resubmitted.

Applicant is given **one month**, or thirty (30) days, whichever is longer, from the mail date of this notice to supply the correction, if the non-compliant amendment is one of the following: a preliminary amendment, a non-final amendment (including a submission for a request for continued examination (RCE) under 37 CFR 1.114), a supplemental amendment filed within a suspension period under 37 CFR 1.103(a) or (c), and an amendment filed in response to a *Quayle* action. If any of above boxes 1. to 4. are checked, the correction required is only the **corrected section** of the non-compliant amendment in compliance with 37 CFR 1.121.

NO Extensions of time under 37 CFR 1.136(a) will be permitted.

Failure to timely respond to this notice will result in:
Abandonment of the application if the non-compliant amendment is a non-final amendment or an amendment filed in response to a *Quayle* action; or
Non-entry of the amendment if the non-compliant amendment is a preliminary amendment or supplemental amendment.

Legal Instruments Examiner (LIE), if applicable _____ Telephone No. _____

Continuation of 4(e) Other: claim 11 needs to be provided with the proper status identifier.

PATENTS

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re Application of Ron M. Redlich, et al.

Serial No. 10/155,192 Examiner Chung

Filed: May 23, 2002 Group 2143

For: Data Security System and Method With Parsing and Dispersion Techniques

AMENDMENT AFTER FINAL OFFICE ACTION

Mail Stop Amendment
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Sir:

This amendment responds to the Notice of Non-Compliance dated October 12, 2006.

Submitted herewith are:

a. transmittal (1 pg., p. 1)

b. specification (1 pg., p. 2)

c. claims (6 pgs, pp. 3 - 8)

d. remarks (16 pgs, pp. 9 - 24)

[Page left blank]

IN THE SPECIFICATION

[0001] This is a continuation-in-part of patent application no. 10/008,209 and 10/008,218, both filed on December 6, 2001, which are continuations-in-part of patent application no. 09/916,397 filed July 27, 2001 which is a regular patent application is based upon provisional patent application no. 60/260,398, filed ~~June 5,~~ January 9, 2001; application no. 60/287,813, filed on May, 2, 2001; application no. 60/267,944, filed February 12, 2001; application no. 60/247,242, filed November 13, 2000 and application no. 60/247,232, filed November 13, 2000.

IN THE CLAIMS

1. (currently amended) A method of securing data having one or more security sensitive words, characters or icons in a computer system with a plurality of physically distinguishable memories respectively designated as a remainder store and a plurality of physically distinct extract stores corresponding to a plurality of security levels for respective subsets of said security sensitive words, characters or icons ~~(FIG. 6, store 1, 2 or 3; FIG. 7, step 422)~~ comprising:

extracting said security sensitive words, characters or icons from said data to obtain extracted data and remainder data therefrom;

storing said extracted data and said remainder data in said respective extract stores corresponding to an associated security level and said remainder store, respectively; and,

permitting full or partial reconstruction of said data via said extracted data from respective extract stores and remainder data only in the presence of a predetermined security clearance corresponding to said associated security level for said respective extract stores.

2. (original) A method as claimed in claim 1 including filtering said data while extracting.

3. (previously amended) A method as claimed in claim 2 including defining a plurality of filters for said respective subsets of security sensitive words, characters or icons and the method includes one of storing or destroying one or more of said plurality of filters after storing said extracted data.

4. (original) A method as claimed in claim 3 including encrypting one or both of said extracted data and said remainder data.

5. (original) A method as claimed in claim 4 wherein the step of permitting reconstruction includes decrypting one or both of said extracted data and remainder data.

6. (previously amended) A method as claimed in claim 1 including establishing a plurality of security levels each with a respective security clearance, each said subset of said security sensitive words, characters or icons being correlated with said plurality of security levels and the step of permitting reconstruction including the step of permitting a plurality of partial reconstructions of said data in the presence of respective ones of said plurality of security clearance levels.

7. (currently amended) A method of securing data with a parsing algorithm via a computer system having physically distinguishable memories designated as a remainder store and a plurality of physically distinct extract stores corresponding to a plurality of security levels for respective subsets of said security sensitive words, characters or icons comprising:

parsing said data to obtain extracted data for respective associated security levels and remainder data therefrom;

storing said extracted data and said remainder data in said extract stores and said remainder store, respectively; and,

permitting full or partial reconstruction of said data via said extracted data from said respective extract stores and remainder data only in the presence of a predetermined security clearance corresponding to said associated security level for said respective extract stores.

8. (original) A method as claimed in claim 7 wherein the step of parsing parses the data granularly.

9. (original) A method as claimed in claim 7 wherein the step of parsing uses said parsing algorithm which is based on one or more of a bit count, a word, a word count, a page count, a line count, a paragraph count, an identifiable document characteristic, an identifiable word characteristic,

an identifiable icon characteristic, an identifiable data object characteristic, capital letters, italics, and underline.

10. (original) A method as claimed in claim 7 wherein said data is financial data and the step of parsing uses said parsing algorithm based upon one of characters in said data and character count in said data.

11. (currently amended) A method as claimed in claim 7 including establishing a plurality of security levels each with a respective security clearance, permitting increasingly greater reconstruction if said data being correlated with increasingly higher levels of said plurality of security levels includes greater ~~securyt~~ security level clearances, and the step of permitting reconstruction including the step of permitting a plurality of partial reconstructions of said data in the presence of respective ones of said plurality of security clearance levels.

12. (original) A method as claimed in claim 7 including a further data security step of one from the group of storing said data without encryption, storing said data with encryption and destroying said data; said further data security step being independent of said parsing, storing and permitting reconstruction.

13. (original) A method as claimed in claim 1 including displaying an advertisement prior to one or the other or both of said steps of extracting and permitting reconstruction.

14. (original) A method as claimed in claim 13 including providing a hyperlink to a vendor's web site with said advertisement.

15. (original) A method as claimed in claim 1 including assessing a charge for extracting, storing or permitting reconstruction.

16. (currently amended) A computer readable medium containing programming instructions for securing data having one or more security sensitive words, characters or icons in a computer system with a plurality physically distinguishable memories respectively designated as a remainder store and a plurality of physically distinct extract stores corresponding to a plurality of security levels for respective subsets of said security sensitive words, characters or icons comprising:

extracting said security sensitive words, characters or icons from said data to obtain extracted data and remainder data therefrom;

storing said extracted data and said remainder data in said respective extract stores corresponding to an associated security level and said remainder store, respectively; and,

permitting full or partial reconstruction of said data via said extracted data from respective extract stores and remainder data only in the presence of a predetermined security clearance corresponding to said associated security level for said respective extract stores.

17. (original) A medium with programming instructions as claimed in claim 16 including filtering said data while extracting.

18. (currently amended) A medium with programming instructions as claimed in claim 16 including defining a plurality of filters for respective subsets of security sensitive words, characters or icons and ~~the method~~ includes one of storing or destroying one or more of said plurality of filters after storing said extracted data.

19. (original) A medium with programming instructions as claimed in claim 18 including encrypting one or both of said extracted data and said remainder data.

20. (original) A medium with programming instructions as claimed in claim 19 wherein the step of permitting reconstruction includes decrypting one or both of said extracted data and remainder data.

21. (previously amended) A medium with programming instructions as claimed in claim 16 including establishing a plurality of security levels each with a respective security clearance, each said subset of said security sensitive words, characters or icons being correlated with said plurality of security levels and the step of permitting reconstruction including the step of permitting a plurality of partial reconstructions of said data in the presence of respective ones of said plurality of security clearance levels.

22. (currently amended) A computer readable medium containing programming instructions for securing data with a parsing algorithm via a computer system having physically distinguishable memories designated as a remainder store and a plurality of physically distinct extract stores corresponding to a plurality of security levels for respective subsets of said security sensitive words, characters or icons comprising:

parsing said data to obtain extracted data for respective associated security levels and remainder data therefrom;

storing said extracted data and said remainder data in said respective extract stores and said remainder store, respectively; and,

permitting full or partial reconstruction of said data via said extracted data from said respective extract stores and remainder data only in the presence of a predetermined security clearance corresponding to said associated security level for said respective extract stores.

23. (original) A medium with programming instructions as claimed in claim 22 wherein the step of parsing parses the data granularly.

24. (original) A medium with programming instructions as claimed in claim 22 wherein the step of parsing uses said parsing algorithm which is based on one or more of a bit count, a word, a word count, a page count, a line count, a paragraph count, an identifiable document characteristic, an identifiable word characteristic, an identifiable icon characteristic, an identifiable data object characteristic, capital letters, italics, and underline.

25. (original) A medium with programming instructions as claimed in claim 22 wherein said data is financial data and the step of parsing uses said parsing algorithm based upon one of characters in said data and character count in said data.

26. (currently amended) A medium with programming instructions as claimed in claim 22 including establishing a plurality of security levels each with a respective security clearance, permitting increasingly greater reconstruction if said data being correlated with increasingly higher levels of said plurality of security levels includes greater ~~secuiryt~~ security level clearances, and the step of permitting reconstruction including the step of permitting a plurality of partial reconstructions of said data in the presence of respective ones of said plurality of security clearance levels.

27. (original) A medium as claimed in claim 22 including a further data security step of one from the group of storing said data without encryption, storing said data with encryption and destroying said data; said further data security step being independent of said parsing, storing and permitting reconstruction.

REMARKS

This amendment responds to the final Office Action dated June 1, 2006.

In the Office Action dated June 1, 2006, on pages 2-11, the patent examiner rejects all the claims as being non-patentable in view of certain prior art disclosed in the following references:

U.S. Patent No. 5,960,080 to Fahlman et al.
U.S. Patent No. 6,602,298 to Kirshenbaum
U.S. Patent No. 6,073,165 to Narasimhan et al.
U.S. Patent No. 6,487,538 to Gupta
U.S. Patent No. 6,771,290 to Hoyle

Previously in the Office Action dated August 11, 2005, the examiner, rejected all claims 1-27 as being non-patentable in view of:

U.S. Patent No. 5,960,080 to Fahlman
U.S. Patent No. 6,602,298 to Kirshenbaum
U.S. Patent No. 6,487,538 to Gupta
U.S. Patent No. 6,771,290 to Hoyle
Published U.S. Application 2002/0116641 to Mastrianni \

In the present Office Action, the examiner confirmed that Mastrianni '641 is not prior art.

In the parent case, Serial No. 09/916,397, filed July 27, 2001, examiner Pyzocha rejected the claims in light of Kirshenbaum '298 (U.S. Patent No. 6,602,298), Lamm '907 (U.S. Patent No. 6,078,907) and/or Fahlman '080 (U.S. Patent No. 5,960,080) but ultimately withdrew such rejections based upon Kirshenbaum '298 and Fahlman '080 (and Lamm'907) when the claims were directed to a "multi-level security" system wherein different security words, for different security levels, were extracted and stored in different extract stores corresponding to respective security levels. Full or partial reconstruction is permitted with different security clearances for each security level (corresponding to different extract stores). The parent case 09/916,397 has been allowed after another search. A sample of one allowed claim is attached as an Addendum hereto. Examiner

Pyzocha located new reference, U.S. Patent No. 6,253,203 to O'Flaherty, during that further search but did not reject the claims based thereon.

In this case, independent claims 1, 7, 16 and 22 were amended on February 16, 2006 to refer to multiple levels of security. Currently, claims 1, 7, 16 and 22 are amended to refer to "physically distinguishable memories" and "physically distinct extract stores" and "reconstruction ... from respective extract stores ...[with] predetermined security clearances." Claims 11 and 18 and 26 are amended to correct typographic errors. The specification has been amended to correct the claim for priority (see supplemental declaration). Claim 1 is an example which refers to multiple security levels:

> A method of securing data having one or more security sensitive words... with a plurality of physically distinguishable memories respectively designated as a remainder store and a plurality of physically distinct extract stores corresponding to a plurality of security levels for respective subsets of said security sensitive words, characters or icons comprising:
>
> extracting ...
>
> storing said extracted data and said remainder data in said respective extract stores corresponding to an associated security level and said remainder store, respectively; and,
>
> permitting full or partial reconstruction of said data via said extracted data from respective extract stores and remainder data only in the presence of a predetermined security clearance corresponding to said associated security level for said respective extract stores.

It is respectfully submitted that the amended claims 1, 7, 16 and 22 all include the concept that various levels of secret data are stored in "respective extract stores" which are physically distinguishable and physically distinct and this is essentially the same as multiple levels of security with multiple levels of secure storage.

In summary, Kirshenbaum '298 and/or Fahlman '080 do not show, teach or suggest multiple extractions of security sensitive words related to each of a plurality of security levels, and separate

storage of those security sensitive words in different, physically distinct secured locations for each security level ("respective extract stores"), and the requirement that the user input a password ("security clearance") for each security level to reconstruct the data from "respective extract stores."

The advantages of the multiple security level - multiple dispersal and storage claim are:

- distributed, secure storage of multiple levels of secure data

- less risk of loss of secure data by attacks (intrusions) due to the dispersed storage of data

- system provides more physical barriers to electronic intrusions into storage sites

- avoids single point of machine-security clearance failure (possible with prior art systems)

- enhances the greater sharing of data by granular separation and dispersed storage of data at multiple levels of security

Kirshenbaum '298 does not show separate storage of secured data, separate and apart from unsecured data.[1] Both secured and non-secured data is stored in a single database 14. Col. 3, lines 40-44; col. 5, lines 7-10 ("The data set is stored in a database ... the document comprises secure portion and non-secure portions"); col. 5, lines 36-37 ("to retrieve those secure and non-secure portions of the document and to send the retrieved portions of the document to the output device.").

Contrary to the Examiner's contention, Kirshenbaum '298 does not show nor teach physically distinguishable memories, nor physically distinct extract stores. The Examiner suggests that the different portions of the single document in Kirshenbaum '298 represent different stores, which different stores can be called "extract stores" (Examiner's characterization). However, it is well known that computer memory, when it stores a single document, does not locate all portions

[1] Due to the confusing nature of Kirshenbaum '298, a brief analysis is provided here in conjunction with Fahlman '080. A detailed analysis is presented later.

of that document in a single, physically distinct and physically distinguishable memory location. The document storage location, and all parts thereof, are stored at generally randomly assigned locations on the hard drive or memory disk. The randomness is based, in part, upon the storage of other documents on the hard drive or memory disk. Physically separate memories with physically distinct extract stores are not the same as the general concept of storing a document in a general purpose computer. The confusion caused by the ambiguity in Kirshenbaum '298 is described in detail in Applicant's Detailed Analysis at the end of these Remarks.

To have physically distinguishable memories and physically distinct extract stores, the location of the stores must be effectively static. These effectively static physically distinguishable memories and physically distinct extract stores are almost a necessity in light of the secure nature of the entire system. In other words, to maintain multiple security levels, each with a respective security clearance, the corresponding extract stores must be defined and confined within certain physically distinguishable memories and physically distinct extract stores. Otherwise, one could not maintain a high degree of security for the entire system.

If the Kirshenbaum "store anywhere" system was employed, the security system could be easily compromised by a hack attack on the document itself. With physically distinguishable memories and physically distinct extract stores, each with a separate security clearance, the quality of security is greatly enhanced.

Kirshenbaum '298 does not have storing said remainder [non-secret] data in said remainder store. Kirshenbaum '298 teaches away from this claimed aspect of the invention.

Kirshenbaum '298 does not show extraction of many levels of secured data from a document. At best, Kirshenbaum '298 discloses scanning and storing <u>of identifying codes</u> but the identifier

codes are not secured data. "In a particular embodiments, the processor is configured to identify a security clearance level of a user and then to enable the user to print portions of a document at any security level below the user's level." Col. 5, line 54. "The identifier codes printed on non-secure portions of a document are machine readable ... and are provided to a processor by extracting the identifier codes from the machine readable format with a scanning machine." Col. 5, line 64 - Col. 6, line 1.

The present invention requires multiple security levels, each having a sub-set of security sensitive words, extraction and storage for each of the security levels to respective physically distinct extract stores, a plurality of predetermined security clearances, and particular security clearances needed to reconstruct data from each of the respective extract stores.

In the present invention, different security clearances must be input to reconstruct data from each extract store (each security level storage facility) in order to obtain the data and permit full or partial reconstruction. Therefore, the present invention requires multiple extraction, each extraction for each security level, separate storage thereof, storage of the remainder or non-secured data, and, permitting full or partial reconstruction of said data via said extracted data and remainder data only with predetermined security clearances for respective extract stores.

Fahlman '080 does not show, teach or suggest a remainder store for non-secured data, multiple security levels, multiple extraction of security data, storage of extract data at multiple levels, nor presentment of corresponding security codes at each security level. In fact, nowhere does Fahlman '080 discuss password or security clearance control.

In Fig. 1, step 111 of Fahlman "automatically" merges sensitive information into non-sensitive information. In FIG. 2, step 217, the Fahlman system again "automatically" merges

secured data with un-secured data. Fahlman discusses merging secured data with unsecured data at col. 2, line 22, col. 2, line 34, and col. 2, line 49, and col. 2, line 54 and col. 2, line 40. Fahlman '080 discloses identifying security information and extracting that information and replacing it with place holders. Col. 2, line 37 and col. 3, line 47. "The sanitized message is then transmitted with a low level of security." Col. 3, line 54. The sensitive information is stripped from the original message and stored in a separate location or together with the sanitized message. Col. 3, line 64. A map showing the location of the security information is also stored with the secured information. Col. 3, line 63. "Then, in step 111, the sensitive terms received from the second path are merged with the sanitized message to create a final confidential message." Col. 4, line 1. The examiner should note that Fahlman does not discuss reconstruction or merger in the presence of any type of security clearance. In contrast, the present invention requires multiple security clearances, each unique to a security level, wherein these secured extract stores are physically distinct, in physically distinguishable memories. Fahlman also states: "Then, in step 217, the sensitive terms are automatically merged back into the serviced message to create a final message." Col. 5, line 27. No discussion of multi-level security clearance is noted. The same is true regarding merger of security information and non-secured information at col. 6, line 62.

Fahlman does not seem to store remainder or non-secure data in a separate remainder store location apart from secure data. Fahlman (1) identifies secret words, (2) replaces the words with placeholders, then (3) transmits the "sanitized message." Fig. 1, step 107, Fig. 2, step 209, col 2, line 19, col. 2, line 34, col. 2, line 49, col. 3, line 54 (transmission - no storage of non-secure data), col. 4, line 64 (transmission - no separate storage). Fahlman does discuss storing the sanitized message, that is, the non-secure data and the placeholders, but not separate storage of the remainder

data. Col. 3, line 63, col. 4, line 47, col. 5, line 1. In the present invention, remainder, non-secret data is stored separately from secret data. See claim 1, "storing" step. Fahlman teaches away from this aspect of the present invention.

Fahlman does not discuss multiple security levels nor multiple storage sites for each security level. Since multiple security levels are not addressed in Fahlman, multiple security clearance codes to obtain access and reconstruct the secured data from multiple extract stores is not discussed. Since Fahlman does not use any type of password or security clearance procedures, there is no need for a plurality of predetermined security clearances to reconstruct said data from multiple extract stores only in the presence of said predetermined security clearances.

There is no motivation nor suggestion to combine Kirshenbaum's presentment of a security code with Fahlman's single level extraction and storage of secret data. There is no suggestion nor motivation to use security codes to reconstruct "full or partial" data from respective extract stores in the presence of such security codes.

DETAILED ANALYSIS OF Kirshenbaum '298

The technical disclosure of Kirshenbaum '298 is difficult to read and understand. Applicant herein will first reproduce key portions of the Kirshenbaum '298 text without comment, and then provide commentary.

"Content manipulator 12 is utilized to create a communication 60 having a plurality of sub-components (62, 64, 66, 68, 70, 72 and 74)." col. 3, line 20, herein "3/20."

Embodiment One

"In the embodiment of FIG. 1, content manipulator 12 is in data communication with a database 14 and is configured to store communication 60 within database 14." 3/33.

> Content manipulator 12 is preferably configured to save a plurality of machine-readable identifier codes associated with communication 60 in database 14, and which identify different sub-components (62, 64, 66, 68, 70, 72 and 74) of communication 60. Database 14 can comprise, for example, a network, such as, for example, the Internet. Database 14 is configured for electronic storage of the sub-components of communication 60 and the machine-readable identifier codes associated with communication 60.

3/39.

"Printer 16 is utilized to print sub-components 62, 64 and 66 of communication 60." 3/52.

"Processor 32 is configured to identify all sub-components of communication 60 that are stored on database 14, and to retrieve at least one sub-component from database 14 that was not printed as part of document 24." 4/15.

"The sub-components retrieved from database 14 by processor 32 are sent to output device 34 and subsequently displayed as a second document 44." 2/33.

Embodiment Two

"Another particular method of application of the system 10 of FIG. 1 is to protect secure portions of a document while enabling distribution of non-secure portions of the document (the non-secure portions can be separate portions from the secure portions, or can be non-secure variants of the secure portions)." 4/48.

> In an initial step of FIG. 2 (labeled as 100) a data set is created which corresponds to a document. The data set can be created utilizing, for example, the content manipulator 12 described above with reference to FIG. 1. The data set is stored in a database (which can comprise, for example, the database 14 of FIG. 1). The document comprises secure portions and non-secure portions. Such secure portions and non-secure portions can correspond to sub-components 62, 64, 66, 68, 70, 72 and 74 of a document 60 described above with reference to FIG. 1.

5/2.

"The processor is configured to retrieve both secure and non-secure portions of the document, and to send the retrieved portions of the document to the output device. The processor is further configured to recognize if an access code provides clearance to the secure portions of the document." 5/36.

> In particular embodiments, a user may be required to input its individual access code before any portions of a document are sent to the output device. In other embodiments, the user may be required to input its individual access code only before secure portions of the document are sent to the output device. In particular embodiments, the processor is configured to identify a security clearance level of a user and then to enable the user to print portions of a document at any security level below the user's level.

5/48.

Embodiment Three

> Another application of the present invention is described with reference to a flow chart of FIG. 3. At an initial step 200, a data set is provided on the Internet. The data set has an Internet address associated therewith (for purposes of interpreting ... includes any information that locates the data set on the internet, such as, for example, a URL). At a subsequent step 202, the Internet address is displayed on a substrate. The substrate can comprise, for example, paper, ... display screen of a television, ... a movie theater, or ... a billboard. In applications in which a URL address is broadcast by radio or television, the substrate can comprise a piece of paper upon which a user has jotted the address.

6/9.

"At a next step 204, the address is provided to a machine configured to digitize the address, link to the address on the Internet, and download at least portion of the data set." 6/25.

APPLICANT'S ANALYSIS

Applicant respectfully submits that Kirshenbaum '298 (a) poorly defines and uses the terms "document" and "data set" and "database" and effectively describes three different systems. By defining database 14 as "comprising" a network and the Internet, Kirshenbaum '298 has created confusion as to where the sub-components are stored. The following passage states that "identifier

codes" are used to "identify different sub-components... [62 - 74]" and these identifier codes are stored in the database 14, and, by definition, the identifier codes are stored on the Internet.

> Content manipulator 12 is preferably configured to save a plurality of machine-readable identifier codes associated with communication 60 in database 14, and which identify different sub-components (62, 64, 66, 68, 70, 72 and 74) of communication 60. Database 14 can comprise, for example, a network, such as, for example, the Internet. Database 14 is configured for electronic storage of the sub-components of communication 60 and the machine-readable identifier codes associated with communication 60.

3/39.

Therefore, the sub-components 62 - 74 are INITIALLY stored in the Internet, and the identifier codes associated with sub-components 62 - 74 are subsequently stored on the Internet. There is no EXTRACTION of sub-components 62 - 74.

This interpretation of Kirshenbaum '298 is consistent with the Third Embodiment wherein the data set includes an Internet address which is used by consumer to locate and obtain all or portions of the document.

> Another application of the present invention is described with reference to a flow chart of FIG. 3. At an initial step 200, a data set is provided on the Internet. The data set has an Internet address associated therewith (for purposes of interpreting ... includes any information that locates the data set on the internet, such as, for example, a URL). At a subsequent step 202, the Internet address is displayed on a substrate. The substrate can comprise, for example, paper, ... display screen of a television, ... a movie theater, or ... a billboard. In applications in which a URL address is broadcast by radio or television, the substrate can comprise a piece of paper upon which a user has jotted the address.

6/9.

"At a next step 204, the address is provided to a machine configured to digitize the address, link to the address on the Internet, and download at least portion of the data set." 6/25. Multiple Internet addresses can be displayed, the user can select one or more and the system downloads selected site data. See 6/36.

However, the Second Embodiment (beginning at 4/48) seems to be completely different than the First and Third Embodiments in that the Second Embodiment discusses retrieval of secure and non-secure data.

> In an initial step of FIG. 2 (labeled as 100) a data set is created which corresponds to a document. The data set can be created utilizing, for example, the content manipulator 12 described above with reference to FIG. 1. The data set is stored in a database (which can comprise, for example, the database 14 of FIG. 1). The document comprises secure portions and non-secure portions. Such secure portions and non-secure portions can correspond to sub-components 62, 64, 66, 68, 70, 72 and 74 of a document 60 described above with reference to FIG. 1.

5/2.

However, the First Embodiment clearly states that it is the machine-readable identifier codes that are "saved" in the database 14 (which maybe on the Interent). See quote below regarding "saving" identifiers.

> Content manipulator 12 is preferably configured to save a plurality of machine-readable identifier codes associated with communication 60 in database 14, and which identify different sub-components (62, 64, 66, 68, 70, 72 and 74) of communication 60. Database 14 can comprise, for example, a network, such as, for example, the Internet. Database 14 is configured for electronic storage of the sub-components of communication 60 and the machine-readable identifier codes associated with communication 60.

3/39.

In conclusion, Kirshenbaum '298 does not show, teach or suggest that the sub-components 62 - 74 are selectively extracted, due to some security issues, then stored on in separate memories or on the Internet, but rather: "Content manipulator 12 is ... configured to save ... identifier codes associated with communication 60 in database 14, and which [identifier codes] identify different sub-components (62, 64, 66, 68, 70, 72 and 74) of communication 60." 3/39. "Database 14 [in which is saved the identifier codes] can comprise, for example, a network, such as, for example, the Internet." 3/45. Kirshenbaum '298 then goes on the globally state: "Database 14 is configured for electronic storage of the sub-components of communication 60 and the machine-readable identifier codes associated with communication 60." 3/46. This global statement, when understood in context

with the specific statement regarding "saving identifier codes in a database on the Internet" is not a clear enough disclosure to show, teach or suggest the specific, claimed steps of the present invention.

> A method of securing data having one or more security sensitive words... with a plurality of physically distinguishable memories respectively designated as a remainder store and a plurality of physically distinct extract stores corresponding to a plurality of security levels for respective subsets of said security sensitive words, characters or icons comprising:
>
> extracting ...
>
> storing said extracted data and said remainder data in said respective extract stores corresponding to an associated security level and said remainder store, respectively; and,
>
> permitting full or partial reconstruction of said data via said extracted data from respective extract stores and remainder data only in the presence of a predetermined security clearance corresponding to said associated security level for said respective extract stores.

To make the argument differently, if the Examiner believes that a database is any organized set of data, and if the person of ordinary skill in the art understands, according to Kirshenbaum '298, that the Internet is included in the "database," then each and every electronic document on any network falls within this definition. The communication 60 and all sub-components 62 - 74 are on the Internet-network. However, just because every electronic document falls within the definition, this does not explain the (a) extraction and storage into physically distinct extract stores based upon multiple levels of security; and (b) reconstruction based upon multiple security clearances from those multiple secured extract stores.

It is respectfully submitted that the Examiner is unjustifiably extrapolating Kirshenbaum's specific statement regarding "saving identifier codes in a database on the Internet" 3/39, and, based upon hindsight learned only after reading the present invention, assumes that this "saving identifier

codes" is the same as extraction and storage of security words or icons into physically distinct extract stores based upon multiple levels of security. [2]

Simply put, Kirshenbaum '298 does not show extraction for multiple levels of security. The Second Embodiment in Kirshenbaum '298 assumes the document, subject to the security system, is already stored in database. 5/2 [3]

Each of the references, Kirshenbaum '298, and Fahlman '080, describe complete systems and there is no reason to add to or substitute portions of one disclosure with another. For example, there is no reason to combine Kirshenbaum's singular document storage and reproduction system with Fahlman's secret transmission system. It is improper to take the multiple security level - password system of Kirshenbaum '298 (Second Embodiment), the single extract process of Fahlman '080, and somehow extrapolate into (a) extraction of multiple security sensitive words or icons and (b) separate storage of extracted items into physically distinct memories, and (c) full or partial reconstruction from those separate extract stores inthe presence of predetermined respective security clearances.

---

[2] For sake of clarity, Kirshenbaum '298 states: "Content manipulator 12 is preferably configured to save a plurality of machine-readable identifier codes associated with communication 60 in database 14, and which identify different sub-components (62, 64, 66, 68, 70, 72 and 74) of communication 60. Database 14 can comprise, for example, a network, such as, for example, the Internet. Database 14 is configured for electronic storage of the sub-components of communication 60 and the machine-readable identifier codes associated with communication 60." 3/39.

[3] "In an initial step of FIG. 2 (labeled as 100) a data set is created which corresponds to a document. The data set can be created utilizing, for example, the content manipulator 12 described above with reference to FIG. 1. The data set is stored in a database (which can comprise, for example, the database 14 of FIG. 1). The document comprises secure portions and non-secure portions. Such secure portions and non-secure portions can correspond to sub-components 62, 64, 66, 68, 70, 72 and 74 of a document 60 described above with reference to FIG. 1." 5/2

The Manual of Patent Examining Procedures ("MPEP") explains that it is not obvious to modify a system with a plurality of sensors controlling a plurality of valves to create the inventive and patentable single sensor which controls a plurality of valves:

> In *In re Kotzab,* 217 F.3d 1365, 55 USPQ2d 1313 (Fed.Cir. 2000), the claims were drawn to an injection molding method using a single temperature sensor to control a plurality of flow control valves. The primary reference disclosed a multizone device having multiple sensors, each of which controlled an associated flow control valve, and also taught that one system may be used to control a number of valves. The court found that there was insufficient evidence to show that one system was the same as one sensor. While the control of multiple valves by a single sensor rather than by multiple sensors was a "technologically simple concept," there was no finding "as to the specific, understanding or principle within the knowledge of the skilled artisan" that would have provided the motivation to use a single sensor as the system to control more than one valve. 217 F.3d at 1371, 55 USPQ2d at 1318.

MPEP § 2143.01.

In the present invention, Applicant has developed a system designed to safeguard a plurality of security levels with specific dispersal and retrieval techniques. The single data storage systems of the prior art Fahlman '080 and Kirshenbaum '298 cannot be extended to the claimed multi-level security storage systems. Each of these references discloses extraction and single storage of security words or icons in physically separate memories.

> The mere fact that references can be combined or modified does not render the resultant combination obvious unless the prior art also suggests the desirability of the combination. *In re Mills*, 916 F.2d 680, 16 USPQ2d 1430 (Fed. Cir. 1990) (Claims were directed to an apparatus for producing an aerated cementitious composition by drawing air into the cementitious composition by driving the output pump at a capacity greater than the feed rate. The prior art reference taught that the feed means can be run at a variable speed, however the court found that this does not require that the output pump be run at the claimed speed so that air is drawn into the mixing chamber and is entrained in the ingredients during operation. Although a prior art device "may be capable of being modified to run the way the apparatus is claimed, there must be a suggestion or motivation in the reference to do so." 916 F.2d at 682, 16 USPQ2d at 1432.).

MPEP § 2143.01.

In the prior art cited herein, there is no suggestion of the desirability of multi-level secured data storage, extraction for each security level, and full or partial reconstruction using multiple extract stores. Further, the examiner's suggestion -- that one can take certain aspects of

Kirshenbaum '298 and mix it with other aspects of Fahlman '080 -- "modifies" that reference to an unacceptable degree. "If proposed modification would render the prior art invention being modified unsatisfactory for its intended purpose, then there is no suggestion or motivation to make the proposed modification. *In re Gordon*, 733 F.2d 900, 221 USPQ 1125 (Fed. Cir. 1984)." MPEP § 2143.01. This change to Fahlman '080 modifies the operation of the reference improperly. "If the proposed modification or combination of the prior art would change the principle of operation of the prior art invention being modified, then the teachings of the references are not sufficient to render the claims prima facie obvious. *In re Ratti*, 270 F.2d 810, 123 USPQ 349 (CCPA 1959)." MPEP § 2143.01.

With respect to Gupta '538 and Hoyle '290, those references are cited by the examiner as secondary references. The secondary references do not cure the defects described above with respect to the primary references. With respect to Narasimhan '165, that reference does not cure the defects noted above with respect to the primary references.

Applicant respectfully requests that the examiner withdraw the rejections and approve the patentability of claims 1 - 27.

Respectfully submitted,
By /Robert Kain/

Robert C. Kain, Jr.
Reg. No. 30,648
Fleit, Kain, Gibbons, Gutman, Bongini & Bianco, P.L.
750 Southeast Third Avenue, Suite 100
Fort Lauderdale, FL 33316-1153
Telephone: 954-768-9002
Facsimile: 954-768-0158

Certificate of e-Filing

I hereby certify that this correspondence is being e-Filed with the Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450 on October 17, 2006.

/Robert Kain/
Robert C. Kain, Jr.
Reg. No. 30,648

ADDENDUM

90. (parent case 09/916,397 - one of several allowed claims) A method for securing data in a computer network with one or more security sensitive words, characters or icons and a plurality of security levels each with a respective security clearance, subsets of said security sensitive words, characters or icons being correlated with respective ones of said plurality of security levels, said computer network having a plurality of computers interconnected together, each of said plurality of computers having a memory therein, one of said plurality of computers designated as a data input computer, said plurality of computers including a first computer designated as a remainder store and a further plurality of computers designated as extract stores for respective ones of said plurality of security levels, comprising:

extracting said security sensitive words, characters or icons for said plurality of security levels from said data to obtain extracted data for respective security levels and remainder data;

storing said extracted data in extract stores corresponding to respective security levels and said remainder data in said remainder store;

presenting a plurality of predetermined security clearances to obtain access to respective ones of said extract stores; and,

permitting full or partial reconstruction of said data via said extracted data and remainder data only in the presence of respective ones of said predetermined security clearances after presentment thereof.

Electronic Acknowledgement Receipt

EFS ID:	1256238
Application Number:	10155192
International Application Number:	
Confirmation Number:	9510
Title of Invention:	Data security system and method with parsing and dispersion techniques
First Named Inventor/Applicant Name:	Ron M. Redlich
Customer Number:	4897
Filer:	Robert C. Kain
Filer Authorized By:	
Attorney Docket Number:	6851-2-CIP-3
Receipt Date:	17-OCT-2006
Filing Date:	23-MAY-2002
Time Stamp:	10:00:24
Application Type:	Utility

Payment information:

Submitted with Payment	no

File Listing:

Document Number	Document Description	File Name	File Size(Bytes)	Multi Part /.zip	Pages (if appl.)
1	Amendment After Final	6851-02cip3-amdt-3d-101706-trans.pdf	18043	no	1
Warnings:					

Information:					
2	Specification	6851-02cip3-amdt-3d-101706-spec.pdf	16329	no	1
Warnings:					
Information:					
3	Claims	6851-02cip3-amdt-3d-101706-clms.pdf	27573	no	6
Warnings:					
Information:					
4	Applicant Arguments/Remarks Made in an Amendment	6851-02cip3-amdt-3d-101706-rem.pdf	59155	no	16
Warnings:					
Information:					
		Total Files Size (in bytes):	121100		

This Acknowledgement Receipt evidences receipt on the noted date by the USPTO of the indicated documents, characterized by the applicant, and including page counts, where applicable. It serves as evidence of receipt similar to a Post Card, as described in MPEP 503.

New Applications Under 35 U.S.C. 111
If a new application is being filed and the application includes the necessary components for a filing date (see 37 CFR 1.53(b)-(d) and MPEP 506), a Filing Receipt (37 CFR 1.54) will be issued in due course and the date shown on this Acknowledgement Receipt will establish the filing date of the application.

National Stage of an International Application under 35 U.S.C. 371
If a timely submission to enter the national stage of an international application is compliant with the conditions of 35 U.S.C. 371 and other applicable requirements a Form PCT/DO/EO/903 indicating acceptance of the application as a national stage submission under 35 U.S.C. 371 will be issued in addition to the Filing Receipt, in due course.

PTO/SB/06 (12-04)
Approved for use through 7/31/2006. OMB 0651-0032
U.S. Patent and Trademark Office; U.S. DEPARTMENT OF COMMERCE
Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it displays a valid OMB control number.

PATENT APPLICATION FEE DETERMINATION RECORD Substitute for Form PTO-875	Application or Docket Number 10155192	Filing Date: 05/23/2002	☐ To be Mailed

APPLICATION AS FILED – PART I

	(Column 1)	(Column 2)	SMALL ENTITY ☒		OR	OTHER THAN SMALL ENTITY	
FOR	NUMBER FILED	NUMBER EXTRA	RATE ($)	FEE ($)		RATE ($)	FEE ($)
☐ BASIC FEE (37 CFR 1.16(a), (b), or (c))	N/A	N/A	N/A			N/A	
☐ SEARCH FEE (37 CFR 1.16(k), (i), or (m))	N/A	N/A	N/A			N/A	
☐ EXAMINATION FEE (37 CFR 1.16(o), (p), or (q))	N/A	N/A	N/A			N/A	
TOTAL CLAIMS (37 CFR 1.16(i))	minus 20 =	*	X $25 =		OR	X $50 =	
INDEPENDENT CLAIMS (37 CFR 1.16(h))	minus 3 =	*	X $100 =			X $200 =	
☐ APPLICATION SIZE FEE (37 CFR 1.16(s))	If the specification and drawings exceed 100 sheets of paper, the application size fee due is $250 ($125 for small entity) for each additional 50 sheets or fraction thereof. See 35 U.S.C. 41(a)(1)(G) and 37 CFR 1.16(s).						
☐ MULTIPLE DEPENDENT CLAIM PRESENT (37 CFR 1.16(j))			+ $180	0		+$360	
* If the difference in column 1 is less than zero, enter "0" in column 2.			TOTAL			TOTAL	

APPLICATION AS AMENDED – PART II

		(Column 1)		(Column 2)	(Column 3)	SMALL ENTITY		OR	OTHER THAN SMALL ENTITY	
	10.17.06	CLAIMS REMAINING AFTER AMENDMENT		HIGHEST NUMBER PREVIOUSLY PAID FOR	PRESENT EXTRA	RATE ($)	ADDITIONAL FEE ($)		RATE ($)	ADDITIONAL FEE ($)
AMENDMENT A	Total (37 CFR 1.16(i))	* 27	Minus	** 27	= 0	X $25 =	0	OR	X $50=	
	Independent (37 CFR 1.16(h))	* 4	Minus	** 4	= 0	X $100 =	0	OR	X $200=	
	☐ Application Size Fee (37 CFR 1.16(s))									
	☐ FIRST PRESENTATION OF MULTIPLE DEPENDENT CLAIM (37 CFR 1.16(j))							OR		
						TOTAL ADD'L FEE		OR	TOTAL ADD'L FEE	

		(Column 1)		(Column 2)	(Column 3)					
		CLAIMS REMAINING AFTER AMENDMENT		HIGHEST NUMBER PREVIOUSLY PAID FOR	PRESENT EXTRA	RATE ($)	ADDITIONAL FEE ($)		RATE ($)	ADDITIONAL FEE ($)
AMENDMENT B	Total (37 CFR 1.16(i))	*	Minus	**	=	X $25 =		OR	X $50 =	
	Independent (37 CFR 1.16(h))	*	Minus	**	=	X $100 =		OR	X $200 =	
	☐ Application Size Fee (37 CFR 1.16(s))									
	☐ FIRST PRESENTATION OF MULTIPLE DEPENDENT CLAIM (37 CFR 1.16(j))							OR		
	CALCULATE					TOTAL ADD'L FEE		OR	TOTAL ADD'L FEE	

* If the entry in column 1 is less than the entry in column 2, write "0" in column 3.

** If the "Highest Number Previously Paid For" IN THIS SPACE is less than 20, enter "20".

*** If the "Highest Number Previously Paid For" IN THIS SPACE is less than 3, enter "3".

The "Highest Number Previously Paid For" (Total or Independent) is the highest number found in the appropriate box in column 1.

Legal Instrument Examiner:
Tammy Acree

This collection of information is required by 37 CFR 1.16. The information is required to obtain or retain a benefit by the public which is to file (and by the USPTO to process) an application. Confidentiality is governed by 35 U.S.C. 122 and 37 CFR 1.14. This collection is estimated to take 12 minutes to complete, including gathering, preparing, and submitting the completed application form to the USPTO. Time will vary depending upon the individual case. Any comments on the amount of time you require to complete this form and/or suggestions for reducing this burden, should be sent to the Chief Information Officer, U.S. Patent and Trademark Office, U.S. Department of Commerce, P.O. Box 1450, Alexandria, VA 22313-1450. DO NOT SEND FEES OR COMPLETED FORMS TO THIS ADDRESS. **SEND TO: Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450.**

If you need assistance in completing the form, call 1-800-PTO-9199 and select option 2.

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.
10/155,192	05/23/2002	Ron M. Redlich	6851-2-CIP-3	9510

4897 7590 09/20/2007

ROBERT C. KAIN, JR.
750 SOUTHEAST THIRD AVENUE
SUITE 100
FT LAUDERDALE, FL 333161153

EXAMINER	
DALENCOURT, YVES	
ART UNIT	**PAPER NUMBER**
2157	

MAIL DATE	DELIVERY MODE
09/20/2007	PAPER

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

The time period for reply, if any, is set in the attached communication.

MN

Office Action Summary	Application No.	Applicant(s)	
	10/155,192	REDLICH ET AL.	
	Examiner	Art Unit	
	Yves Dalencourt	2157	

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

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *17 October 2006*.

2a)☐ This action is **FINAL**. 2b)☒ This action is non-final.

3)☐ Since this application is in condition for allowance 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.

Disposition of Claims

4)☒ Claim(s) *1-27* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-27* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☐ The drawing(s) filed on _____ is/are: a)☐ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☐ Information Disclosure Statement(s) (PTO/SB/08) Paper No(s)/Mail Date _____.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____.
5) ☐ Notice of Informal Patent Application
6) ☐ Other: _____.

DETAILED ACTION

This office action is responsive to Request for Continued Examination (RCE) filed on 10/17/2006.

Response to Amendment

The Examiner has acknowledged the amended claims 1, 7, 11, 16, 18, 22, and 26.

Claim Rejections - 35 USC § 101

35 U.S.C. 101 reads as follows:

> Whoever invents or discovers any new and useful process, machine, manufacture, or composition of matter, or any new and useful improvement thereof, may obtain a patent therefor, subject to the conditions and requirements of this title.

Claims 16 and 22 are rejected under 35 U.S.C. 101 because the claimed invention is directed to non-statutory subject matter.

Claim 16 and 22 are not limited to tangible embodiments. In view of Applicant's disclosure (see specification paragraph [0095]), the computer-readable media is not limited to tangible embodiments. Such computer-readable media is being defined as means that can transmit or transport the program for use by or in connection with the instruction execution system, apparatus, or device. The computer readable medium may include computer readable information in a transitory state medium (signal).

In view of such, the claims are not limited to statutory subject matter and are therefore non-statutory.

Claims 17 – 21 and 23 – 27 are necessarily rejected as being dependent upon

the rejection of claims 16 and 22..

Double Patenting

The nonstatutory double patenting rejection is based on a judicially created doctrine grounded in public policy (a policy reflected in the statute) so as to prevent the unjustified or improper timewise extension of the "right to exclude" granted by a patent and to prevent possible harassment by multiple assignees. A nonstatutory obviousness-type double patenting rejection is appropriate where the conflicting claims are not identical, but at least one examined application claim is not patentably distinct from the reference claim(s) because the examined application claim is either anticipated by, or would have been obvious over, the reference claim(s). See, e.g., *In re Berg*, 140 F.3d 1428, 46 USPQ2d 1226 (Fed. Cir. 1998); *In re Goodman*, 11 F.3d 1046, 29 USPQ2d 2010 (Fed. Cir. 1993); *In re Longi*, 759 F.2d 887, 225 USPQ 645 (Fed. Cir. 1985); *In re Van Ornum*, 686 F.2d 937, 214 USPQ 761 (CCPA 1982); *In re Vogel*, 422 F.2d 438, 164 USPQ 619 (CCPA 1970); and *In re Thorington*, 418 F.2d 528, 163 USPQ 644 (CCPA 1969). .

A timely filed terminal disclaimer in compliance with 37 CFR 1.321(c) or 1.321(d) may be used to overcome an actual or provisional rejection based on a nonstatutory double patenting ground provided the conflicting application or patent either is shown to be commonly owned with this application, or claims an invention made as a result of activities undertaken within the scope of a joint research agreement.

Effective January 1, 1994, a registered attorney or agent of record may sign a terminal disclaimer. A terminal disclaimer signed by the assignee must fully comply with 37 CFR 3.73(b).

Claims 1, 7, 16, and 22 are rejected on the ground of nonstatutory obviousness-type double patenting as being unpatentable over claims 1, 23, and 46 of U.S. Patent No. 7,146,644. Although the conflicting claims are not identical, they are not patentably distinct from each other because they are both performing the steps of filtering, storing, extracting, reconstructing, and securing data in a "computer system" of the present Application, as opposed to " computer network " of the US Patent 7,146, 644, is an obvious variation in the art for the purpose of achieving the same end results.

Claims 1, 7, 16, and 22 are rejected on the ground of nonstatutory obviousness-type double patenting as being unpatentable over claims 1, 23, and 46 of U.S. Patent No. 7,140,044. Although the conflicting claims are not identical, they are not patentably distinct from each other because they are both performing the steps of filtering, storing, extracting, reconstructing, and securing data in a "computer system" of the present Application, as opposed to " computer network " of the US Patent 7,140, 044, is an obvious variation in the art for the purpose of achieving the same end results.

Claims 1, 7, 16, and 22 are rejected on the ground of nonstatutory obviousness-type double patenting as being unpatentable over claims 1, 12, 20, 26, 31, 36, 41, 51, 58, 64, 69, 73, 77, and 82 of U.S. Patent No. 7,191,252. Although the conflicting claims are not identical, they are not patentably distinct from each other because they both performing the steps of filtering, storing, extracting, reconstructing, and substituting a " data " of the present Application as opposed to an " email data" of the U.S. Patent No. 7,191,252, is an obvious variation in the art for the purpose of achieving the same end results.

Claims 1, 7, 16, and 22 are provisionally rejected on the ground of nonstatutory obviousness-type double patenting as being unpatentable over claims 1, 15, 27, 41, 53, and 67 of copending Application No. 10277196. Although the conflicting claims are not identical, they are not patentably distinct from each other because they are both performing the steps of filtering, storing, extracting, reconstructing, and substituting an "

extracted data " of the present Application, as opposes to a "subset of extracted data " would be an obvious variation in the art for the purpose of achieving the same end results.

This is a provisional obviousness-type double patenting rejection because the conflicting claims have not in fact been patented.

The Examiner has rejected only the independent claims because of numerous applications that qualify as "nonstatutory obviousness-type double patenting" rejection.

Contact Information

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Yves Dalencourt whose telephone number is (571) 272-3998. The examiner can normally be reached on M-TH 7:30AM - 6:00PM.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Ario Etienne can be reached on (571) 272 4001. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

September 10, 2006

YVES DALENCOURT
PRIMARY EXAMINER
TECHNOLOGY CENTER 2100

Notice of References Cited	Application/Control No. 10/155,192	Applicant(s)/Patent Under Reexamination REDLICH ET AL.	
	Examiner Yves Dalencourt	Art Unit 2157	Page 1 of 1

U.S. PATENT DOCUMENTS

*		Document Number Country Code-Number-Kind Code	Date MM-YYYY	Name	Classification
*	A	US-7,140,044	11-2006	Redlich et al.	726/27
*	B	US-7,146,644	12-2006	Redlich et al.	726/27
*	C	US-7,191,252	03-2007	Redlich et al.	709/246
	D	US-			
	E	US-			
	F	US-			
	G	US-			
	H	US-			
	I	US-			
	J	US-			
	K	US-			
	L	US-			
	M	US-			

FOREIGN PATENT DOCUMENTS

*		Document Number Country Code-Number-Kind Code	Date MM-YYYY	Country	Name	Classification
	N					
	O					
	P					
	Q					
	R					
	S					
	T					

NON-PATENT DOCUMENTS

*		Include as applicable: Author, Title Date, Publisher, Edition or Volume, Pertinent Pages)
	U	
	V	
	W	
	X	

*A copy of this reference is not being furnished with this Office action. (See MPEP § 707.05(a).)
Dates in MM-YYYY format are publication dates. Classifications may be US or foreign.

PATENTS

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re Application of Ron M. Redlich, et al.

Serial No. 10/155,192 Examiner Dalencourt
Confirm no. 9510
Filed: May 23, 2002 Group 2157

For: Data Security System and Method With Parsing and Dispersion Techniques

AMENDMENT

Mail Stop Amendment
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Sir:

This amendment responds to the Office Action dated September 20, 2007. Please charge Deposit Account no. 03-1231 for the $65.00 fee for the Terminal Disclaimer. If the fee is inadequate, please debit or credit Deposit Account no. 03-1231. Submitted herewith are:

a. transmittal (1 pg., p. 1)

b. claims (6 pgs, pp. 2 - 7)

c. remarks (3 pgs, pp. 8 - 10)

d. Terminal Disclaimer (2 pgs.)

[Page left blank]

IN THE CLAIMS

1. (Previously Amended) A method of securing data having one or more security sensitive words, characters or icons in a computer system with a plurality of physically distinguishable memories respectively designated as a remainder store and a plurality of physically distinct extract stores corresponding to a plurality of security levels for respective subsets of said security sensitive words, characters or icons comprising:

extracting said security sensitive words, characters or icons from said data to obtain extracted data and remainder data therefrom;

storing said extracted data and said remainder data in said respective extract stores corresponding to an associated security level and said remainder store, respectively; and,

permitting full or partial reconstruction of said data via said extracted data from respective extract stores and remainder data only in the presence of a predetermined security clearance corresponding to said associated security level for said respective extract stores.

2. (original) A method as claimed in claim 1 including filtering said data while extracting.

3. (previously amended) A method as claimed in claim 2 including defining a plurality of filters for said respective subsets of security sensitive words, characters or icons and the method includes one of storing or destroying one or more of said plurality of filters after storing said extracted data.

4. (original) A method as claimed in claim 3 including encrypting one or both of said extracted data and said remainder data.

5. (original) A method as claimed in claim 4 wherein the step of permitting reconstruction includes decrypting one or both of said extracted data and remainder data.

6. (previously amended) A method as claimed in claim 1 including establishing a plurality of security levels each with a respective security clearance, each said subset of said security sensitive words, characters or icons being correlated with said plurality of security levels and the step of permitting reconstruction including the step of permitting a plurality of partial reconstructions of said data in the presence of respective ones of said plurality of security clearance levels.

7. (Previously Amended) A method of securing data with a parsing algorithm via a computer system having physically distinguishable memories designated as a remainder store and a plurality of physically distinct extract stores corresponding to a plurality of security levels for respective subsets of said security sensitive words, characters or icons comprising:

parsing said data to obtain extracted data for respective associated security levels and remainder data therefrom;

storing said extracted data and said remainder data in said extract stores and said remainder store, respectively; and,

permitting full or partial reconstruction of said data via said extracted data from said respective extract stores and remainder data only in the presence of a predetermined security clearance corresponding to said associated security level for said respective extract stores.

8. (original) A method as claimed in claim 7 wherein the step of parsing parses the data granularly.

9. (original) A method as claimed in claim 7 wherein the step of parsing uses said parsing algorithm which is based on one or more of a bit count, a word, a word count, a page count, a line count, a paragraph count, an identifiable document characteristic, an identifiable word characteristic,

an identifiable icon characteristic, an identifiable data object characteristic, capital letters, italics, and underline.

10. (original) A method as claimed in claim 7 wherein said data is financial data and the step of parsing uses said parsing algorithm based upon one of characters in said data and character count in said data.

11. (Previously Amended) A method as claimed in claim 7 including establishing a plurality of security levels each with a respective security clearance, permitting increasingly greater reconstruction if said data being correlated with increasingly higher levels of said plurality of security levels includes greater security level clearances, and the step of permitting reconstruction including the step of permitting a plurality of partial reconstructions of said data in the presence of respective ones of said plurality of security clearance levels.

12. (original) A method as claimed in claim 7 including a further data security step of one from the group of storing said data without encryption, storing said data with encryption and destroying said data; said further data security step being independent of said parsing, storing and permitting reconstruction.

13. (original) A method as claimed in claim 1 including displaying an advertisement prior to one or the other or both of said steps of extracting and permitting reconstruction.

14. (original) A method as claimed in claim 13 including providing a hyperlink to a vendor's web site with said advertisement.

15. (original) A method as claimed in claim 1 including assessing a charge for extracting, storing or permitting reconstruction.

16. (currently amended) A computer readable medium encoded with ~~containing~~ programming instructions for securing data having one or more security sensitive words, characters or icons in a computer system with a plurality physically distinguishable memories respectively designated as a remainder store and a plurality of physically distinct extract stores corresponding to a plurality of security levels for respective subsets of said security sensitive words, characters or icons comprising:

extracting said security sensitive words, characters or icons from said data to obtain extracted data and remainder data therefrom;

storing said extracted data and said remainder data in said respective extract stores corresponding to an associated security level and said remainder store, respectively; and,

permitting full or partial reconstruction of said data via said extracted data from respective extract stores and remainder data only in the presence of a predetermined security clearance corresponding to said associated security level for said respective extract stores.

17. (original) A medium with programming instructions as claimed in claim 16 including filtering said data while extracting.

18. (Previously Amended) A medium with programming instructions as claimed in claim 16 including defining a plurality of filters for respective subsets of security sensitive words, characters or icons and includes one of storing or destroying one or more of said plurality of filters after storing said extracted data.

19. (original) A medium with programming instructions as claimed in claim 18 including encrypting one or both of said extracted data and said remainder data.

20. (original) A medium with programming instructions as claimed in claim 19 wherein the step of permitting reconstruction includes decrypting one or both of said extracted data and remainder data.

21. (Previously Amended) A medium with programming instructions as claimed in claim 16 including establishing a plurality of security levels each with a respective security clearance, each said subset of said security sensitive words, characters or icons being correlated with said plurality of security levels and the step of permitting reconstruction including the step of permitting a plurality of partial reconstructions of said data in the presence of respective ones of said plurality of security clearance levels.

22. (currently amended) A computer readable medium encoded with ~~containing~~ programming instructions for securing data with a parsing algorithm via a computer system having physically distinguishable memories designated as a remainder store and a plurality of physically distinct extract stores corresponding to a plurality of security levels for respective subsets of said security sensitive words, characters or icons comprising:

parsing said data to obtain extracted data for respective associated security levels and remainder data therefrom;

storing said extracted data and said remainder data in said respective extract stores and said remainder store, respectively; and,

permitting full or partial reconstruction of said data via said extracted data from said respective extract stores and remainder data only in the presence of a predetermined security clearance corresponding to said associated security level for said respective extract stores.

23. (original) A medium with programming instructions as claimed in claim 22 wherein the step of parsing parses the data granularly.

24. (original) A medium with programming instructions as claimed in claim 22 wherein the step of parsing uses said parsing algorithm which is based on one or more of a bit count, a word, a word count, a page count, a line count, a paragraph count, an identifiable document characteristic, an identifiable word characteristic, an identifiable icon characteristic, an identifiable data object characteristic, capital letters, italics, and underline.

25. (original) A medium with programming instructions as claimed in claim 22 wherein said data is financial data and the step of parsing uses said parsing algorithm based upon one of characters in said data and character count in said data.

26. (Previously Amended) A medium with programming instructions as claimed in claim 22 including establishing a plurality of security levels each with a respective security clearance, permitting increasingly greater reconstruction if said data being correlated with increasingly higher levels of said plurality of security levels includes greater security level clearances, and the step of permitting reconstruction including the step of permitting a plurality of partial reconstructions of said data in the presence of respective ones of said plurality of security clearance levels.

27. (original) A medium as claimed in claim 22 including a further data security step of one from the group of storing said data without encryption, storing said data with encryption and destroying said data; said further data security step being independent of said parsing, storing and permitting reconstruction.

REMARKS

This amendment responds to the Office Action dated September 20, 2007. In the Office Action, the examiner rejected claims 16 -27 indicating that the claims were not proper under section 101 of the Patent Statutes. These claims have been amended to recite that the programmable medium is "encoded with" the programming instructions. The Patent Office has earlier indicated that such language clarifies and corrects the claims to comply with Section 101.

The examiner also requested a Terminal Disclaimer.

In the Office Action, the examiner also issued a double patenting rejection based upon U.S. Patent No. 7,146,644 (Serial No. 10/008,218, filed 12/06/2001); U.S. Patent No. 7,191,252 (Serial No. 10/155,525, filed 05/23/2002); U.S. Patent No. 7,140,044 (Serial No. 10/008,209, filed 12/06/2001); and Serial Nos. 10/277,196 (filed 12/31/2002). Applicant submits, under protest, a Terminal Disclaimer herewith.

> If a "provisional" nonstatutory obviousness-type double patenting (ODP) rejection is the only rejection remaining in the earlier filed of the two pending applications, while the later-filed application is rejectable on other grounds, the examiner should withdraw that rejection and permit the earlier-filed application to issue as a patent without a terminal disclaimer.

MPEP 804 (I)(B)(1), p. 800-17.

> The filing of a terminal disclaimer to obviate a rejection based on nonstatutory double patenting is not an admission of the propriety of the rejection. *Quad Environmental Technologies Corp. v. Union Sanitary District*, 946 F.2d 870, 20 USPQ2d 1392 (Fed. Cir. 1991). The court indicated that the "filing of a terminal disclaimer simply serves the statutory function of removing the rejection of double patenting, and raises neither a presumption nor estoppel on the merits of the rejection."

MPEP 804.02(II), p. 800-32.

Applicant is submitting this terminal disclaimer for this case in order to expedite prosecution. Applicant disagrees that the Examiner adequately explained the link or combination of the present

claims with those claims in one or more of U.S. Patent No. 7,146,644 (Serial No. 10/008,218, filed 12/06/2001); U.S. Patent No. 7,191,252 (Serial No. 10/155,525, filed 05/23/2002); U.S. Patent No. 7,140,044 (Serial No. 10/008,209, filed 12/06/2001); and Serial Nos. 10/277,196 (filed 12/31/2002) with the cited prior art.

> Since the analysis employed in an obviousness-type double patenting determination parallels the guidelines for a 35 U.S.C. 103(a) rejection, the factual inquiries set forth in *Graham v. John Deere Co.,* 383 U.S. 1, 148 USPQ 459 (1966), that are applied for establishing a background for determining obviousness under 35 U.S.C. 103 are employed when making an obvious-type double patenting analysis. These factual inquiries are summarized as follows: (A) Determine the scope and content of a patent claim relative to a claim in the application at issue; (B) Determine the differences between the scope and content of the patent claim as determined in (A) and the claim in the application at issue; (C) Determine the level of ordinary skill in the pertinent art; and (D) Evaluate any objective indicia of nonobviousness.
>
> The conclusion of obviousness-type double patenting is made in light of these factual determinations. Any obviousness-type double patenting rejection should make clear: (A) The differences between the inventions defined by the conflicting claims — a claim in the patent compared to a claim in the application; and (B) The reasons why a person of ordinary skill in the art would conclude that the invention defined in the claim at issue would have been< an obvious variation of the invention defined in a claim in the patent.

MPEP § 804(II)(B)(1) p. 800-21

It is respectfully submitted that the Examiner accept the enclosed Terminal Disclaimer and withdraw the double patenting rejection with respect to U.S. Patent No. 7,146,644 (Serial No. 10/008,218, filed 12/06/2001); U.S. Patent No. 7,191,252 (Serial No. 10/155,525, filed 05/23/2002); U.S. Patent No. 7,140,044 (Serial No. 10/008,209, filed 12/06/2001); and Serial Nos. 10/277,196 (filed 12/31/2002).

Applicant respectfully requests that the examiner withdraw the rejections and approve the patentability of claims 1 - 27.

Respectfully submitted,
By /Robert Kain/
Robert C. Kain, Jr.
Reg. No. 30,648

Fleit, Kain, Gibbons, Gutman, Bongini & Bianco, P.L.
750 Southeast Third Avenue, Suite 100
Fort Lauderdale, FL 33316-1153
Telephone: 954-768-9002
Facsimile: 954-768-0158

Certificate of e-Filing

I hereby certify that this correspondence is being e-Filed with the Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450 on October 9, 2007.

/Robert Kain/
Robert C. Kain, Jr.
Reg. No. 30,648

F:\RCK\CLIENTS\Redlich\Patents\6851-02cip3-amdt-4th-100907.wpd

PATENTS

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re Application of Ron M. Redlich, et al. Confirm no. 9510
Serial No. 10/155,192 Examiner Dalencourt
Filed: May 23, 2002 Group 2157
For: Data Security System and Method With Parsing and Dispersion Techniques

TERMINAL DISCLAIMER

Commissioner for Patents
Alexandria, VA 22313-1450
Sir:

Petitioner, Digital Doors, Inc., a Florida corporation, having its principal place of business at 4201 Collins Avenue, Suite 2103, Miami Beach, FL 33140, represents that it is the owner by assignment of all rights to (i) the captioned application and (ii) U.S. Patent No. 7,146,644 (Serial No. 10/008,218, filed 12/06/2001); U.S. Patent No. 7,191,252 (Serial No. 10/155,525, filed 05/23/2002); U.S. Patent No. 7,140,044 (Serial No. 10/008,209, filed 12/06/2001); and Serial Nos. 10/277,196 (filed 12/31/2002).

The captioned application was assigned to Digital Doors, Inc. and the assignment was filed with the original application. The evidentiary document accompanying or referred to in the instant terminal disclaimer has been reviewed by the undersigned and it is certified that to the best of assignee, Digital Doors, Inc.'s knowledge and belief, title is in the assignee.

Petitioner, Digital Doors, Inc., hereby disclaims the terminal part of any patent granted on the above identified application, which would extend beyond the expiration date of any patent granted on U.S. Patent No. 7,146,644 (Serial No. 10/008,218, filed 12/06/2001); U.S. Patent No. 7,191,252 (Serial No. 10/155,525, filed 05/23/2002); U.S. Patent No. 7,140,044 (Serial No. 10/008,209, filed 12/06/2001); and Serial Nos. 10/277,196 (filed 12/31/2002), and, except as provided below, hereby agrees that any patent so granted on the above identified application shall

be enforceable only for and during such period and that the listed patents will remain commonly owned. Petitioner, Digital Doors, Inc., is a small entity.

Petitioner does not disclaim any terminal part of any patent granted on the above identified application prior to the expiration date of the full statutory term of any patent granted on U.S. Patent No. 7,146,644 (Serial No. 10/008,218, filed 12/06/2001); U.S. Patent No. 7,191,252 (Serial No. 10/155,525, filed 05/23/2002); U.S. Patent No. 7,140,044 (Serial No. 10/008,209, filed 12/06/2001); and Serial Nos. 10/277,196 (filed 12/31/2002) in the event that any latter patent: expires for failure to pay a maintenance fee, is held unenforceable, is found invalid, is statutorily disclaimed in whole or terminally disclaimed under 37 C.F.R. §1.321(a), has all claims canceled by a reexamination certificate, is reissued, or is otherwise terminated prior to the expiration of its statutory term as presently shortened by any terminal disclaimer, except for the separation of legal title stated above.

Petitioner expressly reserves and does not waive the right to extend the term of any patent granted on U.S. Patent No. 7,146,644 (Serial No. 10/008,218, filed 12/06/2001); U.S. Patent No. 7,191,252 (Serial No. 10/155,525, filed 05/23/2002); U.S. Patent No. 7,140,044 (Serial No. 10/008,209, filed 12/06/2001); and Serial Nos. 10/277,196 (filed 12/31/2002), or any patent granted on the above identified application under any statute which permits such an extension of term.

I hereby declare that all statements made herein of our own knowledge are true, and that all statements made on information and belief are believed to be true; and further, that these statements were made with the knowledge that willful false statements, and the like so made, are punishable by fine or imprisonment, or both, under Section 1001 of Title 18 of the United States Code, and that such willful false statements may jeopardize the validity of the application or any patent issuing thereon.

Dated: __October 9, 2007

Digital Doors, Inc.
By: __/Robert C. Kain/ _____
Robert C. Kain, Jr. Reg. No. 30,648
Fleit, Kain, Gibbons, Gutman, Bongini & Bianco, P.L.
750 Southeast Third Avenue, Suite 100
Ft. Lauderdale, Florida 33316-1153
(954) 768-9002
Attorney of Record for Petitioner

F:\RCK\CLIENTS\Redlich\6851-02-cip3-Term-Discl-100907.wpd

Electronic Acknowledgement Receipt

EFS ID:	2293632
Application Number:	10155192
International Application Number:	
Confirmation Number:	9510
Title of Invention:	Data security system and method with parsing and dispersion techniques
First Named Inventor/Applicant Name:	Ron M. Redlich
Customer Number:	4897
Filer:	Robert C. Kain
Filer Authorized By:	
Attorney Docket Number:	6851-2-CIP-3
Receipt Date:	09-OCT-2007
Filing Date:	23-MAY-2002
Time Stamp:	11:41:18
Application Type:	Utility under 35 USC 111(a)

Payment information:

Submitted with Payment	no

File Listing:

Document Number	Document Description	File Name	File Size(Bytes) /Message Digest	Multi Part /.zip	Pages (if appl.)
1	Amendment - After Non-Final Rejection	6851-02cip3-amdt-4th-100907-trans.pdf	15017 1e088afd46410a3a1c8ce36a4d8ced4b44446fe9	no	1

Warnings:

Information:					
2	Claims	6851-02cip3-amdt-4th-100907-clms.pdf	23044 ca86bb09c333d83bfa120615129accb9567518af	no	6
Warnings:					
Information:					
3	Applicant Arguments/Remarks Made in an Amendment	6851-02cip3-amdt-4th-100907-rem.pdf	29531 3999e7d5eb8898a1e571d674da8ab16f1929a106	no	3
Warnings:					
Information:					
4	Terminal Disclaimer Filed	6851-02-cip3-Term-Discl-100907.pdf	18654 7b0191b4350c95e8ef250109728423aabeefd47f	no	2
Warnings:					
Information:					
		Total Files Size (in bytes):	86246		

This Acknowledgement Receipt evidences receipt on the noted date by the USPTO of the indicated documents, characterized by the applicant, and including page counts, where applicable. It serves as evidence of receipt similar to a Post Card, as described in MPEP 503.

New Applications Under 35 U.S.C. 111
If a new application is being filed and the application includes the necessary components for a filing date (see 37 CFR 1.53(b)-(d) and MPEP 506), a Filing Receipt (37 CFR 1.54) will be issued in due course and the date shown on this Acknowledgement Receipt will establish the filing date of the application.

National Stage of an International Application under 35 U.S.C. 371
If a timely submission to enter the national stage of an international application is compliant with the conditions of 35 U.S.C. 371 and other applicable requirements a Form PCT/DO/EO/903 indicating acceptance of the application as a national stage submission under 35 U.S.C. 371 will be issued in addition to the Filing Receipt, in due course.

New International Application Filed with the USPTO as a Receiving Office
If a new international application is being filed and the international application includes the necessary components for an international filing date (see PCT Article 11 and MPEP 1810), a Notification of the International Application Number and of the International Filing Date (Form PCT/RO/105) will be issued in due course, subject to prescriptions concerning national security, and the date shown on this Acknowledgement Receipt will establish the international filing date of the application.

PTO/SB/06 (12-04)
Approved for use through 7/31/2006. OMB 0651-0032
U.S. Patent and Trademark Office; U.S. DEPARTMENT OF COMMERCE
Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it displays a valid OMB control number.

PATENT APPLICATION FEE DETERMINATION RECORD Substitute for Form PTO-875	Application or Docket Number 10155192	Filing Date: 05/23/2002	☐ To be Mailed

APPLICATION AS FILED – PART I

	(Column 1)	(Column 2)	SMALL ENTITY ☒		OR	OTHER THAN SMALL ENTITY	
FOR	NUMBER FILED	NUMBER EXTRA	RATE ($)	FEE ($)		RATE ($)	FEE ($)
☐ BASIC FEE (37 CFR 1.16(a), (b), or (c))	N/A	N/A	N/A			N/A	
☐ SEARCH FEE (37 CFR 1.16(k), (i), or (m))	N/A	N/A	N/A			N/A	
☐ EXAMINATION FEE (37 CFR 1.16(o), (p), or (q))	N/A	N/A	N/A			N/A	
TOTAL CLAIMS (37 CFR 1.16(i))	minus 20 =	*	X $25 =		OR	X $50 =	
INDEPENDENT CLAIMS (37 CFR 1.16(h))	minus 3 =	*	X $100 =			X $200 =	
☐ APPLICATION SIZE FEE (37 CFR 1.16(s))	If the specification and drawings exceed 100 sheets of paper, the application size fee due is $250 ($125 for small entity) for each additional 50 sheets or fraction thereof. See 35 U.S.C. 41(a)(1)(G) and 37 CFR 1.16(s).						
☐ MULTIPLE DEPENDENT CLAIM PRESENT (37 CFR 1.16(j))			+ $180	0		+$360	
* If the difference in column 1 is less than zero, enter "0" in column 2.			TOTAL			TOTAL	

APPLICATION AS AMENDED – PART II

		(Column 1)		(Column 2)	(Column 3)	SMALL ENTITY		OR	OTHER THAN SMALL ENTITY	
AMENDMENT A	10.17.06	CLAIMS REMAINING AFTER AMENDMENT		HIGHEST NUMBER PREVIOUSLY PAID FOR	PRESENT EXTRA	RATE ($)	ADDITIONAL FEE ($)		RATE ($)	ADDITIONAL FEE ($)
	Total (37 CFR 1.16(i))	* 27	Minus	** 27	= 0	X $25 =	0	OR	X $50=	
	Independent (37 CFR 1.16(h))	* 4	Minus	*** 4	= 0	X $100 =	0	OR	X $200=	
	☐ Application Size Fee (37 CFR 1.16(s))									
	☐ FIRST PRESENTATION OF MULTIPLE DEPENDENT CLAIM (37 CFR 1.16(j))							OR		
						TOTAL ADD'L FEE		OR	TOTAL ADD'L FEE	

		(Column 1)		(Column 2)	(Column 3)					
AMENDMENT B	10/8/10	CLAIMS REMAINING AFTER AMENDMENT		HIGHEST NUMBER PREVIOUSLY PAID FOR	PRESENT EXTRA	RATE ($)	ADDITIONAL FEE ($)		RATE ($)	ADDITIONAL FEE ($)
	Total (37 CFR 1.16(i))	* 27	Minus	** 27	=	X $25 =		OR	X $50 =	
	Independent (37 CFR 1.16(h))	* 7	Minus	*** 4	=	X $100 =		OR	X $200 =	
	☐ Application Size Fee (37 CFR 1.16(s))									
	☐ FIRST PRESENTATION OF MULTIPLE DEPENDENT CLAIM (37 CFR 1.16(j))							OR		
	CALCULATE					TOTAL ADD'L FEE		OR	TOTAL ADD'L FEE	

* If the entry in column 1 is less than the entry in column 2, write "0" in column 3.
** If the "Highest Number Previously Paid For" IN THIS SPACE is less than 20, enter "20".
*** If the "Highest Number Previously Paid For" IN THIS SPACE is less than 3, enter "3".
The "Highest Number Previously Paid For" (Total or Independent) is the highest number found in the appropriate box in column 1.

Legal Instrument Examiner:
Tammy Acree

This collection of information is required by 37 CFR 1.16. The information is required to obtain or retain a benefit by the public which is to file (and by the USPTO to process) an application. Confidentiality is governed by 35 U.S.C. 122 and 37 CFR 1.14. This collection is estimated to take 12 minutes to complete, including gathering, preparing, and submitting the completed application form to the USPTO. Time will vary depending upon the individual case. Any comments on the amount of time you require to complete this form and/or suggestions for reducing this burden, should be sent to the Chief Information Officer, U.S. Patent and Trademark Office, U.S. Department of Commerce, P.O. Box 1450, Alexandria, VA 22313-1450. DO NOT SEND FEES OR COMPLETED FORMS TO THIS ADDRESS. **SEND TO: Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450.**

If you need assistance in completing the form, call 1-800-PTO-9199 and select option 2.

Document code: WFEE

United States Patent and Trademark Office
Sales Receipt for Accounting Date: 10/25/2007

SCOTTON	SALE	#00000001	Mailroom Dt:	10/09/2007	031231	10155192
	01	FC : 2814		65.00 DA		

Application Number	Application/Control No.	Applicant(s)/Patent under Reexamination	
	10/155,192	REDLICH ET AL.	

Document Code - DISQ	Internal Document – DO NOT MAIL

TERMINAL DISCLAIMER	☒ APPROVED	☐ DISAPPROVED
Date Filed : 10/9/07	This patent is subject to a Terminal Disclaimer	

Approved/Disapproved by:
BRIAN

U.S. Patent and Trademark Office

Freeform Search

Database: US Pre-Grant Publication Full-Text Database; US Patents Full-Text Database; US OCR Full-Text Database; EPO Abstracts Database; JPO Abstracts Database; Derwent World Patents Index; IBM Technical Disclosure Bulletins

Term: 14 and L5

Display: 10 **Documents in Display Format:** - **Starting with Number** 1

Generate: ○ Hit List ◉ Hit Count ○ Side by Side ○ Image

Search Clear Interrupt

Search History

DATE: Saturday, November 10, 2007 **Purge Queries** Printable Copy Create Case

Set Name side by side	Query	Hit Count	Set Name result set
DB=PGPB,USPT,EPAB,JPAB,DWPI,TDBD; PLUR=YES; OP=OR			
L6	l4 and L5	21	L6
L5	l1 and L2	21	L5
L4	(restor$3 or reconstruct$3) adj7 (data or message$1 or email$1 or content)	62406	L4
L2	(extract$3 or censor$3 or delet$3) adj7 (word$1 or character$1)	28512	L2
L1	secur$3 near3 data and extract$3 near3 security near3 sensitive near3 word$1	21	L1

END OF SEARCH HISTORY

Refine Search

Search Results -

Term	Documents
(11 AND 3).PGPB,USPT,EPAB,JPAB,DWPI,TDBD.	7
(L3 AND L11).PGPB,USPT,EPAB,JPAB,DWPI,TDBD.	7

Database: US Pre-Grant Publication Full-Text Database; US Patents Full-Text Database; US OCR Full-Text Database; EPO Abstracts Database; JPO Abstracts Database; Derwent World Patents Index; IBM Technical Disclosure Bulletins

Search: L12

Refine Search | Recall Text | Clear | Interrupt

Search History

DATE: Monday, November 12, 2007 **Purge Queries** Printable Copy Create Case

Set Name side by side	Query	Hit Count	Set Name result set
DB=PGPB,USPT,EPAB,JPAB,DWPI,TDBD; PLUR=YES; OP=OR			
L12	l3 and L11	7	L12
L11	l2 and L10	26	L11
L10	713/166,189.ccls.	1880	L10
L9	l3 and L6	128	L9
L8	l3 and L7	71	L8
L7	726/27.ccls.	840	L7
L6	709/244,246.ccls.	2173	L6
L5	L3 and L4	21	L5
L4	L1 and L2	21	L4
L3	(restor$3 or reconstruct$3) adj7 (data or message$1 or email$1 or content)	62431	L3
L2	(extract$3 or censor$3 or delet$3) adj7 (word$1 or character$1)	28529	L2

L1	secur$3 near3 data and extract$3 near3 security near3 sensitive near3 word$1	21	L1

END OF SEARCH HISTORY

IEEE Xplore® RELEASE 2.4

Home | Login | Logout | Access Information | Alerts | Purchase History | Cart |

Welcome United States Patent and Trademark Office

Search Session History

BROWSE SEARCH IEEE XPLORE GUIDE

Mon, 12 Nov 2007, 2:44:59 PM EST

Edit an existing query or compose a new query in the Search Query Display.

Select a search number (#) to:

- Add a query to the Search Query Display
- Combine search queries using AND, OR, or NOT
- Delete a search
- Run a search

Search Query Display

[redacted]

Recent Search Queries

#1	((secur* <near> data and extract* <near> security <near> security <near> sensitive <near> word*)<in>metadata)
#2	((secur* <near> data and extract* <near> security <near> security <near> sensitive <near> word*)<in>metadata)
#3	(secur* <near> data and extract* <near> (word and character) <IN>metadata)
#4	((restor* or reconstruct*) <near> (data or message or email or e-mail or content)<IN>metadata)
#5	((extract* or censor* or delet*) <near> (word or character) <IN>metadata)
#6	((permit* or allow*) <near> (full or partial) <near> reconstruction<IN>metadata)

[redacted]

Help Contact Us Privacy &

Indexed by Inspec®

© Copyright 2007 IEEE –

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

NOTICE OF ALLOWANCE AND FEE(S) DUE

4897 7590 11/15/2007

ROBERT C. KAIN, JR.
750 SOUTHEAST THIRD AVENUE
SUITE 100
FT LAUDERDALE, FL 333161153

EXAMINER	
DALENCOURT, YVES	
ART UNIT	PAPER NUMBER
2157	

DATE MAILED: 11/15/2007

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/155,192	05/23/2002	Ron M. Redlich	6851-2-CIP-3	9510

TITLE OF INVENTION: DATA SECURITY SYSTEM AND METHOD WITH PARSING AND DISPERSION TECHNIQUES

APPLN. TYPE	SMALL ENTITY	ISSUE FEE DUE	PUBLICATION FEE DUE	PREV. PAID ISSUE FEE	TOTAL FEE(S) DUE	DATE DUE
nonprovisional	YES	$720	$300	$0	$1020	02/15/2008

THE APPLICATION IDENTIFIED ABOVE HAS BEEN EXAMINED AND IS ALLOWED FOR ISSUANCE AS A PATENT. PROSECUTION ON THE MERITS IS CLOSED. THIS NOTICE OF ALLOWANCE IS NOT A GRANT OF PATENT RIGHTS. THIS APPLICATION IS SUBJECT TO WITHDRAWAL FROM ISSUE AT THE INITIATIVE OF THE OFFICE OR UPON PETITION BY THE APPLICANT. SEE 37 CFR 1.313 AND MPEP 1308.

THE ISSUE FEE AND PUBLICATION FEE (IF REQUIRED) MUST BE PAID WITHIN THREE MONTHS FROM THE MAILING DATE OF THIS NOTICE OR THIS APPLICATION SHALL BE REGARDED AS ABANDONED. THIS STATUTORY PERIOD CANNOT BE EXTENDED. SEE 35 U.S.C. 151. THE ISSUE FEE DUE INDICATED ABOVE DOES NOT REFLECT A CREDIT FOR ANY PREVIOUSLY PAID ISSUE FEE IN THIS APPLICATION. IF AN ISSUE FEE HAS PREVIOUSLY BEEN PAID IN THIS APPLICATION (AS SHOWN ABOVE), THE RETURN OF PART B OF THIS FORM WILL BE CONSIDERED A REQUEST TO REAPPLY THE PREVIOUSLY PAID ISSUE FEE TOWARD THE ISSUE FEE NOW DUE.

HOW TO REPLY TO THIS NOTICE:

I. Review the SMALL ENTITY status shown above.

If the SMALL ENTITY is shown as YES, verify your current SMALL ENTITY status:

A. If the status is the same, pay the TOTAL FEE(S) DUE shown above.

B. If the status above is to be removed, check box 5b on Part B - Fee(s) Transmittal and pay the PUBLICATION FEE (if required) and twice the amount of the ISSUE FEE shown above, or

If the SMALL ENTITY is shown as NO:

A. Pay TOTAL FEE(S) DUE shown above, or

B. If applicant claimed SMALL ENTITY status before, or is now claiming SMALL ENTITY status, check box 5a on Part B - Fee(s) Transmittal and pay the PUBLICATION FEE (if required) and 1/2 the ISSUE FEE shown above.

II. PART B - FEE(S) TRANSMITTAL, or its equivalent, must be completed and returned to the United States Patent and Trademark Office (USPTO) with your ISSUE FEE and PUBLICATION FEE (if required). If you are charging the fee(s) to your deposit account, section "4b" of Part B - Fee(s) Transmittal should be completed and an extra copy of the form should be submitted. If an equivalent of Part B is filed, a request to reapply a previously paid issue fee must be clearly made, and delays in processing may occur due to the difficulty in recognizing the paper as an equivalent of Part B.

III. All communications regarding this application must give the application number. Please direct all communications prior to issuance to Mail Stop ISSUE FEE unless advised to the contrary.

IMPORTANT REMINDER: Utility patents issuing on applications filed on or after Dec. 12, 1980 may require payment of maintenance fees. It is patentee's responsibility to ensure timely payment of maintenance fees when due.

PTOL-85 (Rev. 08/07) Approved for use through 08/31/2010.

PART B - FEE(S) TRANSMITTAL

Complete and send this form, together with applicable fee(s), to: Mail Mail Stop ISSUE FEE
Commissioner for Patents
P.O. Box 1450
Alexandria, Virginia 22313-1450
or Fax (571)-273-2885

INSTRUCTIONS: This form should be used for transmitting the ISSUE FEE and PUBLICATION FEE (if required). Blocks 1 through 5 should be completed where appropriate. All further correspondence including the Patent, advance orders and notification of maintenance fees will be mailed to the current correspondence address as indicated unless corrected below or directed otherwise in Block 1, by (a) specifying a new correspondence address; and/or (b) indicating a separate "FEE ADDRESS" for maintenance fee notifications.

CURRENT CORRESPONDENCE ADDRESS (Note: Use Block 1 for any change of address)

4897 7590 11/15/2007

ROBERT C. KAIN, JR.
750 SOUTHEAST THIRD AVENUE
SUITE 100
FT LAUDERDALE, FL 333161153

Note: A certificate of mailing can only be used for domestic mailings of the Fee(s) Transmittal. This certificate cannot be used for any other accompanying papers. Each additional paper, such as an assignment or formal drawing, must have its own certificate of mailing or transmission.

Certificate of Mailing or Transmission
I hereby certify that this Fee(s) Transmittal is being deposited with the United States Postal Service with sufficient postage for first class mail in an envelope addressed to the Mail Stop ISSUE FEE address above, or being facsimile transmitted to the USPTO (571) 273-2885, on the date indicated below.

(Depositor's name)
(Signature)
(Date)

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/155,192	05/23/2002	Ron M. Redlich	6851-2-CIP-3	9510

TITLE OF INVENTION: DATA SECURITY SYSTEM AND METHOD WITH PARSING AND DISPERSION TECHNIQUES

APPLN. TYPE	SMALL ENTITY	ISSUE FEE DUE	PUBLICATION FEE DUE	PREV. PAID ISSUE FEE	TOTAL FEE(S) DUE	DATE DUE
nonprovisional	YES	$720	$300	$0	$1020	02/15/2008

EXAMINER	ART UNIT	CLASS-SUBCLASS
DALENCOURT, YVES	2157	709-244000

1. Change of correspondence address or indication of "Fee Address" (37 CFR 1.363).

☐ Change of correspondence address (or Change of Correspondence Address form PTO/SB/122) attached.

☐ "Fee Address" indication (or "Fee Address" Indication form PTO/SB/47; Rev 03-02 or more recent) attached. **Use of a Customer Number is required.**

2. For printing on the patent front page, list

(1) the names of up to 3 registered patent attorneys or agents OR, alternatively,

(2) the name of a single firm (having as a member a registered attorney or agent) and the names of up to 2 registered patent attorneys or agents. If no name is listed, no name will be printed.

1 _____
2 _____
3 _____

3. ASSIGNEE NAME AND RESIDENCE DATA TO BE PRINTED ON THE PATENT (print or type)

PLEASE NOTE: Unless an assignee is identified below, no assignee data will appear on the patent. If an assignee is identified below, the document has been filed for recordation as set forth in 37 CFR 3.11. Completion of this form is NOT a substitute for filing an assignment.

(A) NAME OF ASSIGNEE (B) RESIDENCE: (CITY and STATE OR COUNTRY)

Please check the appropriate assignee category or categories (will not be printed on the patent) : ☐ Individual ☐ Corporation or other private group entity ☐ Government

4a. The following fee(s) are submitted:

☐ Issue Fee
☐ Publication Fee (No small entity discount permitted)
☐ Advance Order - # of Copies _____

4b. Payment of Fee(s): **(Please first reapply any previously paid issue fee shown above)**

☐ A check is enclosed.
☐ Payment by credit card. Form PTO-2038 is attached.
☐ The Director is hereby authorized to charge the required fee(s), any deficiency, or credit any overpayment, to Deposit Account Number _____ (enclose an extra copy of this form).

5. **Change in Entity Status** (from status indicated above)

☐ a. Applicant claims SMALL ENTITY status. See 37 CFR 1.27. ☐ b. Applicant is no longer claiming SMALL ENTITY status. See 37 CFR 1.27(g)(2).

NOTE: The Issue Fee and Publication Fee (if required) will not be accepted from anyone other than the applicant; a registered attorney or agent; or the assignee or other party in interest as shown by the records of the United States Patent and Trademark Office.

Authorized Signature _____ Date _____

Typed or printed name _____ Registration No. _____

This collection of information is required by 37 CFR 1.311. The information is required to obtain or retain a benefit by the public which is to file (and by the USPTO to process) an application. Confidentiality is governed by 35 U.S.C. 122 and 37 CFR 1.14. This collection is estimated to take 12 minutes to complete, including gathering, preparing, and submitting the completed application form to the USPTO. Time will vary depending upon the individual case. Any comments on the amount of time you require to complete this form and/or suggestions for reducing this burden, should be sent to the Chief Information Officer, U.S. Patent and Trademark Office, U.S. Department of Commerce, P.O. Box 1450, Alexandria, Virginia 22313-1450. DO NOT SEND FEES OR COMPLETED FORMS TO THIS ADDRESS. SEND TO: Commissioner for Patents, P.O. Box 1450, Alexandria, Virginia 22313-1450.

Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it displays a valid OMB control number.

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.
10/155,192	05/23/2002	Ron M. Redlich	6851-2-CIP-3	9510

4897 7590 11/15/2007

ROBERT C. KAIN, JR.
750 SOUTHEAST THIRD AVENUE
SUITE 100
FT LAUDERDALE, FL 333161153

EXAMINER	
DALENCOURT, YVES	
ART UNIT	PAPER NUMBER
2157	

DATE MAILED: 11/15/2007

Determination of Patent Term Adjustment under 35 U.S.C. 154 (b)
(application filed on or after May 29, 2000)

The Patent Term Adjustment to date is 873 day(s). If the issue fee is paid on the date that is three months after the mailing date of this notice and the patent issues on the Tuesday before the date that is 28 weeks (six and a half months) after the mailing date of this notice, the Patent Term Adjustment will be 873 day(s).

If a Continued Prosecution Application (CPA) was filed in the above-identified application, the filing date that determines Patent Term Adjustment is the filing date of the most recent CPA.

Applicant will be able to obtain more detailed information by accessing the Patent Application Information Retrieval (PAIR) WEB site (http://pair.uspto.gov).

Any questions regarding the Patent Term Extension or Adjustment determination should be directed to the Office of Patent Legal Administration at (571)-272-7702. Questions relating to issue and publication fee payments should be directed to the Customer Service Center of the Office of Patent Publication at 1-(888)-786-0101 or (571)-272-4200.

Notice of Allowability	Application No.	Applicant(s)	
	10/155,192	REDLICH ET AL.	
	Examiner	Art Unit	
	Yves Dalencourt	2157	

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

All claims being allowable, PROSECUTION ON THE MERITS IS (OR REMAINS) CLOSED in this application. If not included herewith (or previously mailed), a Notice of Allowance (PTOL-85) or other appropriate communication will be mailed in due course. **THIS NOTICE OF ALLOWABILITY IS NOT A GRANT OF PATENT RIGHTS.** This application is subject to withdrawal from issue at the initiative of the Office or upon petition by the applicant. See 37 CFR 1.313 and MPEP 1308.

1. ☒ This communication is responsive to *amendment filed on 10/09/2007*.

2. ☒ The allowed claim(s) is/are *1-27*.

3. ☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
   a) ☐ All b) ☐ Some* c) ☐ None of the:
      1. ☐ Certified copies of the priority documents have been received.
      2. ☐ Certified copies of the priority documents have been received in Application No. _____ .
      3. ☐ Copies of the certified copies of the priority documents have been received in this national stage application from the International Bureau (PCT Rule 17.2(a)).

   * Certified copies not received: _____.

Applicant has THREE MONTHS FROM THE "MAILING DATE" of this communication to file a reply complying with the requirements noted below. Failure to timely comply will result in ABANDONMENT of this application.
THIS THREE-MONTH PERIOD IS NOT EXTENDABLE.

4. ☐ A SUBSTITUTE OATH OR DECLARATION must be submitted. Note the attached EXAMINER'S AMENDMENT or NOTICE OF INFORMAL PATENT APPLICATION (PTO-152) which gives reason(s) why the oath or declaration is deficient.

5. ☐ CORRECTED DRAWINGS (as "replacement sheets") must be submitted.
   (a) ☐ including changes required by the Notice of Draftsperson's Patent Drawing Review (PTO-948) attached
      1) ☐ hereto or 2) ☐ to Paper No./Mail Date _____.
   (b) ☐ including changes required by the attached Examiner's Amendment / Comment or in the Office action of Paper No./Mail Date _____.

   Identifying indicia such as the application number (see 37 CFR 1.84(c)) should be written on the drawings in the front (not the back) of each sheet. Replacement sheet(s) should be labeled as such in the header according to 37 CFR 1.121(d).

6. ☐ DEPOSIT OF and/or INFORMATION about the deposit of BIOLOGICAL MATERIAL must be submitted. Note the attached Examiner's comment regarding REQUIREMENT FOR THE DEPOSIT OF BIOLOGICAL MATERIAL.

Attachment(s)

1. ☐ Notice of References Cited (PTO-892)
2. ☐ Notice of Draftperson's Patent Drawing Review (PTO-948)
3. ☐ Information Disclosure Statements (PTO/SB/08), Paper No./Mail Date _____
4. ☐ Examiner's Comment Regarding Requirement for Deposit of Biological Material
5. ☐ Notice of Informal Patent Application
6. ☐ Interview Summary (PTO-413), Paper No./Mail Date _____ .
7. ☐ Examiner's Amendment/Comment
8. ☐ Examiner's Statement of Reasons for Allowance
9. ☐ Other _____.

YVES DALENCOURT
PRIMARY EXAMINER
TECHNOLOGY CENTER 2100

DETAILED ACTION

This office action is responsive to communication filed on 10/09/2007.

Terminal Disclaimer

The terminal disclaimer filed on 10/09/2007 has been reviewed and is accepted. The terminal disclaimer has been recorded.

EXAMINER'S AMENDMENT

An examiner's amendment to the record appears below. Should the changes and/or additions be unacceptable to applicant, an amendment may be filed as provided by 37 CFR 1.312. To ensure consideration of such an amendment, it MUST be submitted no later than the payment of the issue fee.

Authorization for this examiner's amendment was given in a telephone interview with Robert C. Kain, Jr. on 11/12/2007.

The application has been amended as follows:

In claims 16 – 27, please insert before medium (line 1) --- storage ---.

Contact Information

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Yves Dalencourt whose telephone number is (571) 272-3998. The examiner can normally be reached on M-TH 7:30AM - 6:00PM.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Ario Etienne can be reached on (571) 272 4001. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

November 12, 2007

YVES DALENCOURT
PRIMARY EXAMINER
TECHNOLOGY CENTER 2100

Issue Classification

Application/Control No.	Applicant(s)/Patent Under Reexamination
10155192	REDLICH ET AL.
Examiner	**Art Unit**
Dalencourt, Yves	2157

ORIGINAL		INTERNATIONAL CLASSIFICATION	
CLASS	SUBCLASS	CLAIMED	NON-CLAIMED
709	244	G 0 6 F 15 / 173	

CROSS REFERENCE(S)

CLASS	SUBCLASS (ONE SUBCLASS PER BLOCK)				
709	206	218	219	246	
726	26	27			
713	166	189			

NONE (Assistant Examiner) (Date)	YVES DALENCOURT PRIMARY EXAMINER TECHNOLOGY CENTER 2100	**Total Claims Allowed:** 27
(Legal Instruments Examiner) (Date)	Yves Dalencourt 11/12/2007 (Primary Examiner) (Date)	O.G. Print Claim(s): 1 / O.G. Print Figure: 1A & 1B

Index of Claims	Application/Control No.	Applicant(s)/Patent Under Reexamination
	10155192	REDLICH ET AL.
	Examiner	Art Unit
	Dalencourt, Yves	2157

✓	Rejected	-	Cancelled	N	Non-Elected	A	Appeal
=	Allowed	÷	Restricted	I	Interference	O	Objected

☐ Claims renumbered in the same order as presented by applicant ☐ CPA ☐ T.D. ☐ R.1.47

CLAIM		DATE								
Final	Original									
1	1									
2	2									
3	3									
4	4									
5	5									
6	6									
10	7									
11	8									
12	9									
13	10									
14	11									
15	12									
7	13									
8	14									
9	15									
16	16									
17	17									
18	18									
19	19									
20	20									
21	21									
22	22									
23	23									
24	24									
25	25									
26	26									
27	27									

Search Notes	Application/Control No.	Applicant(s)/Patent Under Reexamination
	10155192	REDLICH ET AL.
	Examiner	Art Unit
	Dalencourt, Yves	2157

SEARCHED

Class	Subclass	Date	Examiner

SEARCH NOTES

Search Notes	Date	Examiner
WEST (updated text search)	11/10/2007	YD
WEST (updatedt text search)	11/12/2007	YD
IEEE Xplore	11/12/2007	YD

INTERFERENCE SEARCH

Class	Subclass	Date	Examiner
WEST (updated text search 709/26)		11/12/2007	YD

PART B - FEE(S) TRANSMITTAL

Complete and send this form, together with applicable fee(s), to: Mail **Mail Stop ISSUE FEE**
Commissioner for Patents
P.O. Box 1450
Alexandria, Virginia 22313-1450
or Fax (571)-273-2885

INSTRUCTIONS: This form should be used for transmitting the ISSUE FEE and PUBLICATION FEE (if required). Blocks 1 through 5 should be completed where appropriate. All further correspondence including the Patent, advance orders and notification of maintenance fees will be mailed to the current correspondence address as indicated unless corrected below or directed otherwise in Block 1, by (a) specifying a new correspondence address; and/or (b) indicating a separate "FEE ADDRESS" for maintenance fee notifications.

CURRENT CORRESPONDENCE ADDRESS (Note: Use Block 1 for any change of address)

Robert C. Kain, Jr.
750 Southeast Third Avenue
Suite 100
Ft. Lauderdale, Florida 333161153

Note: A certificate of mailing can only be used for domestic mailings of the Fee(s) Transmittal. This certificate cannot be used for any other accompanying papers. Each additional paper, such as an assignment or formal drawing, must have its own certificate of mailing or transmission.

Certificate of Mailing or Transmission

I hereby certify that this Fee(s) Transmittal is being deposited with the United States Postal Service with sufficient postage for first class mail in an envelope addressed to the Mail Stop ISSUE FEE address above, or being facsimile transmitted to the USPTO (571) 273-2885, on the date indicated below.

Robert C. Kain, Jr.	(Depositor's name)
/rck/	(Signature)
01/22/2008	(Date)

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/155,192	05/23/2002	Ron M. Redlich	6851-02-cip-3	9510

TITLE OF INVENTION:

APPLN. TYPE	SMALL ENTITY	ISSUE FEE DUE	PUBLICATION FEE DUE	PREV. PAID ISSUE FEE	TOTAL FEE(S) DUE	DATE DUE
nonprovisional	yes	$720	$300	$0	$1020	02/15/2008

EXAMINER	ART UNIT	CLASS-SUBCLASS

1. Change of correspondence address or indication of "Fee Address" (37 CFR 1.363).

☐ Change of correspondence address (or Change of Correspondence Address form PTO/SB/122) attached.

☐ "Fee Address" indication (or "Fee Address" Indication form PTO/SB/47; Rev 03-02 or more recent) attached. **Use of a Customer Number is required.**

2. For printing on the patent front page, list

(1) the names of up to 3 registered patent attorneys or agents OR, alternatively,

(2) the name of a single firm (having as a member a registered attorney or agent) and the names of up to 2 registered patent attorneys or agents. If no name is listed, no name will be printed.

1 Robert C. Kain, Jr.
2 Fleit Kain
3

3. ASSIGNEE NAME AND RESIDENCE DATA TO BE PRINTED ON THE PATENT (print or type)

PLEASE NOTE: Unless an assignee is identified below, no assignee data will appear on the patent. If an assignee is identified below, the document has been filed for recordation as set forth in 37 CFR 3.11. Completion of this form is NOT a substitute for filing an assignment.

(A) NAME OF ASSIGNEE: Digital Doors, Inc.

(B) RESIDENCE: (CITY and STATE OR COUNTRY): Miami Beach, Florida

Please check the appropriate assignee category or categories (will not be printed on the patent) : ☐ Individual ☑ Corporation or other private group entity ☐ Government

4a. The following fee(s) are submitted:

☑ Issue Fee
☑ Publication Fee (No small entity discount permitted)
☐ Advance Order - # of Copies _____

4b. Payment of Fee(s): **(Please first reapply any previously paid issue fee shown above)**

☐ A check is enclosed.
☐ Payment by credit card. Form PTO-2038 is attached.
☑ The Director is hereby authorized to charge the required fee(s), any deficiency, or credit any overpayment, to Deposit Account Number 031231 (enclose an extra copy of this form).

5. **Change in Entity Status** (from status indicated above)

☐ a. Applicant claims SMALL ENTITY status. See 37 CFR 1.27.

☐ b. Applicant is no longer claiming SMALL ENTITY status. See 37 CFR 1.27(g)(2).

NOTE: The Issue Fee and Publication Fee (if required) will not be accepted from anyone other than the applicant; a registered attorney or agent; or the assignee or other party in interest as shown by the records of the United States Patent and Trademark Office.

Authorized Signature /rck/ Date 01/22/2008

Typed or printed name Robert C. Kain, Jr. Registration No. 30648

This collection of information is required by 37 CFR 1.311. The information is required to obtain or retain a benefit by the public which is to file (and by the USPTO to process) an application. Confidentiality is governed by 35 U.S.C. 122 and 37 CFR 1.14. This collection is estimated to take 12 minutes to complete, including gathering, preparing, and submitting the completed application form to the USPTO. Time will vary depending upon the individual case. Any comments on the amount of time you require to complete this form and/or suggestions for reducing this burden, should be sent to the Chief Information Officer, U.S. Patent and Trademark Office, U.S. Department of Commerce, P.O. Box 1450, Alexandria, Virginia 22313-1450. DO NOT SEND FEES OR COMPLETED FORMS TO THIS ADDRESS. SEND TO: Commissioner for Patents, P.O. Box 1450, Alexandria, Virginia 22313-1450.

Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it displays a valid OMB control number.

Electronic Patent Application Fee Transmittal

Application Number:	10155192
Filing Date:	23-May-2002
Title of Invention:	DATA SECURITY SYSTEM AND METHOD WITH PARSING AND DISPERSION TECHNIQUES
First Named Inventor/Applicant Name:	Ron M. Redlich
Filer:	Robert C. Kain
Attorney Docket Number:	6851-2-CIP-3

Filed as Small Entity

Utility Filing Fees

Description	Fee Code	Quantity	Amount	Sub-Total in USD($)
Basic Filing:				
Pages:				
Claims:				
Miscellaneous-Filing:				
Petition:				
Patent-Appeals-and-Interference:				
Post-Allowance-and-Post-Issuance:				
Utility Appl issue fee	2501	1	720	720
Publ. Fee- early, voluntary, or normal	1504	1	300	300

Description	Fee Code	Quantity	Amount	Sub-Total in USD($)
Extension-of-Time:				
Miscellaneous:				
			Total in USD ($)	**1020**

Electronic Acknowledgement Receipt

EFS ID:	2745272
Application Number:	10155192
International Application Number:	
Confirmation Number:	9510
Title of Invention:	DATA SECURITY SYSTEM AND METHOD WITH PARSING AND DISPERSION TECHNIQUES
First Named Inventor/Applicant Name:	Ron M. Redlich
Customer Number:	4897
Filer:	Robert C. Kain
Filer Authorized By:	
Attorney Docket Number:	6851-2-CIP-3
Receipt Date:	22-JAN-2008
Filing Date:	23-MAY-2002
Time Stamp:	07:20:08
Application Type:	Utility under 35 USC 111(a)

Payment information:

Submitted with Payment	yes
Payment Type	Electronic Funds Transfer
Payment was successfully received in RAM	$1020
RAM confirmation Number	11360
Deposit Account	
Authorized User	

File Listing:

Document Number	Document Description	File Name	File Size(Bytes) /Message Digest	Multi Part /.zip	Pages (if appl.)

1	Issue Fee Payment (PTO-85B)	6851-02-cip3-issue-fee-transmittal-012208.pdf	77512 e4739289fad7642d0b6eae67d1127b5623c4abf5	no	1
Warnings:					
Information:					
2	Fee Worksheet (PTO-06)	fee-info.pdf	8305 548909da4b2f1cae19b16c12e94f4e58acc38a0f	no	2
Warnings:					
Information:					
		Total Files Size (in bytes):	85817		

This Acknowledgement Receipt evidences receipt on the noted date by the USPTO of the indicated documents, characterized by the applicant, and including page counts, where applicable. It serves as evidence of receipt similar to a Post Card, as described in MPEP 503.

New Applications Under 35 U.S.C. 111
If a new application is being filed and the application includes the necessary components for a filing date (see 37 CFR 1.53(b)-(d) and MPEP 506), a Filing Receipt (37 CFR 1.54) will be issued in due course and the date shown on this Acknowledgement Receipt will establish the filing date of the application.

National Stage of an International Application under 35 U.S.C. 371
If a timely submission to enter the national stage of an international application is compliant with the conditions of 35 U.S.C. 371 and other applicable requirements a Form PCT/DO/EO/903 indicating acceptance of the application as a national stage submission under 35 U.S.C. 371 will be issued in addition to the Filing Receipt, in due course.

New International Application Filed with the USPTO as a Receiving Office
If a new international application is being filed and the international application includes the necessary components for an international filing date (see PCT Article 11 and MPEP 1810), a Notification of the International Application Number and of the International Filing Date (Form PCT/RO/105) will be issued in due course, subject to prescriptions concerning national security, and the date shown on this Acknowledgement Receipt will establish the international filing date of the application.

PTO/SB/08A (07-05)
Approved for use through 07/31/2006. OMB 0651-0031
U.S. Patent and Trademark Office; U.S. DEPARTMENT OF COMMERCE
Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it contains a valid OMB control number.

Substitute for form 1449/PTO

INFORMATION DISCLOSURE STATEMENT BY APPLICANT

(Use as many sheets as necessary)

Sheet 1 of 1

Complete if Known	
Application Number	10/155,192
Filing Date	May 23, 2002
First Named Inventor	Ron M. Redlich, et al.
Art Unit	2143
Examiner Name	Chung
Attorney Docket Number	6851-02-cip3

U. S. PATENT DOCUMENTS

Examiner Initials*	Cite No.[1]	Document Number Number-Kind Code[2] (if known)	Publication Date MM-DD-YYYY	Name of Patentee or Applicant of Cited Document	Pages, Columns, Lines, Where Relevant Passages or Relevant Figures Appear
/YD/		US- 6,253,203	06-26-2001	O'Flaherty et al.	
		US-			
		US-			
		US-			
		US-			
		US-			
		US-			
		US-			
		US-			
		US-			
		US-			
		US-			
		US-			
		US-			
		US-			
		US-			
		US-			
		US-			
		US-			

FOREIGN PATENT DOCUMENTS

Examiner Initials*	Cite No.[1]	Foreign Patent Document Country Code[3]-Number[4]-Kind Code[5] (if known)	Publication Date MM-DD-YYYY	Name of Patentee or Applicant of Cited Document	Pages, Columns, Lines, Where Relevant Passages Or Relevant Figures Appear	T[6]

Examiner Signature	/Yves Dalencourt/	Date Considered	01/30/2008

*EXAMINER: Initial if reference considered, whether or not citation is in conformance with MPEP 609. Draw line through citation if not in conformance and not considered. Include copy of this form with next communication to applicant. [1] Applicant's unique citation designation number (optional). [2] See Kinds Codes of USPTO Patent Documents at www.uspto.gov or MPEP 901.04. [3] Enter Office that issued the document, by the two-letter code (WIPO Standard ST.3). [4] For Japanese patent documents, the indication of the year of the reign of the Emperor must precede the serial number of the patent document. [5] Kind of document by the appropriate symbols as indicated on the document under WIPO Standard ST.16 if possible. [6] Applicant is to place a check mark here if English language Translation is attached.

This collection of information is required by 37 CFR 1.97 and 1.98. The information is required to obtain or retain a benefit by the public which is to file (and by the USPTO to process) an application. Confidentiality is governed by 35 U.S.C. 122 and 37 CFR 1.14. This collection is estimated to take 2 hours to complete, including gathering, preparing, and submitting the completed application form to the USPTO. Time will vary depending upon the individual case. Any comments on the amount of time you require to complete this form and/or suggestions for reducing this burden, should be sent to the Chief Information Officer, U.S. Patent and Trademark Office, P.O. Box 1450, Alexandria, VA 22313-1450. DO NOT SEND FEES OR COMPLETED FORMS TO THIS ADDRESS. **SEND TO: Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450.**

If you need assistance in completing the form, call 1-800-PTO-9199 (1-800-786-9199) and select option 2.

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.	ISSUE DATE	PATENT NO.	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/155,192	03/25/2008	7349987	6851-2-CIP-3	9510

4897 7590 03/05/2008

ROBERT C. KAIN, JR.
750 SOUTHEAST THIRD AVENUE
SUITE 100
FT LAUDERDALE, FL 333161153

ISSUE NOTIFICATION

The projected patent number and issue date are specified above.

Determination of Patent Term Adjustment under 35 U.S.C. 154 (b)
(application filed on or after May 29, 2000)

The Patent Term Adjustment is 873 day(s). Any patent to issue from the above-identified application will include an indication of the adjustment on the front page.

If a Continued Prosecution Application (CPA) was filed in the above-identified application, the filing date that determines Patent Term Adjustment is the filing date of the most recent CPA.

Applicant will be able to obtain more detailed information by accessing the Patent Application Information Retrieval (PAIR) WEB site (http://pair.uspto.gov).

Any questions regarding the Patent Term Extension or Adjustment determination should be directed to the Office of Patent Legal Administration at (571)-272-7702. Questions relating to issue and publication fee payments should be directed to the Customer Service Center of the Office of Patent Publication at (571)-272-4200.

APPLICANT(s) (Please see PAIR WEB site http://pair.uspto.gov for additional applicants):

Ron M. Redlich, Miami Beach, FL;
Martin A. Nemzow, Miami Beach, FL;