(11) Application No. AU 2003243253 B2

(19) AUSTRALIAN PATENT OFFICE

(54) Title

Method and system for analyzing and addressing alarms from network intrusion detection systems

(51)[6] International Patent Classification(s)

G06F 21/00 (2006.01) 8BMEP **H04L**

H04L 29/06 (2006.01) 29/06

G06F 21/00 20060101ALI2005100

20060101AFI2005100 8BMEP

PCT/US03/15546

(21) Application No: 2003243253 (22) Application Date: 2003.05.14

(87) WIPO No: WO03/098413

(30) Priority Data

(31) Number	(32) Date	(33) Country
60/319,242	2002.05.14	US

(43) Publication Date : 2003.12.02

(43) Publication Journal Date : 2004.01.29

(71) Applicant(s)

Cisco Technology, Inc.

(72) Inventor(s)

Snapp, Steven R.; Campos, Stephen B.; Cohen, Nathan M.; Shanklin, Steven D.; Burke, Stephen A.; Rowland, Craig H.

(74) Agent/Attorney

Pizzeys, Level 2, Woden Plaza Offices, Woden, ACT, 2606

(56) Related Art

WO 2001/084270

WO 1999/057625

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

CORRECTED VERSION

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
27 November 2003 (27.11.2003)

PCT

(10) International Publication Number
WO 2003/098413 A1

(51) International Patent Classification[7]: G06F 1/00, H04L 29/06, 12/26

(21) International Application Number: PCT/US2003/015546

(22) International Filing Date: 14 May 2003 (14.05.2003)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:
60/319,242 14 May 2002 (14.05.2002) US

(71) Applicant: **CISCO TECHNOLOGY, INC.** [US/US]; 170 West Tasman Drive, San Jose, CA 95134-1706 (US).

(72) Inventors: **ROWLAND, Craig, H.**; 6908 Dogwood Hollow, Austin, TX 78750-8213 (US). **COHEN, Nathan, M.**; 1800 Aggie Lane, Austin, TX 78757-1834 (US). **SHANKLIN, Steven, D.**; 10015 Austral Cove, Austin, TX 78739-1719 (US). **SNAPP, Steven, R.**; 1332 Roadrunner Drive, Cedar Park, TX 78613-1822 (US). **CAMPOS, Stephen, B.**; 9622 Dalewood Drive, Austin, TX 78729-3554 (US). **BURKE, Stephen, A.**; 3 Scott Crescent, Austin, TX 78703-1724 (US).

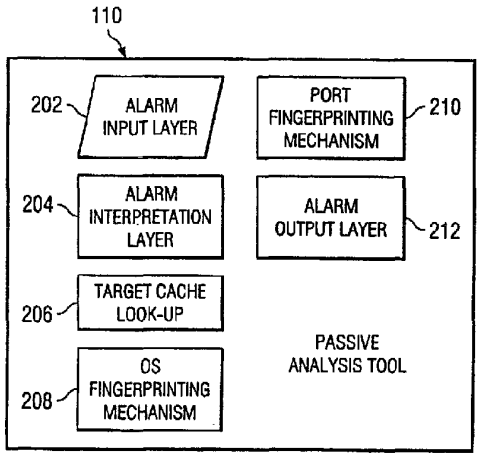
(74) Agent: **SHOWALTER, Barton, E.**; Baker & Botts, L.L.P., 2001 Ross Avenue, Dallas, TX 75201-2980 (US).

(81) Designated States *(national)*: AE, AG, AL, AM, AT, AU, AZ, BA, BB, BG, BR, BY, BZ, CA, CH, CN, CO, CR, CU, CZ, DE, DK, DM, DZ, EC, 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, MZ, NO, NZ, OM, PH, PL, PT, RO, RU, SC, SD, SE, SG, SK, SL, TJ, TM, TN, TR, TT, TZ, UA, UG, UZ, VC, VN, YU, ZA, ZM, ZW.

(84) Designated States *(regional)*: ARIPO patent (GH, GM, KE, LS, MW, MZ, SD, SL, SZ, TZ, UG, ZM, ZW), Eurasian patent (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European patent (AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HU, IE, IT, LU, MC, NL, PT, RO, SE, SI, SK, TR), OAPI patent (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, ML, MR, NE, SN, TD, TG).

[Continued on next page]

(54) Title: METHOD AND SYSTEM FOR ANALYZING AND ADDRESSING ALARMS FROM NETWORK INTRUSION DETECTION SYSTEMS



(57) Abstract: According to one embodiment of the invention, a method for analyzing and addressing alarms from network intrusion detection systems includes receiving an alarm indicating an attack on a target host may have occurred, automatically accessing the target host in response to the alarm, and identifying the presence of the attack on the target host.

WO 2003/098413 A1

Declarations under Rule 4.17:
— *as to applicant's entitlement to apply for and be granted a patent (Rule 4.17(ii)) for all designations*
— *as to the applicant's entitlement to claim the priority of the earlier application (Rule 4.17(iii)) for all designations*

Published:
— *with international search report*

(48) Date of publication of this corrected version:
6 May 2004

(15) Information about Correction:
see PCT Gazette No. 19/2004 of 6 May 2004, Section II

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.

METHOD AND SYSTEM FOR ANALYZING AND ADDRESSING ALARMS FROM NETWORK INTRUSION DETECTION SYSTEMS

TECHNICAL FIELD OF THE INVENTION

This invention relates generally to intrusion detection and more particularly to a method and system for analyzing and addressing alarms from network intrusion detection systems.

BACKGROUND OF THE INVENTION

Network Intrusion Detection Systems ("NIDS") are typically designed to monitor network activity in real-time to spot suspicious or known malicious activity and to report these findings to the appropriate personnel. By keeping watch on all activity, NIDS have the potential to warn about computer intrusions relatively quickly and allow administrators time to protect or contain intrusions, or allow the NIDS to react and stop the attack automatically. In the security industry, a NIDS may either be a passive observer of the traffic or an active network component that reacts to block attacks in real-time.

Because many NIDS are passive observers of the network traffic, they often lack certain knowledge of the attacking and defending host that makes it impossible to determine if an attack is successful or unsuccessful. Much like an eavesdropper overhearing a conversation between two strangers, NIDS very often lack knowledge of the context of the attack and, therefore, "alarm" on network activity that may not be hostile or relevant.

Some systems attempt to address this problem by building a static map of the network they are monitoring. This knowledge is usually built by scanning all the

systems on the network and saving the result to a database for later retrieval. This system is inadequate for most networks because the topology, types, and locations of network devices constantly change and requires the administrator to maintain a static database. Additionally, the stress of constantly scanning and keeping the network databases up to date is very intensive and may often slow down or cause network services to stop functioning.

SUMMARY OF THE INVENTION

According to one embodiment of the invention, a method for analyzing and addressing alarms from network intrusion detection systems includes receiving an alarm indicating an attack on a target host may have occurred, automatically accessing the target host in response to the alarm, and identifying the presence of the attack on the target host.

Some embodiments of the invention provide numerous technical advantages. Other embodiments may realize some, none, or all of these advantages. For example, according to one embodiment, the false alarm rate of network intrusion detection systems ("NIDS") is substantially reduced or eliminated, which leads to a lower requirement of personnel monitoring of NIDS to respond to every alarm. A lower false alarm rate is facilitated even though knowledge of the entire protected network is not required. Because knowledge of the network is not required, hosts may be dynamically added to the network.

According to another embodiment, critical attacks on a network are escalated and costly intrusions are

remediated. By actively investigating an attack on the actual targeted host, computer forensic evidence is collected, automated recovery from network attacks may be accomplished via clean-up of compromised hosts. In addition, the ability to perform a random or schedule scan of target hosts for successful attacks enhances security by double-checking the already deployed security technologies.

Other advantages may be readily ascertainable by those skilled in the art from the following figures, description, and claims.

BRIEF DESCRIPTION OF THE DRAWINGS

For a more complete understanding of the present invention and the advantages thereof, reference is now made to the following description taken in conjunction with the accompanying drawings, wherein like reference numbers represent like parts, and which:

FIGURE 1 is a schematic diagram illustrating a system for reducing the false alarm rate of network intrusion detection systems (NIDS) and for analyzing and addressing alarms from NIDS by utilizing a passive analysis tool and an active analysis tool according to one embodiment of the invention;

FIGURE 2 is a block diagram illustrating various functional components of the passive analysis tool of FIGURE 1 according to the one embodiment of the invention;

FIGURE 3 is a flowchart illustrating a method for reducing the false alarm rate of network intrusion detection systems according to one embodiment of the invention;

FIGURE 4 is a flowchart illustrating a method that may be used in conjunction with the method of FIGURE 3 according to one embodiment of the invention;

FIGURE 5 is a block diagram illustrating various functional components of the active analysis tool of FIGURE 1 according to the one embodiment of the invention; and

FIGURE 6 is a flowchart illustrating a method for analyzing and addressing alarms from network intrusion detection systems according to one embodiment of the invention.

DETAILED DESCRIPTION OF EXAMPLE EMBODIMENTS OF THE INVENTION

Embodiments of the invention are best understood by referring to FIGURES 1 through 6 of the drawings, like numerals being used for like and corresponding parts of the various drawings.

FIGURE 1 is a schematic diagram illustrating a system 100 for reducing the false alarm rate of a network intrusion detection system ("NIDS") 108 and for analyzing and addressing alarms from NIDS 108 by utilizing a passive analysis tool 110 and an active analysis tool 111 according to one embodiment of the present invention. In the illustrated embodiment, NIDS 108 is coupled to a link 106 that communicatively couples an unprotected network 102 with a protected network 104. System 100 also includes a network administrator 112 that utilizes passive analysis tool 110 and active analysis tool 111, as described in more detail below.

Unprotected network 102 may be any suitable network external to protected network 104. Examples of unprotected network 102 are the Internet, an Extranet,

and a local area network. Protected network 104 may be any suitable network, such as a local area network, wide area network, virtual private network, or any other suitable network desired to be secure from unprotected network 102. Link 106 couples unprotected network 102 to protected network 104 and may be any suitable communications link or channel. In one embodiment, communications link 106 is operable to transmit data in "packets" between unprotected network 102 and protected network 104; however, communications link 106 may be operable to transmit data in other suitable forms.

In one embodiment, NIDS 108 is any suitable network-based intrusion detection system operable to analyze data packets transmitted over communications link 106 in order to detect any potential attacks on protected network 104. NIDS 108 may be any suitable combination of hardware, firmware, and/or software. Typically, NIDS 108 includes one or more sensors having the ability to monitor any suitable type of network having any suitable data link protocol. In a particular embodiment, the sensors associated with NIDS 108 are operable to examine data packets on an IP ("Internet Protocol") network using any suitable protocol, such as TCP ("Transmission Control Protocol"), UDP ("User Datagram Protocol"), and ICMP ("Internet Control Message Protocol"). Upon detection of a possible attack on protected network 104, NIDS 108 is operable to generate an alarm indicating that an attack on protected network 104 may have occurred and may block the attack outright. This alarm is then transmitted to passive analysis tool 110 for analysis as described below.

According to the teachings of one embodiment of the present invention, passive analysis tool 110 receives an alarm from NIDS 108 and, using the information associated with the alarm, determines if an attack is real or a false alarm. Passive analysis tool 110 significantly lowers the false alarm rate for network intrusion detection systems, such as NIDS 108, in the network environment and lowers the requirement of personnel, such as network administrator 112, monitoring these systems to respond to every alarm. Details of passive analysis tool 110 are described in greater detail below in conjunction with FIGURES 2 through 4. Although illustrated in FIGURE 1 as being separate from NIDS 108, passive analysis tool 110 may be integral with NIDS 108 such that separate hardware is not required. In any event, NIDS 108 and passive analysis tool 110 work in conjunction with one another to analyze, reduce, or escalate alarms depending on the detected severity and accuracy of the attack. One technical advantage is that some embodiments of the invention may eliminate alarms targeted at the wrong operating system, vendor, application, or network hardware.

According to the teachings of another embodiment of the present invention, active analysis tool 111 receives a confirmed alarm from passive analysis tool 110 and automatically logs on to a target host 120 to investigate the detected attack and actively determine whether or not the attack was successful on target host 120. In addition, active analysis tool 111 may respond to the attacks by collecting relevant forensic evidence and initiating any suitable remedial measure. The process of actively investigating the attack on target host 120

assures a high degree of accuracy in a short response time. Forensic information may be copied from target host 120 to preserve its integrity against tampering, thereby allowing network administrator 112 or other suitable personnel to quickly analyze target host 120 and determine what changes were made after the attack happened and to use this information for later prosecution of the attacker. Details of active analysis tool 111 are described in greater detail below in conjunction with FIGURES 5 and 6. Although illustrated in FIGURE 1 as being separate from NIDS 108, active analysis tool 111 may be integral with NIDS 108 such that separate hardware is not required.

Network administrator 112 may be any suitable personnel that utilizes passive analysis tool 110 and active analysis tool 111 in order to monitor potential attacks on protected network 104 and respond thereto, if appropriate. Network administrator 112 typically has passive analysis tool 110 and active analysis tool 111 residing on his or her computer in order to receive filtered alarms from passive analysis tool, as denoted by reference numeral 114, and to receive information on investigated alarms, as denoted by reference numeral 115.

FIGURE 2 is a block diagram illustrating various functional components of passive analysis tool 110 in accordance with one embodiment of the present invention. The present invention contemplates more, less, or different components than those shown in FIGURE 2. These components may be pieces of software associated with passive analysis tool 110 that may be executed by a processor. In the illustrated embodiment, passive analysis tool 110 includes an alarm input layer 202, an

alarm interpretation layer 204, a target cache look-up 206, an operating system ("OS") fingerprinting mechanism 208, a port fingerprinting mechanism 210 and an alarm output layer 212. The general functions of each of these components are now described before a more detailed description of some functions of passive analysis tool 110 is undertaken in conjunction with FIGURES 3 and 4.

Alarm input layer 202 is generally responsible for accepting the alarm from NIDS 108 and passing it to other system components for analysis. In one embodiment, alarm input layer 202 accepts the alarm from NIDS 108 and determines if the alarm format is valid. If the alarm format is invalid, then the alarm is disregarded. If the alarm format is valid, then the alarm is sent to alarm interpretation layer 204. Alarm input layer 202 is preferably designed to be NIDS vendor independent so that it may accept alarms from multiple NIDS sources concurrently with no modification.

Generally, alarm interpretation layer 204 receives the alarm from alarm input layer 202 and performs an analysis on the alarm. In one embodiment, alarm interpretation layer 204 determines whether the alarm is from a supported NIDS vendor. If the alarm is not from a supported NIDS vendor, an alert is generated and the alarm is disregarded. If the alarm is from a supported NIDS vendor, then alarm interpretation layer 204 is responsible for determining the NIDS vendor alarm type, relevant operating system type being attacked (e.g., Microsoft Windows, Sun Solaris, Linux, UNIX, etc.), the source address, target network address, the alarm severity, the alarm description, and any other suitable parameters associated with the alarm. Some of this

information is used by passive analysis tool 110 to test if the alarm is real or false, as described in more detail below in conjunction with FIGURES 3 and 4.

Target cache look-up 206 indicates that a look-up is performed by passive analysis tool 110 in order to determine if the vulnerability of target host 120 has already been checked for the particular attack indicated by the alarm. The look-up may be performed in any suitable storage location, such as a local state table or database.

OS fingerprinting mechanism 208 performs a passive analysis of target host 120 to determine the operating system type of target host 120. Briefly, in one example, passive analysis tool 110 sends Internet Protocol ("IP") packets at target host 120 with special combinations of protocol flags, options, and other suitable information in the header in order to ascertain the operating system vendor and version number. Operating system fingerprinting is well known in the industry and, hence, is not described in detail herein. An advantage of this type of OS fingerprinting is that it requires no internal access to target host 120 other than remote network connectivity. OS fingerprinting mechanism 208 may build an operating system type within seconds of execution and stores this information in a suitable storage location for later retrieval and use.

Port fingerprinting mechanism 210 functions to identify a target port address stored in a suitable storage location when a host is added or deleted dynamically. Port fingerprinting mechanism 210 works in conjunction with OS fingerprinting mechanism 208 to determine, for example, if an attacked port on a

particular target host is active or inactive. This allows passive analysis tool 110 to quickly determine an attack could work. For example, an attack against TCP port 80 on a particular target host may be proven to have failed by checking the target host to see if port 80 is active.

Alarm output layer 212 is responsible for taking the analyzed data from passive analysis tool 110 and either escalating or de-escalating the alarm. In other words, alarm output layer 212 functions to report a valid alarm; i.e., that a particular target host is vulnerable to an attack. A valid alarm may be reported in any suitable manner, such as through the use of a graphical user interface or a log file, storing in a database, or any other suitable output. As described in further detail below, a confirmed alarm may be utilized by active analysis tool 111 to determine whether an attack on the target host worked or failed.

Additional description of details of functions of passive analysis tool 110, according to one embodiment of the invention, are described below in conjunction with FIGURES 3 and 4.

FIGURE 3 is a flowchart illustrating an example method for reducing the false alarm rate of network intrusion detection systems according to one embodiment of the present invention. The example method begins at step 300 where an alarm is received from NIDS 108 by passive analysis tool 110. Passive analysis tool 110 identifies the target address from the alarm at step 302. Passive analysis tool 110 then accesses a system cache at step 304 in order to determine if the identified target

host, such as target host 120, has already been checked for that particular attack type.

Accordingly, at decisional step 306, it is determined whether the target address has been found in the system cache. If the target address is found, then at decisional step 308, it is determined whether the cache entry time is still valid. In other words, if target host 120 was checked for a particular type of attack within a recent time period, then this information is stored temporarily in the system cache. Although any suitable time period may be used to store this information, in one embodiment, the information is stored for no more than one hour. If the cache entry time is still valid, then the method continues at step 310 where the OS fingerprint of target host 120 is received by passive analysis tool 110.

Referring back to decisional steps 306 and 308, if the target address is not found in the system cache or if the cache entry time is invalid for the target address that is found in the system cache, then the operating system fingerprint of target host 120 is obtained by passive analysis tool 110 using any suitable OS fingerprinting technique, as denoted by step 312. The operating system fingerprint is then stored in the system cache at step 314. The method then continues at step 310 where the operating system fingerprint of target host 120 is received.

The attack type and the operating system type of target host 120 are compared at step 316 by passive analysis tool 110. At decisional step 318, it is determined whether the operating system type of target host 120 matches the attack type. If there is a match,

then a confirmed alarm is reported by step 320. If there is no match, then a false alarm is indicated, as denoted by step 322. For example, if the attack type is for a Windows system and the operating system fingerprint shows a Windows host, then the alarm is confirmed. However, if the attack type is for a Windows system and the operating system fingerprint shows a UNIX host, then this indicates a false alarm. This then ends the example method outlined in FIGURE 3.

Although the method outlined in FIGURE 3 is described with reference to passive analysis tool 110 comparing an operating system type with an attack type, other suitable characteristics of the operating system may be compared to relevant characteristics of the attack type in order to determine if the alarm is real or false.

Thus, in one embodiment, passive analysis tool 110 screens out potential false alarms while not requiring knowledge of the entire protected network 104. Alarm inputs are received from a deployed NIDS, such as NIDS 108, and analyzed to determine if an attack is real or a false alarm. This is accomplished even though agents are not required to be installed on each computing device of the protected network 104.

FIGURE 4 is a flowchart illustrating an example method that may be used in conjunction with the example method outlined in FIGURE 3 in accordance with an embodiment of the present invention. The example method outlined in FIGURE 4 addresses the dynamic addition of hosts to protected network 104 in order that prior knowledge of the network is not required. The example method in FIGURE 4 begins at step 400 where a dynamic host configuration protocol ("DHCP") server 122 (FIGURE

1) is monitored by passive analysis tool 110. The present invention contemplates any suitable dynamic configuration protocol server being monitored by passive analysis tool 110. At step 402, lease activity is detected by passive analysis tool 110. A "lease" as used herein means that a host has been given a network address for a given period of time. At decisional step 404 it is determined whether a lease issue is detected or a lease expire is detected.

If a lease expire is detected by passive analysis tool 110, then the system cache is accessed, as denoted by step 406. At decisional step 408, it is determined whether the target address associated with the lease expire is found in the system cache. If the target address is found in the system cache, then the entry is purged, at step 410, from the system cache. Passive analysis tool 110 then continues to monitor DHCP server 122. If a target address is not found in the system cache, then the lease expire is disregarded, as denoted by step 412. Passive analysis tool 110 continues to monitor DHCP server 122.

Referring back to decisional step 404, if a lease issue has been detected, then the system cache is accessed, as denoted by step 414. At decisional step 416, it is determined whether the target address associated with the lease issue is found in the system cache. If the target address is found, then the entry is purged, at step 418. If the target address is not found in the system cache, then the method continues at step 420, as described below.

At step 420, the operating system fingerprint of a target host is obtained at step 420. The operating

system fingerprint is stored in the system cache, as denoted by step 422 for a particular time period. Passive analysis tool 110 then continues to monitor DHCP server 122.

The method outlined in FIGURE 4 saves considerable time and money and is more accurate than prior systems in which prior knowledge of the network is required. Passive analysis tool 110 may store entries for a user defined length of time that reduces the number of time operating system fingerprints need to be accomplished, which increases the efficiency of the network intrusion detection system. Another technical advantage is that resources are conserved and the impact on the protected network is low because target system profiles are built only when needed, effectively serving as a "just-in-time" vulnerability analysis.

FIGURE 5 is a block diagram illustrating various functional components of active analysis tool 111 in accordance with one embodiment of the present invention. The present invention contemplates more, less, or different components than those shown in FIGURE 5. These components may be pieces of software associated with active analysis tool 111 that may be executed by a processor. In the illustrated embodiment, active analysis tool 111 includes an alarm input layer 500, an alarm interpretation layer 502, an alarm investigation layer 504, an active analysis layer 506, an alarm response layer 508, and an alarm output layer 510. The general functions of each of these components are now described before a more detailed description of a function of active analysis tool 111 is undertaken in conjunction with FIGURE 6.

Alarm input layer 500 is generally responsible for accepting the alarm from NIDS 108 and passing it to other system components for analysis. In one embodiment, the functionality of alarm input layer 500 may be accomplished by alarm input layer 202 of passive analysis tool 110. In either case, alarm input layer 202 accepts the alarm from NIDS 108 and determines that the alarm format is valid. If the alarm format is invalid, then the alarm is disregarded. If the alarm format is valid, then the alarm is sent to alarm interpretation layer 502. Alarm input layer 500 is preferably designed to be NIDS vendor independent so that it may accept alarms from multiple NIDS sources concurrently with no modification.

Generally, alarm interpretation layer 502 receives the alarm from alarm input layer 500 and performs an analysis on the alarm. Again, in one embodiment, the functionality associated with alarm interpretation layer 502 may be accomplished by alarm interpretation layer 204 of passive analysis tool 110. In either case, alarm interpretation layer 502 determines whether the alarm is from a supported NIDS vendor. If the alarm is not from a supported NIDS vendor, an alert is generated and the alarm is disregarded. If the alarm is from a supported NIDS vendor, then alarm interpretation layer 502 is responsible for determining the NIDS vendor alarm type, relevant operating system type being attacked, the source address, the target network address, the alarm severity, the alarm description, and any other suitable parameters associated with the alarm. Some of this information is used by active analysis tool 111 in order to log onto target host 120 in addition to determine what pieces of information to analyze on target host 120. For example,

an attack type commonly has files that are affected if it is a successful attack (i.e., web log files for web-based attacks). This is described in more detail below in conjunction with FIGURE 6.

Alarm investigation layer 504 indicates that a lookup is performed by active analysis tool 111 in order to determine if target host 120 has already been investigated for the particular attack indicated by the alarm within a particular time period. The lookup may be performed in any suitable storage location, such as a local state table or database.

Active analysis layer 506 performs an active analysis of target host 120 to determine whether or not the attack actually worked. Briefly, in one embodiment, active analysis tool 111 logs onto target host 120 by using an authenticated connection based on the operating system type of target host 120. The authenticated connection may take any suitable form; however, some examples are as follows: Microsoft SMB or CIFS protocol for Microsoft Windows systems, secure shell (SSH) encrypted login for UNIX systems, Telnet unencrypted login for UNIX systems, and user defined authentication methods. The use of native authentication means that no remote agent is needed on target host 120s of protected network 104. This makes deployment of active analysis tool 111 very fast and requires only that network administrator 112 provide top level login privileges and credentials to active analysis tool 111 instead of spending time deploying remote agents throughout protected network 104.

Active analysis layer 506, once logged on to target host 120, analyzes and collects information regarding the

attack to determine whether or not the attack worked. For example, active analysis layer 506 may facilitate the analyzing of audit trails, system binaries, system directories, registry keys, configuration files, or other suitable analysis to determine whether or not the attack worked. The pieces of information on target host 120 that are analyzed are based upon the type of alarm detected. In one embodiment, it is a matter of mapping the detected type of attack to the relevant traces left by the attacker to determine if the attack worked. The determination of whether or not the attack worked is stored in a suitable storage location.

Alarm response layer 508 collects information on successfully executed attacks and determines what actions to take as a result of the successful attack. The actions may take any suitable form and may be configured by network administrator 112. For example, actions may include collecting logs, disabling users, blocking attacking hosts, disabling computer services, or any other suitable user defined action. Collecting information is copied from target host 120 and stored in a suitable storage location to preserve their integrity against tampering. Alarm response layer 508 may also initiate cleanup actions on target host 120 to remove the attacker and related exploited components. The information may also be used to build an attacker profile that can be taken and used to search other hosts on protected network 104 for possible compromise that use similar attack methods.

Alarm output layer 510, in one embodiment, is responsible for taking the information from the investigation of target host 120 and sending it to an

output stream. The output stream may be any suitable output, such as a log file, a graphical user interface, a native NIDS device, a memory, or any other suitable user defined output.

Additional description of details of functions of active analysis tool 111, according to one embodiment of the invention, are described below in conjunction with FIGURE 6.

FIGURE 6 is a flowchart illustrating a method for analyzing and addressing alarms from network intrusion detection systems according to one embodiment of the present invention. The example method starts at step 600 where a confirmed alarm is received from passive analysis tool 110 or other suitable source. Active analysis tool 111 identifies target host 120 at step 602. Active analysis tool 111 then accesses a system cache at step 604 in order to determine if target host 120 has already been investigated for that particular attack type.

Accordingly, at decisional step 606, it is determined whether investigation data has been found for target host 120. If investigation data is found in the system cache, then, at decisional step 608, it is determined whether a cache entry time for the investigation data is still valid. In other words, if a particular target host was already investigated for the particular type of attack within a recent time period, then this investigation data is stored temporarily in the system cache. Although any suitable time period may be used, in one embodiment, the information is stored for no more than one hour. If a cache entry time is still valid, then the method continues at step 610 where the investigation data is accessed by active analysis tool

111. And at step 612, the investigation data is reported to network administrator 112 or other suitable personnel.

Referring back to decisional step 606, if the investigation data was not found in the system cache or if the cache entry time is invalid for a particular target host that is found in the system cache, then target host 120 is accessed, at step 614, by active analysis tool 111. The accessing of target host 120 is accomplished as described above in conjunction with active analysis layer 506 in FIGURE 5. Once target host 120 is accessed, the presence of the attack on target host 120 is identified at step 616. At decision step 618, it is determined whether the attack was successful. Determining whether or not the attack was successful is accomplished by active analysis layer 506 (FIGURE 5). As described above, a set of rules determines what steps should be taken on target host 120 to verify if the attack worked or failed based on the alarm type. This may include the analysis of audit trails, system binaries, system directories, registry keys, configuration files, or any other suitable user defined checks. It may also include checking for suspicious files, directories, users, processes, or other irregular activity. If it is determined at step 618 that the attack is not successful, the investigation data is stored in the system cache 620. The investigation data is then reported at step 612 to network administrator 112 or the suitable personnel.

If it is determined at step 618 that the attack is successful, then forensic information regarding the attack is collected at step 622 and stored at step 624. As described above, this forensic information may be

copied from target host 120 to preserve its integrity against tampering to allow network administrator 112 or other suitable personnel to analyze target host 120 and determine what changes were made after the compromise happened and to use to collect information for later prosecution of the attacker. At step 626, remedial measures are initiated by active analysis tool 111. As described above, this may include collecting logs, disabling users, blocking an attacking host, disabling computer services, or any other suitable user defined action. This then ends the example method that is outlined in FIGURE 6.

Thus, in one embodiment, active analysis tool 111 investigates attacks on a target host in order to determine whether the attack worked or failed, collects forensic information regarding the attack, and potentially initiates remedial measures. This is done even though agents are not required to be installed on each computing device of protected network 104.

Although embodiments of the invention are described with some examples, various modifications may be suggested to one skilled in the art. The present invention intends to encompass those modifications as they fall within the scope of the claims.

Throughout this specification and the claims which follow, unless the context requires otherwise, the word "comprise", and variations such as "comprises" and "comprising", will be understood to imply the inclusion of a stated integer or step or group of integers or steps but not the exclusion of any other integer or step or group of integers or steps.

2003243253 11 Nov 2004

The reference to any prior art in this specification is not, and should not be taken as, an acknowledgement or any form of suggestion that the prior art forms part of the common general knowledge in Australia.

2003243253 02 Nov 2009

WHAT IS CLAIMED IS:

1. A method for analyzing and addressing alarms from network intrusion detection systems, comprising:

receiving from network intrusion detection systems an alarm at an analysis tool located outside a protected network, the alarm indicating an attack on a target host inside the protected network may have occurred;

automatically accessing the target host in response to the alarm by logging on to the target host from outside the protected network using an authenticated connection based on an operating system type of the target host; and

automatically identifying the presence of the attack on the target host.

2. The method of Claim 1, further comprising:

automatically accessing a storage location in response to the alarm;

determining whether investigation data for the target host already exists in the storage location;

if the investigation data exists and the investigation data is still valid, then accessing the investigation data; and

if the investigation data does not exist or if the investigation data exists but is invalid, then:

identifying whether the attack was successful; and

identifying an audit trail of the attack on the target host.

3. The method of Claim 1 or 2, further comprising automatically identifying whether the attack was successful.

2003243253 02 Nov 2009

4. The method of Claim 2 or 3, further comprising storing whether the attack was successful in storage location for a time period.

5. The method of Claim 3, wherein automatically identifying whether the attack was successful comprises automatically identifying an audit trail of the attack on the target host.

6. The method of any of Claims 2 to 5 wherein the audit trail is selected from the group consisting of a modification of one or more registry keys, an entry in an access log file, a modification of a configuration file, a modification of a system directory, a modification of a system binary, a suspicious system process, and a suspicious file.

7. The method of any of Claims 2 to 6, further comprising storing the audit trail in a, or the, storage location if the attack was successful.

8. The method of any of Claims 2 to 7, further comprising initiating a remedial measure if the attack was successful.

9. The method of Claim 8, wherein the remedial measure is selected from the group consisting of blocking an attacking host, disabling a target host, disabling a computer service, and alerting a network administrator.

10. The method of any preceding claim, further comprising receiving top level login privileges in order to access the target host.

2003243253 02 Nov 2009

11. The method of any preceding claim, further comprising:

before automatically accessing the target host, automatically accessing a storage location;

determining whether investigation data for the target host already exists in the storage location; and

if the investigation data exists, then determining whether the investigation data is still valid; and

if the investigation data does not exist, then continuing with the automatically accessing step.

12. The method of any preceding claim, further comprising determining whether the target host is vulnerable to the attack.

13. The method of Claim 12, wherein determining whether the target host is vulnerable to the attack comprises:

identifying characteristics of the alarm, including at least an attack type and a target address of the target host;

querying the target host for an operating system fingerprint;

receiving the operating system fingerprint that includes the operating system type from the target host;

comparing the attack type to the operating system type; and

indicating whether the target host is vulnerable to the attack based on the comparison.

14. The method of any of Claims 2 to 13, further comprising:

after receiving the alarm, determining whether a format for the alarm is valid; and

if the format is not valid, then disregarding the alarm;

otherwise if the format is valid, then continuing the method with the automatically accessing step.

15. The method of any preceding claim comprising:

monitoring a dynamic configuration protocol server;

detecting that a lease issue has occurred for a new target host;

accessing the storage location;

determining whether an operating system fingerprint for the new target host already exists in the storage location; and

if the operating system fingerprint for the new target host does not exist, then:

querying the new target host for the operating system fingerprint;

receiving the operating system fingerprint from the new target host; and

storing the operating system fingerprint of the new target host in the storage location for a time period; and

if the operating system fingerprint for the new target host does exist, then:

purging the existing operating system fingerprint for the new target host from the storage location;

querying the new target host for a new operating system fingerprint;

receiving the new operating system fingerprint from the new target host; and

storing the new operating system fingerprint of the new target host in the storage location for a time period.

2003243253 02 Nov 2009

16. The method of any preceding claim, further comprising:

monitoring a dynamic configuration protocol server;

detecting that a lease expire has occurred for an existing target host; accessing the storage location;

determining whether an operating system fingerprint for the existing target host already exists in the storage location; and

if the operating system fingerprint for the existing target host does exist, then purging the existing operating system fingerprint for the existing target host from the storage location.

2003243253 02 Nov 2009

17. A system for analyzing and addressing alarms from network intrusion detection systems, comprising:

means for receiving from network intrusion detection systems an alarm at an analysis tool located outside a protected network, the alarm indicating an attack on a target host inside the protected network may have occurred;

means for automatically accessing the target host in response to the alarm, the means for accessing comprising means for logging on to the target host from outside the protected network by using an authenticated connection based on an operating system type of the target host; and

means for automatically identifying the presence of the attack on the target host.

18. A system according to Claim 17, further comprising:

a network intrusion detection system (NIDS) operable to transmit an alarm indicating an attack on a target host may have occurred;

and wherein the means for receiving, means for accessing and means for identifying comprises a software program located outside a protected network and embodied in a computer readable medium.

19. The system of Claim 17 or 18, further comprising means for automatically identifying whether the attack was successful.

20. The system of Claim 19, further comprising means for storing whether the attack was successful for a time period.

2003243253 02 Nov 2009

21. The system of Claim 19 or 20, wherein means for identifying whether the attach was successful comprises means for automatically identifying an audit trail of the attack on the target host.

22. The system of Claim 21, wherein the audit trail is selected from the group consisting of a modification of one or more registry keys, an entry in an access log file, a modification of a configuration file, a modification of a system directory, a modification of a system binary, a suspicious system process, and a suspicious file.

23. The system of Claim 21 or 22, further comprising means for storing the audit trail if the attack was successful.

24. The system of any of Claims 19 to 23, further comprising means for initiating a remedial measure if the attack was successful.

25. The system of Claim 24, wherein the remedial measure is selected from the group consisting of blocking an attacking host, disabling a target host, disabling a computer service, and alerting a network administrator.

26. The system of any of Claims 17 to 25, further comprising means for receiving top level login privileges in order to access the target host.

27. The system of any of Claims 17 to 26, further comprising:

means for automatically accessing a storage location before accessing the target host;

2003243253 02 Nov 2009

means for determining whether investigation data of the target host already exists in the storage location; and

if the investigation data exists, then means for determining whether the investigation data is still valid.

28. A method for analyzing and addressing alarms from network intrusion detection systems substantially as herein described.

29. A system for analyzing and addressing alarms from network intrusion detection systems substantially as herein described.

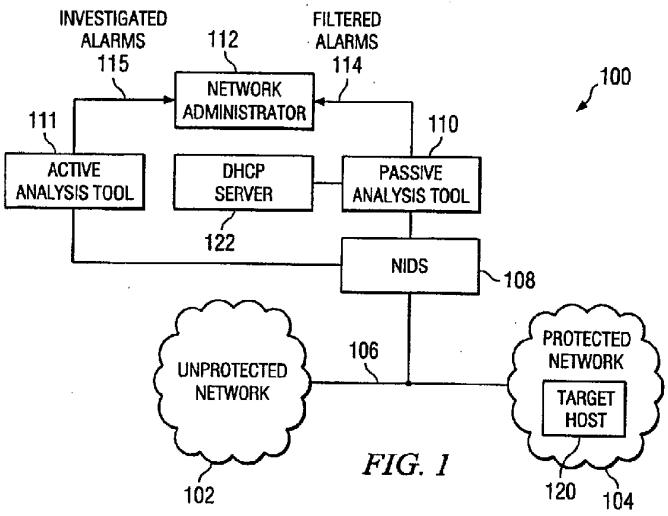

INVESTIGATED ALARMS
115
112
FILTERED ALARMS
114
100
NETWORK ADMINISTRATOR
111
110
ACTIVE ANALYSIS TOOL
DHCP SERVER
PASSIVE ANALYSIS TOOL
122
NIDS
108
UNPROTECTED NETWORK
106
PROTECTED NETWORK
TARGET HOST
102
120
104

FIG. 1

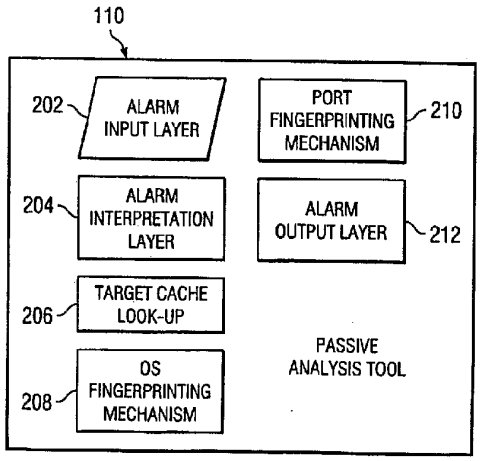

110
202
ALARM INPUT LAYER
PORT FINGERPRINTING MECHANISM
210
204
ALARM INTERPRETATION LAYER
ALARM OUTPUT LAYER
212
206
TARGET CACHE LOOK-UP
208
OS FINGERPRINTING MECHANISM
PASSIVE ANALYSIS TOOL

FIG. 2

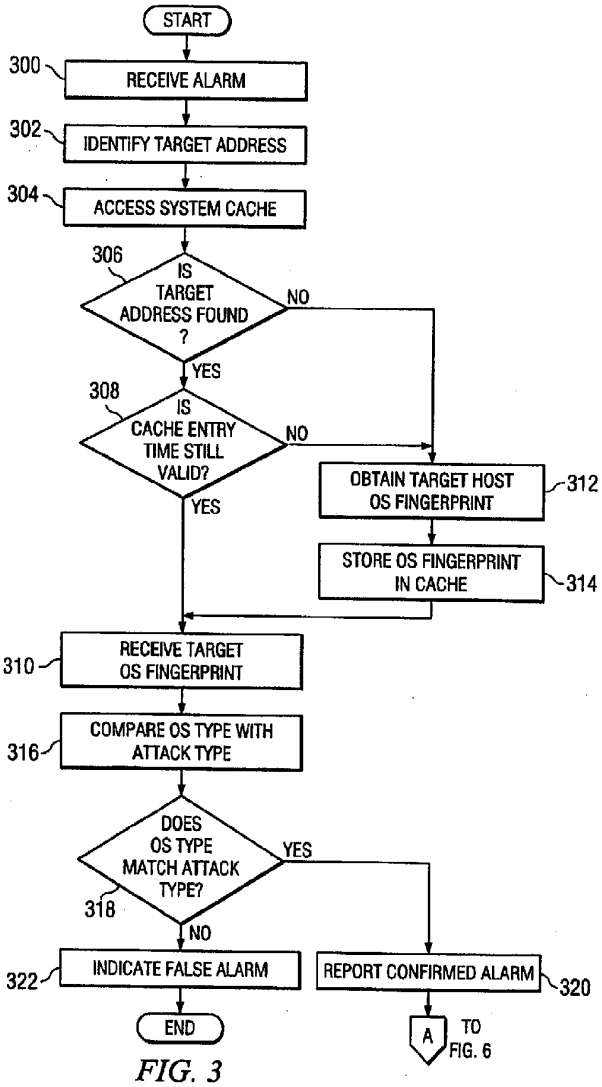

START
300
RECEIVE ALARM
302
IDENTIFY TARGET ADDRESS
304
ACCESS SYSTEM CACHE
306
IS TARGET ADDRESS FOUND ?
NO
YES
308
IS CACHE ENTRY TIME STILL VALID?
NO
YES
OBTAIN TARGET HOST OS FINGERPRINT
312
STORE OS FINGERPRINT IN CACHE
314
310
RECEIVE TARGET OS FINGERPRINT
316
COMPARE OS TYPE WITH ATTACK TYPE
DOES OS TYPE MATCH ATTACK TYPE?
318
YES
NO
322
INDICATE FALSE ALARM
REPORT CONFIRMED ALARM
320
END
A
TO FIG. 6

FIG. 3

FIG. 4

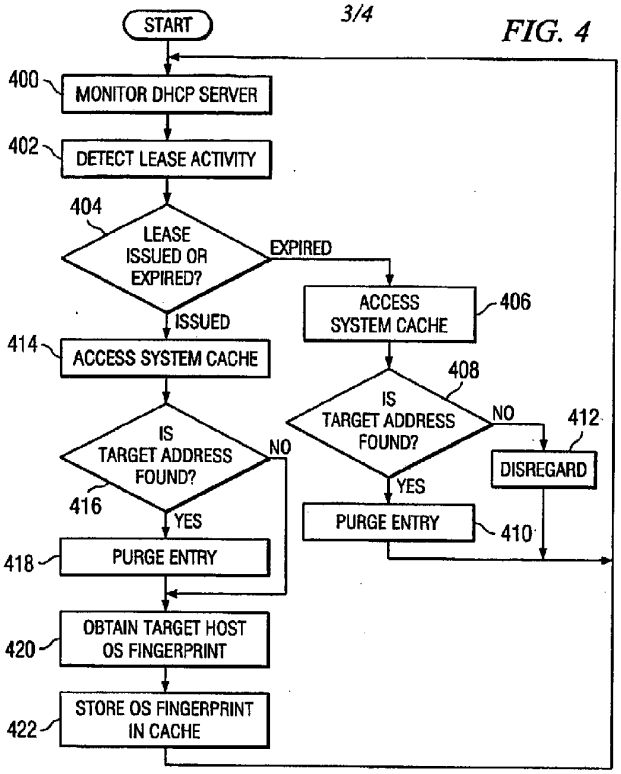
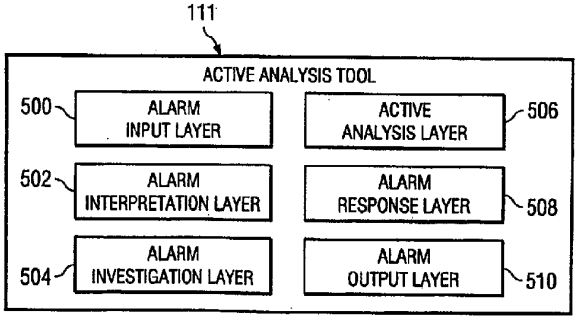

START
400
MONITOR DHCP SERVER
402
DETECT LEASE ACTIVITY
404
LEASE ISSUED OR EXPIRED?
EXPIRED
ISSUED
406
ACCESS SYSTEM CACHE
408
IS TARGET ADDRESS FOUND?
NO
412
DISREGARD
YES
PURGE ENTRY
410
414
ACCESS SYSTEM CACHE
IS TARGET ADDRESS FOUND?
NO
416
YES
418
PURGE ENTRY
420
OBTAIN TARGET HOST OS FINGERPRINT
422
STORE OS FINGERPRINT IN CACHE

FIG. 5


111
ACTIVE ANALYSIS TOOL
500
ALARM INPUT LAYER
ACTIVE ANALYSIS LAYER
506
502
ALARM INTERPRETATION LAYER
ALARM RESPONSE LAYER
508
504
ALARM INVESTIGATION LAYER
ALARM OUTPUT LAYER
510

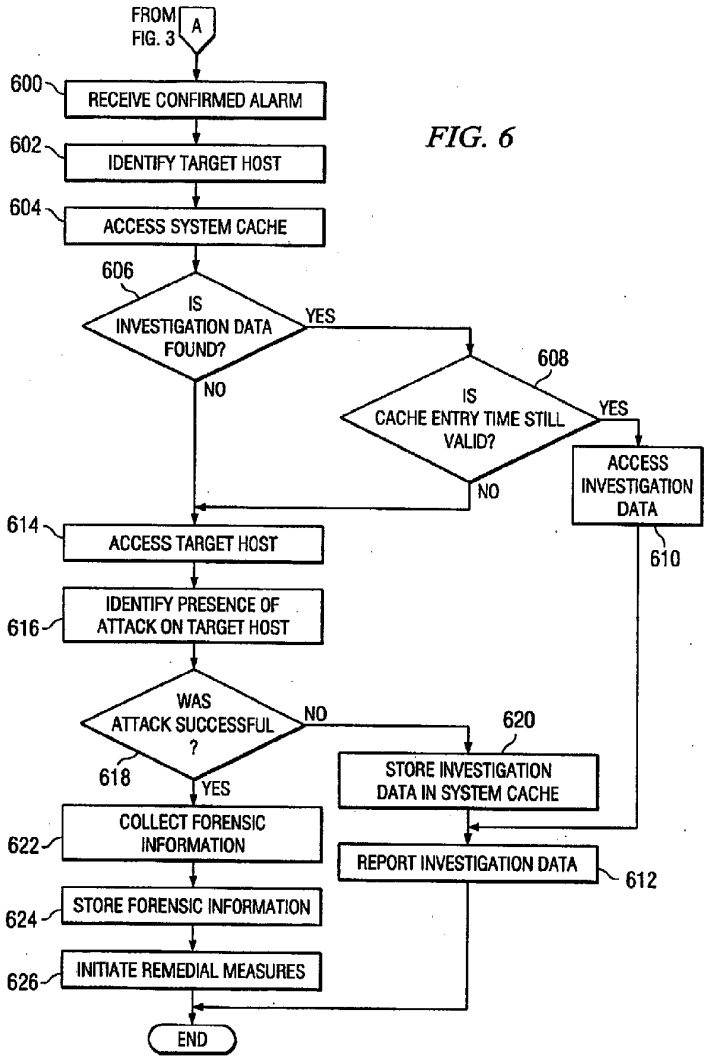

FROM FIG. 3
A
600
RECEIVE CONFIRMED ALARM
602
IDENTIFY TARGET HOST
604
ACCESS SYSTEM CACHE
606
IS INVESTIGATION DATA FOUND?
YES
NO
608
IS CACHE ENTRY TIME STILL VALID?
YES
NO
ACCESS INVESTIGATION DATA
610
614
ACCESS TARGET HOST
616
IDENTIFY PRESENCE OF ATTACK ON TARGET HOST
WAS ATTACK SUCCESSFUL ?
618
NO
YES
620
STORE INVESTIGATION DATA IN SYSTEM CACHE
622
COLLECT FORENSIC INFORMATION
REPORT INVESTIGATION DATA
612
624
STORE FORENSIC INFORMATION
626
INITIATE REMEDIAL MEASURES
END

FIG. 6