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

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
24 March 2011 (24.03.2011)

PCT

(10) International Publication Number
WO 2011/034792 A1

(51) International Patent Classification:
G06F 21/00 (2006.01)

(21) International Application Number:
PCT/US2010/048475

(22) International Filing Date:
10 September 2010 (10.09.2010)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:
12/560,261 15 September 2009 (15.09.2009) US

(71) Applicant *(for all designated States except US)*: **SYMANTEC CORPORATION** [US/US]; 350 Ellis Street, Mountain View, CA 94043 (US).

(72) Inventors; and

(75) Inventors/Applicants *(for US only)*: **SESHADRI, Vijay** [IN/US]; C/o Symantec Corporation, 350 Ellis Street, Mountain View, CA 94043 (US). **RAMZAN, Zulfikar** [US/US]; C/o Symantec Corporation, 350 Ellis Street, Mountain View, CA 94043 (US). **HOAGLAND, James** [US/US]; C/o Symantec Corporation, 350 Ellis Street, Mountain View, CA 94043 (US). **GLICK, Adam, L.** [US/US]; C/o Symantec Corporation, 350 Ellis Street, Mountain View, CA 94043 (US). **WRIGHT, Adam** [US/US]; C/o Symantec Corporation, 350 Ellis Street, Mountain View, CA 94043 (US).

[Continued on next page]

(54) Title: INDIVIDUALIZED TIME-TO-LIVE FOR REPUTATION SCORES OF COMPUTER FILES

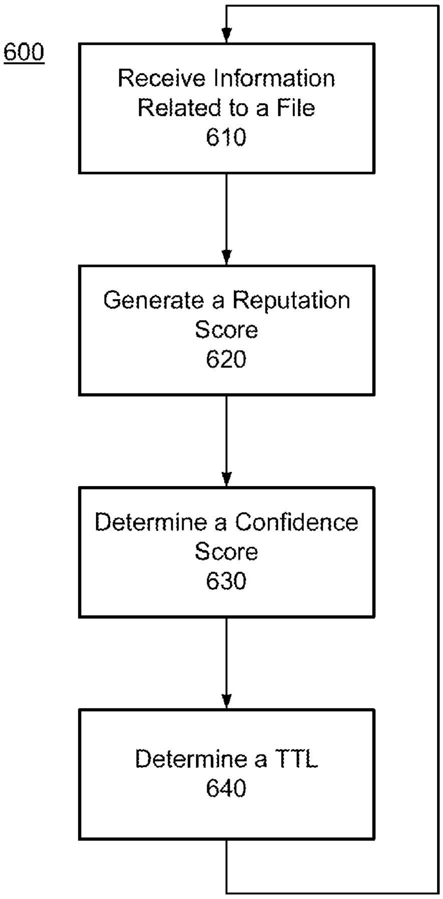
600
Receive Information Related to a File 610
Generate a Reputation Score 620
Determine a Confidence Score 630
Determine a TTL 640

FIG. 6

(57) Abstract: An individualized time -to-live (TTL) is determined for a reputation score of a computer file. The TTL is determined based on the reputation score and the confidence in the reputation score. The confidence can be determined based on attributes such as the reputation score, an age of the file, and a prevalence of the file. The reputation score is used to determine whether the file is malicious during a validity period defined by the TTL, and discarded thereafter.

(74) **Agents**: **ZHANG, Jie** et al.; Fenwick & West LLP, 801 California Street, Mountain View, CA 94041 (US).

(81) **Designated States** *(unless otherwise indicated, for every kind of national protection available)*: AE, AG, AL, AM, AO, AT, AU, AZ, BA, BB, BG, BH, BR, BW, BY, BZ, CA, CH, CL, CN, CO, CR, CU, CZ, DE, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IS, JP, KE, KG, KM, KN, KP, KR, KZ, LA, LC, LK, LR, LS, LT, LU, LY, MA, MD, ME, MG, MK, MN, MW, MX, MY, MZ, NA, NG, NI, NO, NZ, OM, PE, PG, PH, PL, PT, RO, RS, RU, SC, SD, SE, SG, SK, SL, SM, ST, SV, SY, TH, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, ZA, ZM, ZW.

(84) **Designated States** *(unless otherwise indicated, for every kind of regional protection available)*: ARIPO (BW, GH, GM, KE, LR, LS, MW, MZ, NA, SD, SL, SZ, TZ, UG, ZM, ZW), Eurasian (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European (AL, AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HR, HU, IE, IS, IT, LT, LU, LV, MC, MK, MT, NL, NO, PL, PT, RO, SE, SI, SK, SM, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, ML, MR, NE, SN, TD, TG).

Published:

— *with international search report (Art. 21(3))*

INDIVIDUALIZED TIME-TO-LIVE FOR REPUTATION SCORES OF COMPUTER FILES

BACKGROUND

FIELD OF DISCLOSURE

[0001] The disclosure generally relates to the field of computer security, in particular to determining whether a computer file is malicious.

DESCRIPTION OF THE RELATED ART

[0002] A wide variety of malicious software (malware) can attack modern computers. Malware threats include computer viruses, worms, Trojan horse programs, spyware, adware, crimeware, and phishing websites. Malicious entities sometimes attack servers that store sensitive or confidential data that can be used to the malicious entity's own advantage. Similarly, other computers, including home computers, must be constantly protected from malicious software that can be transmitted when a user communicates with others via electronic mail, when a user downloads new programs or program updates, and in many other situations. The different options and methods available to malicious entities for attack on a computer are numerous.

[0003] Conventional techniques for detecting malware, such as signature string scanning, are becoming less effective. Modern malware is often targeted and delivered to only a relative handful of computers. For example, a Trojan horse program can be designed to target computers in a particular department of a particular enterprise. Such malware might never be encountered by security analysts, and thus the security software might never be configured with signatures for detecting such malware. Mass-distributed malware, in turn, can contain polymorphisms that make every instance of the malware unique. As a result, it is difficult to develop signature strings that reliably detect all instances of the malware.

[0004] Newer techniques for detecting malware involve the use of reputation systems. A reputation system can determine the reputation of a file encountered on a computer in order to assess the likelihood that the file is malware. One way to develop the reputation for a file is to collect reports from networked computers on which the file is found and base the reputation on information within the reports. A file's reputation can change over time as more reports are collected. There is a need for a way to efficiently provide the changing reputation scores to the networked computers and other entities that use the reputation scores to detect malware.

SUMMARY

[0005] Embodiments of the present disclosure include methods (and corresponding systems and computer program products) for generating and utilizing individualized time-to-lives (TTLs) for reputation scores of computer files.

[0006] One aspect of the present disclosure is a computer-implemented method for generating an individualized time-to-live (TTL) for a reputation score of a file, comprising: receiving a request from a client system, the request comprising an identifier of the file; generating a reputation score of the file, the reputation score representing an assessment of a trustworthiness of the file; determining a TTL for the reputation score based on the reputation score, the TTL describing a validity period for the reputation score; and transmitting the reputation score and the TTL to the client system in response to the request.

[0007] Another aspect of the present disclosure is a computer system for generating an individualized time-to-live (TTL) for a reputation score of a file, comprising: a computer-readable storage medium comprising executable computer program code for: a communication module for receiving a request from a client system, the request comprising an identifier of the file; a reputation score generation module for generating a reputation score of the file, the reputation score representing an assessment of a trustworthiness of the file; and a TTL determination module for determining a TTL for the reputation score based on the reputation score, the TTL describing a validity period for the reputation score; wherein the communication module is further configured for transmitting the reputation score and the TTL to the client system in response to the request.

[0008] Still another aspect of the present disclosure is a computer-readable storage medium encoded with executable computer program code for generating an individualized time-to-live (TTL) for a reputation score of a file, the computer program code comprising program code for: receiving a request from a client system, the request comprising an identifier of the file; generating a reputation score of the file, the reputation score representing an assessment of a trustworthiness of the file; determining a TTL for the reputation score based on the reputation score, the TTL describing a validity period for the reputation score; and transmitting the reputation score and the TTL to the client system in response to the request.

[0009] The features and advantages described in the specification are not all inclusive and, in particular, many additional features and advantages will be apparent to one of ordinary skill in the art in view of the drawings, specification, and claims. Moreover, it should be noted that the language used in the specification has been principally selected for readability and instructional purposes, and may not have been selected to delineate or circumscribe the disclosed subject matter.

BRIEF DESCRIPTION OF DRAWINGS

[0010] Figure (FIG.) 1 is a high-level block diagram of a computing environment according to one embodiment of the present disclosure.

[0011] FIG. 2 is a high-level block diagram illustrating an example of a computer for use in the computing environment shown in FIG. 1 according to one embodiment of the present disclosure.

[0012] FIG. 3 is a high-level block diagram illustrating modules within a security module according to one embodiment of the present disclosure.

[0013] FIG. 4 is a high-level block diagram illustrating modules within a security system according to one embodiment of the present disclosure.

[0014] FIG. 5 is a flow diagram illustrating a process for determining whether a computer file is malicious based on a reputation score and an individualized time-to-live (TTL) for the reputation score according to one embodiment of the present disclosure.

[0015] FIG. 6 is a flow diagram illustrating a process for continuously generating a current reputation score and an accompanying individualized TTL for the reputation score for a computer file according to one embodiment of the present disclosure.

DETAILED DESCRIPTION

[0016] The Figures (FIGS.) and the following description describe certain embodiments by way of illustration only. One skilled in the art will readily recognize from the following description that alternative embodiments of the structures and methods illustrated herein may be employed without departing from the principles described herein. Reference will now be made in detail to several embodiments, examples of which are illustrated in the accompanying figures. It is noted that wherever practicable similar or like reference numbers may be used in the figures and may indicate similar or like functionality.

SYSTEM ENVIRONMENT

[0017] FIG. 1 is a high-level block diagram that illustrates a computing environment 100 for generating and utilizing an individualized time-to-live (TTL) for a reputation score of a computer file, according to one embodiment of the present disclosure. As shown, the computing environment 100 includes a client system 110 and a security system 120 connected through a network 130. Only one of each entity is illustrated in order to simplify and clarify the present description. There can be other entities in the computing environment 100 as well.

[0018] The client system 110 is an electronic device that can host malicious software. In one embodiment, the client system 110 is a conventional computer system executing, for example, a Microsoft Windows-compatible operating system (OS), Apple OS X, and/or a Linux distribution. In another embodiment, the client system 110 is another device having computer functionality, such as a personal digital assistant (PDA), mobile telephone, video game system, etc. The client system 110 typically stores numerous computer files that can host malicious software.

[0019] Malicious software, sometimes called "malware," is generally defined as software that executes on the client system 110 surreptitiously or that has some surreptitious functionality. Malware can take many forms, such as parasitic viruses that attach to legitimate files, worms that exploit weaknesses in the computer's security in order to infect the computer and spread to other computers, Trojan horse programs that appear legitimate but actually contain hidden malicious code, and spyware that monitors keystrokes and/or other actions on the computer in order to capture sensitive information or display advertisements.

[0020] The client system 110 executes a security module 115 for detecting the presence of malware. The security module 115 can be, for example, incorporated into the OS of the computer or part of a separate comprehensive security package. In one embodiment, the security module 115 is provided by the entity that operates the security system 120. The security module 115 can communicate with the security system 120 via the network 130 .

[0021] In one embodiment, the security module 115 transmits a request for a reputation score of a file to the security system 120, and receives the reputation score and an associated TTL in return. The security module 115 determines whether the file is malicious based on the received reputation score during the period of time defined by the TTL (called the validity period) and other factors such as observed behaviors of the file, and discards the reputation score thereafter.

[0022] The reputation score represents an assessment of the trustworthiness of the associated file. For example, the reputation score can be a continuous value ranging from 0 to 1, which a score of 0 indicating a very low trustworthiness (e.g., the file is malicious) and a score of 1 indicating a very high trustworthiness (e.g., the file is legitimate).

[0023] The TTL specifies the period of time that the associated reputation score is valid. Depending upon the embodiment, the TTL can be specified as a length of time after a specified event (e.g., the reputation score is valid for one week after receipt of the score by the client system 110), as an explicit date (e.g., the reputation score is valid until July 1, 2012), or in another manner.

[0024] The security system 120 is a hardware device and/or software program configured to receive information about computer files (e.g., their behaviors) from the client systems 110 and other sources, generate reputation scores for these files, and determine associated TTLs based on the reputation scores and confidence in the reputation scores. The security system 120 also receives requests from the client systems 110 for reputation scores of particular files, and provides their current reputation scores and TTLs in response. An example of the security system 120 is a web-based system providing security software and services allowing the security module 115 to detect and remediate malware on the client systems 110.

[0025] The security system 120 generates the reputation score for a file based on attributes that correlate to the trustworthiness of the file. For example, the security system 120 can analyze a reputation of the source(s) of the file (e.g., whether the file was downloaded from a reputable website or from a less reputable website), can analyze whether the file is signed with a digital signature (e.g., a file digitally signed by a reputable entity is more trustworthy than a file signed by a less reputable entity or a file without a digital signature) and whether the file is prevalent among the client systems 110, to name a few.

[0026] The security system 120 determines a confidence score for a generated reputation score. The confidence score measures the confidence the security system 120 has in the reputation score (i.e., the likelihood of the reputation score reflecting the true trustworthiness of the underlying file). For example, the confidence score can be a continuous value ranging from 0 to 1, which a score of 0 indicating a very low confidence (e.g., the indication value of the reputation score is very low) and a score of 1 indicating a very high confidence (e.g., the reputation score almost certainly reflects the true trustworthiness of the underlying file).

[0027] The security system 120 determines the confidence score based on attributes that correlate to the confidence in the generated reputation score, such as the value of the reputation score, the underlying file's age (i.e., how long the security system 120 has known about the file), and the file's prevalence across the user base of the security system 120.

[0028] The security system 120 computes the TTL based on attributes that correlate to the trustworthiness of the file (i.e., the reputation score) and the confidence score for the reputation score. Generally, the security system 120 sets long TTLs for reputation scores with high confidence scores, and sets short TTLs for reputation scores with low confidence scores. In other words, the TTL represents an assessment of the length of time it takes for the security system 120 to collect enough additional related information that may cause the security system 120 to generate a materially different reputation score for the file. If the assessment of trustworthiness of a file is unlikely to change for a long time, the TTL is long, and otherwise short. By setting long TTLs for reputation scores with high confidence scores, the client system 110 relies on such reputation scores for a long period of time before requesting an update from the security system 120.

[0029] One or more of the functions of the security system 120 can also be executed on the client system 110 or in a cloud computing environment. As used herein, cloud computing refers to a style of computing in which dynamically scalable and often virtualized resources are provided as a service over the Internet. Thus, cloud computing customers generally do not own the physical infrastructure serving as host to the software platform in question, but instead rent usage of resources from a third-party provider, where consume these resources as a service and pay only for resources used.

[0030] The network 130 enables communications between the client system 110 and the security system 120. In one embodiment, the network 130 uses standard communications technologies and/or protocols. Thus, the network 130 can include links using technologies such as Ethernet, 802.11, worldwide interoperability for microwave access (WiMAX), 3G, digital subscriber line (DSL), asynchronous transfer mode (ATM), InfiniBand, PCI Express Advanced Switching, etc. Similarly, the networking protocols used on the network 130 can include multiprotocol label switching (MPLS), the transmission control protocol/Internet protocol (TCP/IP), the User Datagram Protocol (UDP), the hypertext transport protocol (HTTP), the simple mail transfer protocol (SMTP), the file transfer protocol (FTP), etc. The data exchanged over the network 130 can be represented using technologies and/or formats including the hypertext markup language (HTML), the extensible markup language (XML), etc. In addition, all or some of links can be encrypted using conventional encryption technologies such as secure sockets layer (SSL), transport layer security (TLS), virtual private networks (VPNs), Internet Protocol security (IPsec), etc. In another embodiment, the entities can use custom and/or dedicated data communications technologies instead of, or in addition to, the ones described above. Depending upon the embodiment, the network 130 can also include links to other networks such as the Internet.

COMPUTER ARCHITECTURE

[0031] The entities shown in FIG. 1 are implemented using one or more computers. FIG. 2 is a high-level block diagram illustrating an example computer 200. The computer 200 includes at least one processor 202 coupled to a chipset 204. The chipset 204 includes a memory controller hub 220 and an input/output (I/O) controller hub 222. A memory 206 and a graphics adapter 212 are coupled to the memory controller hub 220, and a display 218 is coupled to the graphics adapter 212. A storage device 208, keyboard 210, pointing device 214, and network adapter 216 are coupled to the I/O controller hub 222. Other embodiments of the computer 200 have different architectures.

[0032] The storage device 208 is a computer-readable storage medium such as a hard drive, compact disk read-only memory (CD-ROM), DVD, or a solid-state memory device. The memory 206 holds instructions and data used by the processor 202. The pointing device 214 is a mouse, track ball, or other type of pointing device, and is used in combination with the keyboard 210 to input data into the computer system 200. The graphics adapter 212 displays images and other information on the display 218. The network adapter 216 couples the computer system 200 to one or more computer networks.

[0033] The computer 200 is adapted to execute computer program modules for providing functionality described herein. As used herein, the term “module” refers to computer program logic used to provide the specified functionality. Thus, a module can be implemented in hardware, firmware, and/or software. In one embodiment, program modules are stored on the storage device 208, loaded into the memory 206, and executed by the processor 202.

[0034] The types of computers 200 used by the entities of FIG. 1 can vary depending upon the embodiment and the processing power required by the entity. For example, the security system 120 might comprise multiple blade servers working together to provide the functionality described herein. As another example, the client system 110 might comprise a mobile telephone with limited processing power. The computers 200 can lack some of the components described above, such as keyboards 210, graphics adapters 212, and displays 218.

Example Architectural Overview of the Security Module

[0035] FIG. 3 is a high-level block diagram illustrating a detailed view of modules within the security module 115 according to one embodiment. Some embodiments of the security module 115 have different and/or other modules than the ones described herein. Similarly, the functions can be distributed among the modules in accordance with other embodiments in a different manner than is described here. As illustrated, the security module 115 includes a file monitor module 310, a security analysis engine 320, a communication module 330, and a data store 340.

[0036] The file monitor module 310 continuously monitors computer files within the client system 110 and collects related information such as their sources (e.g., the websites from which the files were downloaded), digital signatures, behaviors, and system properties such as the dates of creation and the dates of last modification. Such related information is collectively called the "metadata" of the associated files.

[0037] The security analysis engine 320 determines whether the monitored files are malicious based on factors such as their metadata and reputation scores. In one embodiment, the security analysis engine 320 makes the determination for a file when a suspicious behavior (e.g., an attempt to write to the system registry) is observed for that file. For example, the security analysis engine 320 can determines that a first file with a good reputation (i.e., high reputation score) that attempts to write to the registry is probably not malware, and that a second file with a bad reputation (i.e., low reputation score) that attempts to write to the registry is probably malware.

[0038] If the file does not have a reputation score or if the reputation score for the file is expired (i.e., as indicated by its TTL), the security analysis engine 320 coordinates with the communication module 330 to request a current reputation score and an associated TTL from the security system 120. Otherwise, if the file is associated with a valid reputation score (e.g., the validity period has not ended), the security analysis engine 320 determines whether the file is malicious based on the reputation score and other factors such as the metadata.

[0039] In one embodiment, the security analysis engine 320 creates a black list of files determined malicious and a white list of files determined legitimate. If the verdict for a file (i.e., whether the file is malicious) changes, for example, because of a subsequent reputation score change, the security analysis engine 320 updates the black list and/or white list accordingly.

[0040] The communication module 330 communicates with the security system 120 to determine the reputation scores and corresponding TTLs for the monitored files. The communication module 330 transmits an identifier of the file (e.g., a digital fingerprint) and related information (e.g., the metadata) to the security system 120, and receives a reputation score and an accompanying TTL in response. In one embodiment, instead of or in addition to the identifier, the file itself can be transmitted to the security system 120. The communication module 330 stores the reputation score and the TTL in the data store 340 together with other related information for the file.

[0041] The data store 340 stores data used by the client system 110. Examples of such data include identifiers of the computer files resided on the client system 110 and their metadata, reputation scores, and associated TTLs, and information about the security system 120 (e.g., IP address). The data store 340 may be a relational database or any other type of database.

Example Architectural Overview of the Security System

[0042] FIG. 4 is a high-level block diagram illustrating a detailed view of modules within the security system 120 according to one embodiment. Some embodiments of the security system 120 have different and/or other modules than the ones described herein. Similarly, the functions can be distributed among the modules in accordance with other embodiments in a different manner than is described here. As illustrated, the security system 120 includes a communication module 410, a reputation score generation module 420, a confidence determination module 430, a TTL determination module 440, and a data store 450.

[0043] The communication module 410 receives inquiries from the security modules 115 running on the client systems 110 and provides responses to the security modules 115. The inquiries include identifiers of computer files and may also include metadata for the files. The responses include reputation scores for the identified files and TTLs of the reputation scores. The communication module 410 contacts the reputation score generation module 420 and the TTL determination module 440 for the current reputation scores and associated TTLs, accordingly. The communication module 410 also receives information related to computer files from client systems 110 and other resources and stores the received information in the data store 450.

[0044] The reputation score generation module 420 generates reputation scores for computer files. As described above, a reputation score is an assessment of the trustworthiness of the associated file. The reputation score generation module 420 generates the reputation score based on attributes that correlate to the trustworthiness of the file. Examples of such attributes include the prevalence (or popularity) of the file among the user base of the security system 120, the reputation of sources of the file, the reputation scores of other files resided on the same client systems 110 with the file, and the observed behaviors of the file, to name a few. In one embodiment, the reputation score generation module 420 continuously generates reputation scores as requests and information about computer files are received from the client systems 110 and other sources.

[0045] The confidence determination module 430 determines the confidence scores for the reputation scores generated by the reputation score generation module 420. As described above, a confidence score measures the confidence the security system 120 has in the generated reputation score. The confidence determination module 430 determines the confidence score based on attributes that correlate to the confidence in the generated reputation score. Examples of such attributes include the value of the reputation score, the file's age (i.e., how long the security system 120 has known about the file), and the file's prevalence across the user base of the security system 120.

[0046] A reputation score that is on or near the extreme ends of the spectrum serves as an indication of high confidence in the reputation score, since such a reputation score is unlikely to change materially over time. For example, if a reputation score is close to 0, the low end of the spectrum, indicating that the associated file is almost certainly malicious, the chance that the reputation score will change substantially over time is close to zero. In one embodiment, reputation scores are organized into the following reputation bands: extreme (higher than 0.95 or less than 0.05), decisive (between 0.95 and 0.8 or between 0.05 and 0.2), marginally decisive (between 0.7 and 0.8 or between 0.2 and 0.3), and indecisive (between 0.3 and 0.7).

[0047] The length of time the security system 120 has known about a file is another indicator of confidence in the reputation score of the file. If a file is relatively new to the security system 120, the confidence of the reputation score is low since the security system 120 probably has not collected enough information to make a good assessment of its trustworthiness. As the file becomes more familiar to the security system 120, the confidence in the reputation score increases, since the security system 120 is likely to have collected more information and can make a better assessment. In addition, if the security system 120 has known about the file for a long time, the chance of new information that will materially change the reputation score surfacing in the near future is also low. In one embodiment, file ages are organized into the following age bands: old (greater than 6 months), middle-aged (equal or less than 6 months, older than 3 months), young (equal or less than 3 months, older than 1.5 months); very young (equal or less than 1.5 months).

[0048] The prevalence of a file across the user base of the security system 120 is also an indicator of confidence in the reputation score of the file. Files that are more prevalent tend to be better known to the security system 120, and as a result the security system 120 tends to have more information about such prevalent files, boosting confidence of their reputation scores. In one embodiment, file prevalence is categorized into the following prevalence bands: very high (more than 50,000 machines), high (equal or less than 50,000, more than 25,000), medium (equal or less than 25,000, more than 10,000), low (equal or less than 10,000, more than 1,000), very low (equal or less than 1,000).

[0049] The confidence determination module 430 determines the confidence scores by collectively considering multiple confidence indicators such as the ones described above. For example, the confidence determination module 430 can determine the confidence scores by walking through the following table:

Prevalence	Age	Reputation	Confidence Score
Medium	Old	Indecisive	Very High
Very High	Old	Extreme	Very High
High	Old	Decisive	Very High
Medium	Middle-Aged	Marginally Decisive	High
High	Young	Marginally Decisive	Low
Low	Young	Extreme	Low
Very Low	Very Young	Marginally Decisive	Low
Very Low	Very Young	Extreme	Low
Very Low	Very Young	Marginally Decisive	Very Low
Very Low	Very Young	Indecisive	Very Low

The confidence determination module 430 can take the confidence score for the reputation score of a file from the first row (from top down) in the table for which the attributes of the file meet or exceed what is specified in each category. To simplify the issue, the confidence scores determined using the above table has one of the following four values: very high, high, low, and very low.

[0050] The TTL determination module 440 determines TTLs for reputation scores generated by the reputation score generation module 420. As described above, a TTL describes the validity period for its associated reputation score. The TTL determination module 440 determines a TTL based on the associated reputation score and the confidence score determined for the reputation score. Continuing with the above example, the TTL determination module 440 can determine the TTLs by walking through the following table in a similar manner:

Reputation	Confidence Score	TTL
Decisive	Very High	One Year
Marginally Decisive	High	One Month
Marginally Decisive	Low	One Week
Indecisive	Very Low	One Day

To simplify the issue, the TTLs determined using the above table has one of the following four values: one year, one month, one week, and one day.

[0051] The data store 450 stores data used by the security system 120. Examples of such data include information related to computer files (e.g., identifiers such as digital fingerprints, metadata, past/current reputation scores and TTLs), websites, digital signatures, and client systems 110, to name a few. Information related to files can be grouped together and indexed by file identifiers to facilitate rapid retrieval. Similar to the data store 340, the data store 450 may be a relational database or any other type of database.

Overview of Methodology for the Security Module

[0052] FIG. 5 is a flow diagram illustrating a process 500 for the security module 115 to determine whether a computer file is malicious based on a reputation score and an individualized TTL for the reputation score, according to one embodiment. Other embodiments can perform the steps of the process 500 in different orders. Moreover, other embodiments can include different and/or additional steps than the ones described herein.

[0053] Initially, the security module 115 running on the client system 110 identifies 510 a computer file for security examination. For example, the security module 115 monitors files residing on the client system 110, observes an attempt to perform a suspicious activity by one of the monitored files, and identifies 510 that file for security examination.

[0054] The security module 115 determines 520 whether the identified file has a valid reputation score available. For example, the security module 115 searches in the data store 340 for the presence of a reputation score associated with the identified file and the accompanying TTL. If no such reputation score is present or the available reputation score is already expired (i.e., as indicated by the TTL), the security module 115 determines 520 that no valid reputation score is available. Otherwise, if a reputation score for the identified file is present in the data store 340 and the validity period defined by the accompanying TTL has not ended, the security module 115 assumes the reputation score to be a valid assessment of the trustworthiness of the identified file, and determines 520 that the identified file has a valid reputation score available.

[0055] If the security module 115 has a valid reputation score available for the identified file, the security module 115 determines 550 whether the file is malicious based on the reputation score and other factors such as the observed behaviors of the identified file. The security module 115 can further determine whether to allow any attempt to perform suspicious activities by the identified file based on whether the file is determined malicious.

[0056] If no valid reputation score is available, the security module 115 requests 530 a reputation score for the identified file from the security system 120. For example, the security module 115 transmits to the security system 120 a request including an identifier of the file (e.g., the digital fingerprint). The security module 115 may also transmit metadata (e.g., observed behavior) for the identified file to the security system 120 along with the request. The security module 115 then receives 540 a reputation score and an accompanying TTL from the security system 120 in response to the request 530. Once the security module 115 receives the reputation score from the security system 120, the security module 115 determines 550 whether the file is malicious as described above.

OVERVIEW OF METHODOLOGY FOR THE SECURITY SYSTEM

[0057] FIG. 6 is a flow diagram illustrating a process 600 for the security system 120 to continuously generate a current reputation score and an accompanying individualized TTL for the reputation score for a computer file, according to one embodiment. Other embodiments can include different and/or additional steps than the ones described herein.

[0058] The security system 120 receives 610 information related to a computer file. Examples of the related information include an identifier of the computer file (e.g., digital signature), observed behaviors, source(s), prevalence (e.g., the client systems 110 hosting the file), and age (e.g., how long the file exists on a particular the client system 110), to name a few. The security system 120 receives 610 the information from sources such as the security module 115 running on the client system 110, the data store 450, and other sources. The security system 120 stores the related information received from the security module 115 in the data store 450.

[0059] The security system 120 generates 620 a current reputation score for the file based on the received information, determines 630 a confidence score for the generated reputation score, and determines 640 a TTL for the generated reputation score based on factors including the reputation score and the confidence score. The reputation score along with the TTL and the confidence score can be stored in the data store 450 as the current reputation score for the computer file.

[0060] The process 600 is repeated continuously to generate updated reputation scores and TTLs that account for newly-received related information for the file. The security system 120 may repeat the process 600 when new related information is received, when receiving a request for the current reputation score and TTL of the file, or at a predetermined time interval. For example, the security system 120 receives 610 a request for a reputation score for a file along with metadata of that file from the security module 115, generates 620 a reputation score taking into account the received metadata, and determines 630, 640 a confidence score and a TTL, and returns the reputation score and the TTL to the security module 115.

[0061] This approach intelligently determines an individualized TTL for a reputation score of a particular file based on attributes of the file that reflect the confidence in the reputation score. As a result, the number of requests the security modules 115 transmit to the security system 120 regarding files that are highly likely to be good or bad (i.e., files with high-confidence reputation scores) are reduced, and therefore the performances of the client systems 110 and the security system 120 are improved. Thus, by utilizing individualized TTLs, this approach advantageously achieves a tradeoff between providing satisfying real-time security to the client systems 110 and maintaining backend performance and scalability.

ADDITIONAL EMBODIMENTS

[0062] In one embodiment, at least a part of the security system 120 can be implemented in the security module 115 running on the client system 110 such that the client system 110 locally generates reputation scores and TTLs for computer files and does not re-generate the reputation scores until their TTLs expire.

[0063] In another embodiment, the TTL provided by the security system 120 can be further customized by the client system 110 based on its local security policy. For example, if the client system 110 has low security policies in place (e.g., computers in an Internet café) then the client system 110 can extend the TTLs of the reputation scores by a pre-determined and customizable factor. On the other hand, if the client system 110 enforces stringent local security policies, then the client system 110 may reduce the TTLs provided by the security system 120.

[0064] Some portions of above description describe the embodiments in terms of algorithmic processes or operations. These algorithmic descriptions and representations are commonly used by those skilled in the data processing arts to convey the substance of their work effectively to others skilled in the art. These operations, while described functionally, computationally, or logically, are understood to be implemented by computer programs comprising instructions for execution by a processor or equivalent electrical circuits, microcode, or the like. Furthermore, it has also proven convenient at times, to refer to these arrangements of functional operations as modules, without loss of generality. The described operations and their associated modules may be embodied in software, firmware, hardware, or any combinations thereof.

[0065] As used herein any reference to "one embodiment" or "an embodiment" means that a particular element, feature, structure, or characteristic described in connection with the embodiment is included in at least one embodiment. The appearances of the phrase "in one embodiment" in various places in the specification are not necessarily all referring to the same embodiment.

[0066] Some embodiments may be described using the expression "coupled" and "connected" along with their derivatives. It should be understood that these terms are not intended as synonyms for each other. For example, some embodiments may be described using the term "connected" to indicate that two or more elements are in direct physical or electrical contact with each other. In another example, some embodiments may be described using the term "coupled" to indicate that two or more elements are in direct physical or electrical contact. The term "coupled," however, may also mean that two or more elements are not in direct contact with each other, but yet still co-operate or interact with each other. The embodiments are not limited in this context.

[0067] As used herein, the terms "comprises," "comprising," "includes," "including," "has," "having" or any other variation thereof, are intended to cover a non-exclusive inclusion. For example, a process, method, article, or apparatus that comprises a list of elements is not necessarily limited to only those elements but may include other elements not expressly listed or inherent to such process, method, article, or apparatus. Further, unless expressly stated to the contrary, "or" refers to an inclusive or and not to an exclusive or. For example, a condition A or B is satisfied by any one of the following: A is true (or present) and B is false (or not present), A is false (or not present) and B is true (or present), and both A and B are true (or present).

[0068] In addition, use of the “a” or “an” are employed to describe elements and components of the embodiments herein. This is done merely for convenience and to give a general sense of the disclosure. This description should be read to include one or at least one and the singular also includes the plural unless it is obvious that it is meant otherwise.

[0069] Upon reading this disclosure, those of skill in the art will appreciate still additional alternative structural and functional designs for a system and a process for generating and utilizing individualized TTLs for reputation scores of computer files. Thus, while particular embodiments and applications have been illustrated and described, it is to be understood that the present invention is not limited to the precise construction and components disclosed herein and that various modifications, changes and variations which will be apparent to those skilled in the art may be made in the arrangement, operation and details of the method and apparatus disclosed herein without departing from the spirit and scope as defined in the appended claims.

WHAT IS CLAIMED IS:

1. A computer-implemented method for generating an individualized time-to-live (TTL) for a reputation score of a file, comprising:

receiving a request from a client system, the request comprising an identifier of the file;

generating a reputation score of the file, the reputation score representing an assessment of a trustworthiness of the file;

determining a TTL for the reputation score based on the reputation score, the TTL describing a validity period for the reputation score; and

transmitting the reputation score and the TTL to the client system in response to the request.

2. The computer-implemented method of claim 1, further comprising:

determining a confidence score for the reputation score, the confidence score measuring a likelihood of the reputation score reflecting actual trustworthiness of the file, wherein determining the TTL for the reputation score further comprises determining the TTL based on the reputation score and the confidence score.

3. The computer-implemented method of claim 2, wherein a validity period described by a TTL for a reputation score associated with a high confidence score is longer than a validity period described by a TTL for a reputation score associated with a low confidence score.

4. The computer-implemented method of claim 2, wherein the confidence score for a reputation score on or near extreme ends of its spectrum is higher than the confidence score for a reputation score not at the extreme ends.

5. The computer-implemented method of claim 2, wherein determining the confidence score for the reputation score comprises:

determining an age of the file, wherein the confidence score for a reputation score associated with an old file is higher than the confidence score for a reputation score associated with a young file.

6. The computer-implemented method of claim 2, wherein determining the confidence score for the reputation score comprises:

determining a prevalence of the file among client systems, wherein the confidence score for a reputation score associated with a prevalent file is higher than the confidence score for a reputation score associated with a less prevalent file.

7. The computer-implemented method of claim 1, wherein the reputation score is utilized by the client system to determine whether the file is malicious during the validity period described by the TTL.

8. A computer system for generating an individualized time-to-live (TTL) for a reputation score of a file, comprising:

a computer-readable storage medium comprising executable computer program code for:

a communication module for receiving a request from a client system, the request comprising an identifier of the file;

a reputation score generation module for generating a reputation score of the file, the reputation score representing an assessment of a trustworthiness of the file; and

a TTL determination module for determining a TTL for the reputation score based on the reputation score, the TTL describing a validity period for the reputation score;

wherein the communication module is further configured for transmitting the reputation score and the TTL to the client system in response to the request.

9. The computer system of claim 8, further comprising:

a confidence determination module for determining a confidence score for the reputation score, the confidence score measuring a likelihood of the reputation score reflecting actual trustworthiness of the file, wherein determining the TTL for the reputation score further comprises determining the TTL based on the reputation score and the confidence score.

10. The computer system of claim 9, wherein a validity period described by a TTL for a reputation score associated with a high confidence score is longer than a validity period described by a TTL for a reputation score associated with a low confidence score.

11. The computer system of claim 9, wherein the confidence score for a reputation score on or near extreme ends of its spectrum is higher than the confidence score for a reputation score not at the extreme ends.

12. The computer system of claim 9, wherein determining the confidence score for the reputation score comprises:

determining an age of the file, wherein the confidence score for a reputation score associated with an old file is higher than the confidence score for a reputation score associated with a young file.

13. The computer system of claim 9, wherein determining the confidence score for the reputation score comprises:

determining a prevalence of the file among client systems, wherein the confidence score for a reputation score associated with a prevalent file is higher than the confidence score for a reputation score associated with a less prevalent file.

14. The computer system of claim 8, wherein the reputation score is utilized by the client system to determine whether the file is malicious during the validity period described by the TTL.

15. A computer-readable storage medium encoded with executable computer program code for generating an individualized time-to-live (TTL) for a reputation score of a file, the computer program code comprising program code for:

receiving a request from a client system, the request comprising an identifier of the file;

generating a reputation score of the file, the reputation score representing an assessment of a trustworthiness of the file;

determining a TTL for the reputation score based on the reputation score, the TTL describing a validity period for the reputation score; and

transmitting the reputation score and the TTL to the client system in response to the request.

16. The computer-readable storage medium of claim 15, wherein the computer program code further comprises program code for:

determining a confidence score for the reputation score, the confidence score measuring a likelihood of the reputation score reflecting actual trustworthiness of the file, wherein determining the TTL for the reputation score further comprises determining the TTL based on the reputation score and the confidence score.

17. The computer-readable storage medium of claim 16, wherein a validity period described by a TTL for a reputation score associated with a high confidence score is longer than a validity period described by a TTL for a reputation score associated with a low confidence score.

18. The computer-readable storage medium of claim 16, wherein the confidence score for a reputation score on or near extreme ends of its spectrum is higher than the confidence score for a reputation score not at the extreme ends.

19. The computer-readable storage medium of claim 16, wherein determining the confidence score for the reputation score comprises:

determining an age of the file, wherein the confidence score for a reputation score associated with an old file is higher than the confidence score for a reputation score associated with a young file.

20. The computer-readable storage medium of claim 16, wherein determining the confidence score for the reputation score comprises:

determining a prevalence of the file among client systems, wherein the confidence score for a reputation score associated with a prevalent file is higher than the confidence score for a reputation score associated with a less prevalent file.

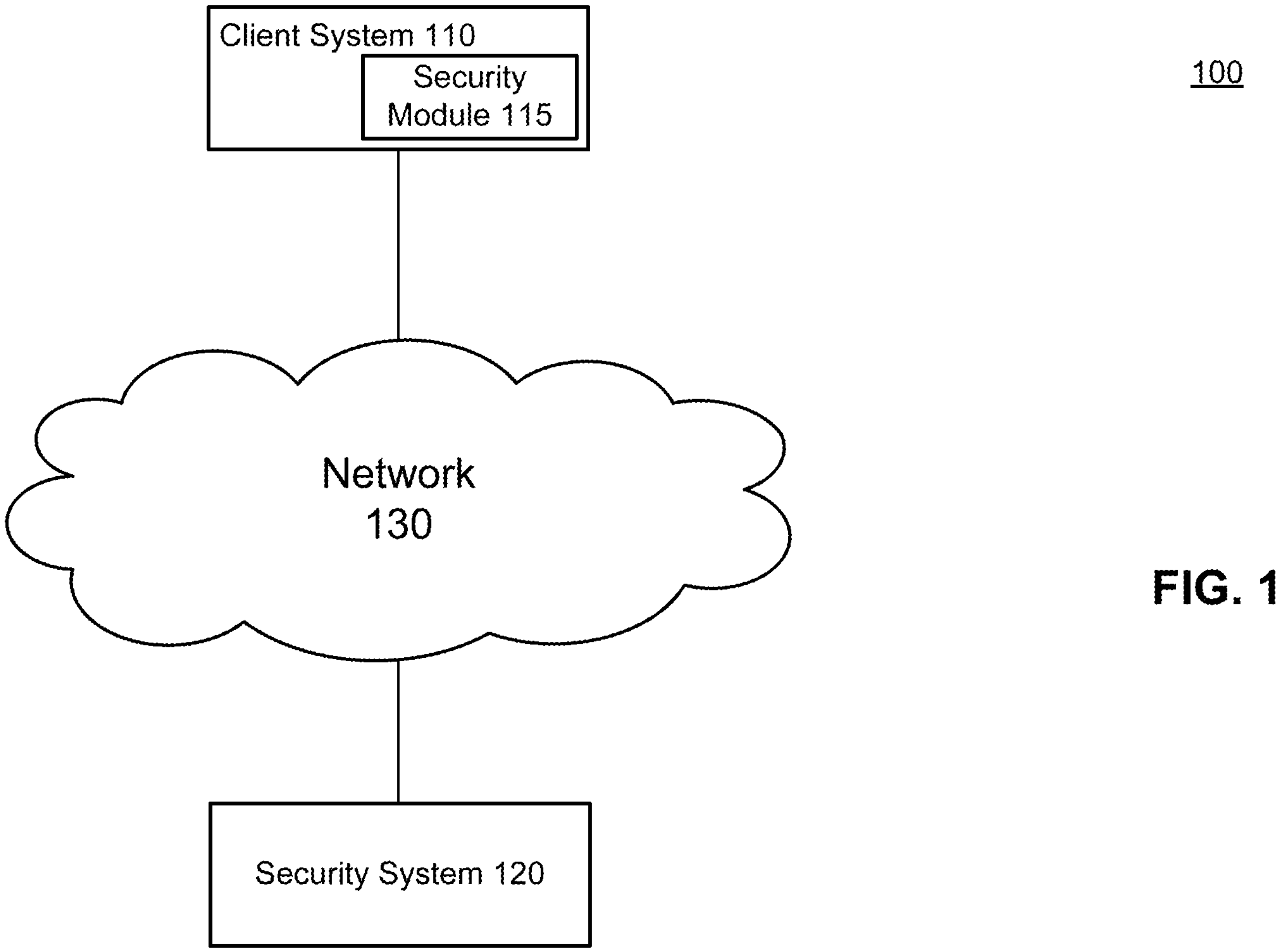
100
Client System 110
Security Module 115
Network 130
Security System 120

FIG. 1

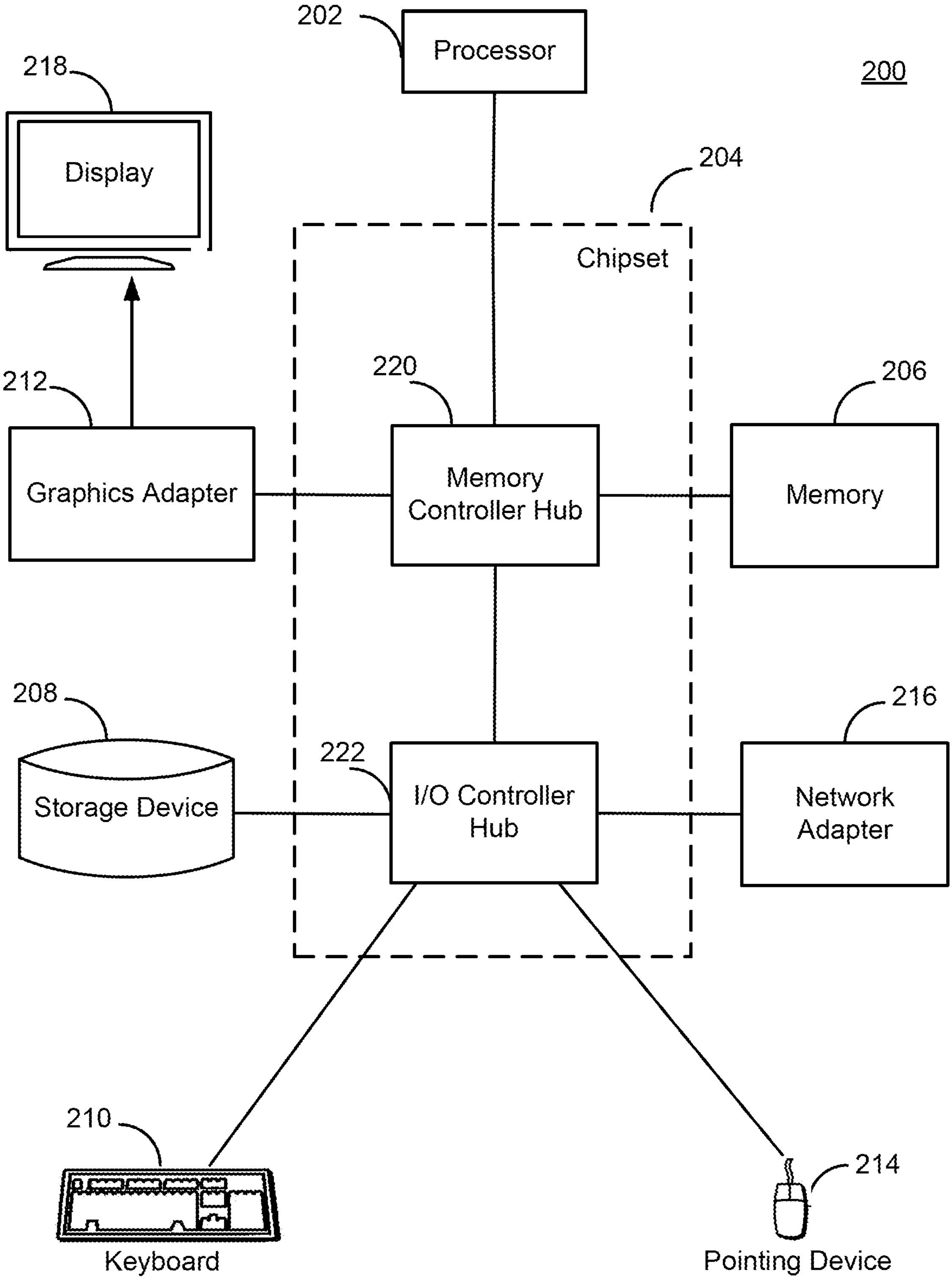
202
Processor
200
218
Display
204
Chipset
212
220
206
Graphics Adapter
Memory Controller Hub
Memory
208
216
222
Storage Device
I/O Controller Hub
Network Adapter
210
214
Keyboard
Pointing Device

FIG. 2

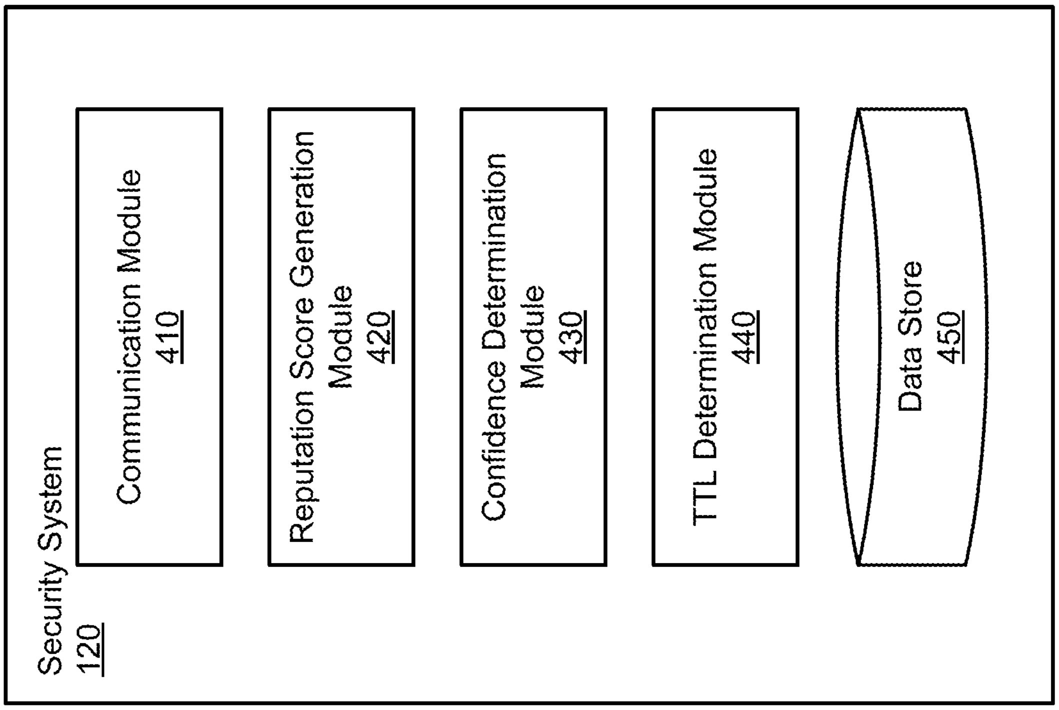
Security System
120
Communication Module
410
Reputation Score Generation Module
420
Confidence Determination Module
430
TTL Determination Module
440
Data Store
450

FIG. 4

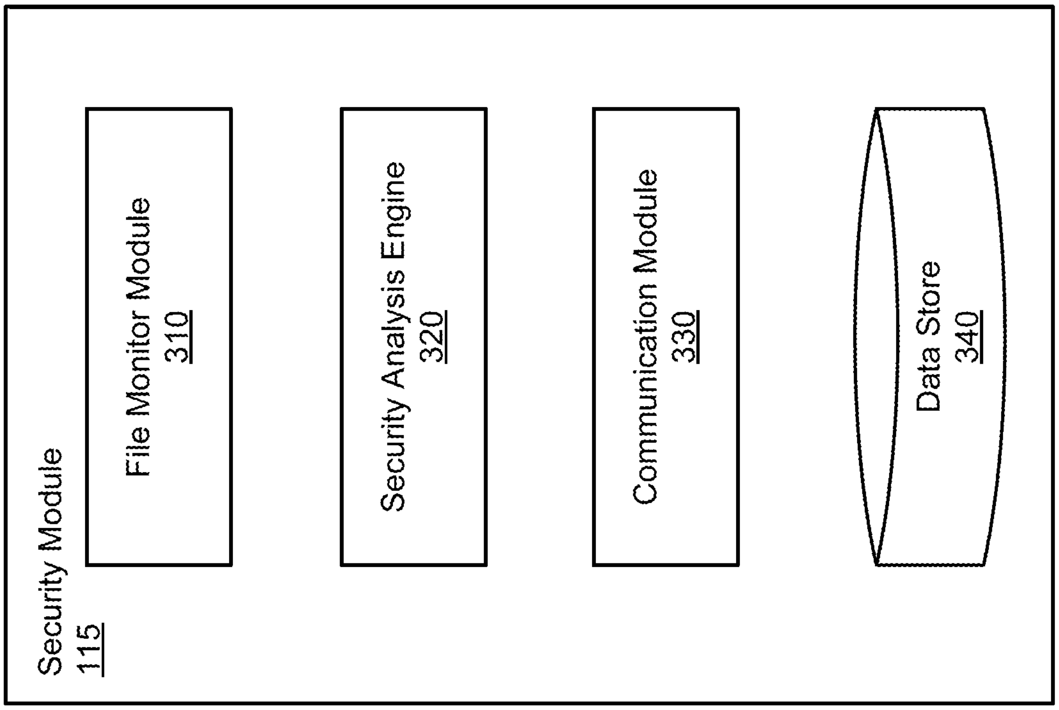
Security Module
115
File Monitor Module
310
Security Analysis Engine
320
Communication Module
330
Data Store
340

FIG. 3

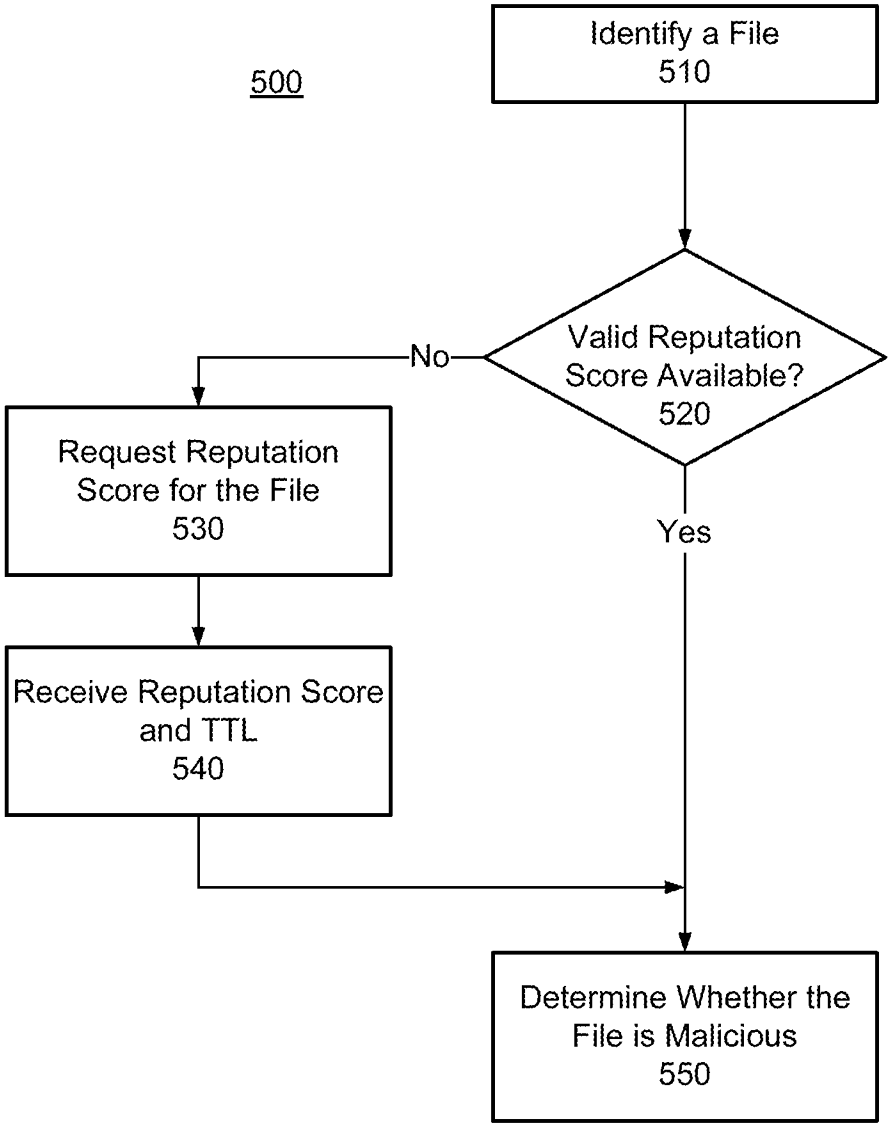
500
Identify a File
510
Valid Reputation
Score Available?
520
No
Yes
Request Reputation
Score for the File
530
Receive Reputation Score
and TTL
540
Determine Whether the
File is Malicious
550

FIG. 5

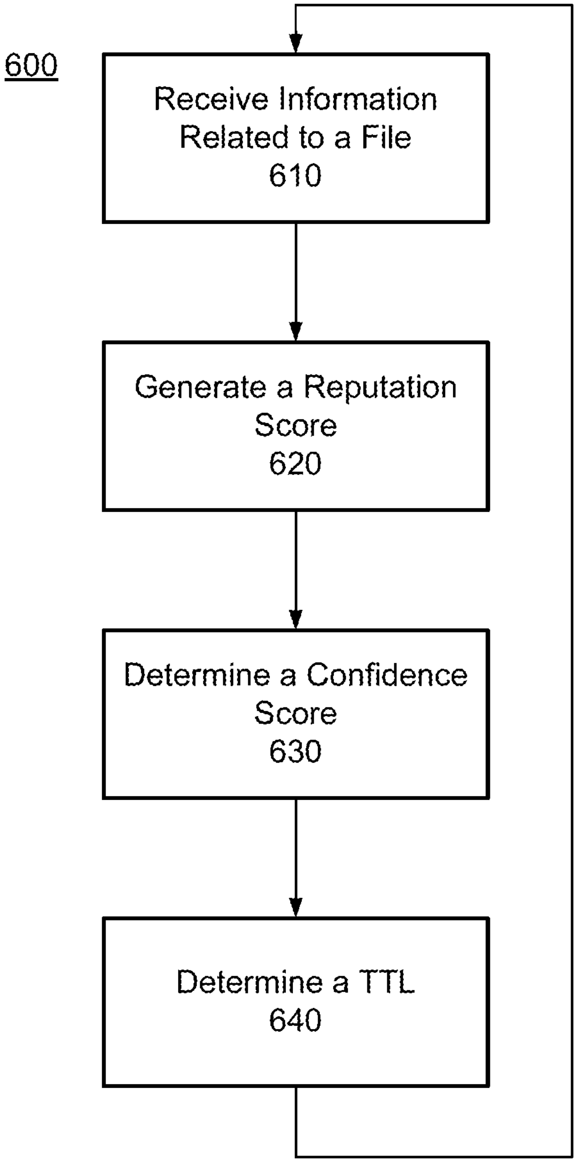
600
Receive Information
Related to a File
610
Generate a Reputation
Score
620
Determine a Confidence
Score
630
Determine a TTL
640

FIG. 6

INTERNATIONAL SEARCH REPORT

International application No
PCT/US2010/048475

A. CLASSIFICATION OF SUBJECT MATTER
INV. G06F21/00
ADD.

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)
G06F

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practical, search terms used)

EPO-Internal

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	US 2009/187442 A1 (KOHANIM GREGORY [US] ET AL) 23 July 2009 (2009-07-23) the whole document paragraph [0035]; claim 13 -----	1-20
A	JOSANG ET AL: "A survey of trust and reputation systems for online service provision", DECISION SUPPORT SYSTEMS, ELSEVIER SCIENCE PUBLISHERS, AMSTERDAM, NL, vol. 43, no. 2, 10 February 2007 (2007-02-10), pages 618-644, XP005744882, ISSN: 0167-9236, DOI: DOI:10.1016/J.DSS.2005.05.019 the whole document page 639, column 1, line - four last lines -----	1-20

□ Further documents are listed in the continuation of Box C.

☒ See patent family annex.

* Special categories of cited documents :

"A" document defining the general state of the art which is not considered to be of particular relevance

"E" earlier document but published on or after the international filing date

"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)

"O" document referring to an oral disclosure, use, exhibition or other means

"P" document published prior to the international filing date but later than the priority date claimed

"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art.

"&" document member of the same patent family

Date of the actual completion of the international search
18 November 2010

Date of mailing of the international search report
01/12/2010

Name and mailing address of the ISA/
European Patent Office, P.B. 5818 Patentlaan 2
NL - 2280 HV Rijswijk
Tel. (+31-70) 340-2040,
Fax: (+31-70) 340-3016

Authorized officer

Mäenpää, Jari

INTERNATIONAL SEARCH REPORT

Information on patent family members

International application No

PCT/US2010/048475

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
US 2009187442 A1	23-07-2009	WO 2009094086 A2	30-07-2009