

**Method for detecting unauthorized port scanning in a computer network,
associated computer program product and device**

The present invention relates to a method for detecting unauthorized port scanning
5 in a computer network. Such a method can in particular be used to prevent a computer
attack in such a network.

The computer network is made up of a plurality of communication terminals able to
communicate with one another via a plurality of communication ports. In general, each
communication port is associated with one or more services, generally a software
10 program of the corresponding terminal, and allows remote software to access this
software in order to share digital data via the computer network.

In order to manage access to the associated software, each communication port
comprises a so-called open operating state in which it is able to receive incoming
connections from a remote terminal, and a so-called closed operating state in which it
15 systematically rejects any incoming connection from such a terminal.

Some communication ports further comprise a so-called filtered operating state in
which they are able to receive only incoming connections from a predetermined list of
terminals in the network. For terminals that are not part of this list, a port in the filtered
state sends nothing in response to the initiated connection.

In the computer field, port scanning is a technique making it possible to determine,
20 for a given set of IP (Internet Protocol) addresses, those which correspond to existing
terminals, as well as information relative to those terminals from a remote terminal via the
network. This information can in particular comprise the version of the operating system
used in each of the target terminals, the names and versions of the different services
25 offered by these terminals and all of the open and closed ports for these terminals.

This technique generally consists of sending each port of the target terminals a
data stream, and next analyzing the received streams in response to the sent streams or
the absence of such responses. In particular, during this analysis, the ports having
rejected or ignored the initiated connection are considered to be closed. The ports having
30 confirmed the initiated connection are then considered to be open.

Port scanning is for example used by network administrators to monitor security,
but also by computer pirates in order to prepare computer attack on at least some
terminals of the network. The latter case involves unauthorized port scanning.

Unauthorized port scanning is thus widely used by computer pirates to prepare a
35 computer attack via one or more open ports determined during that scan.

Unauthorized port scanning detection is therefore an important technical problem in combating computer piracy.

Different methods for detecting such scanning already exist in the state of the art.

The majority of these methods propose to detect port scans in the network in a centralized manner, i.e., from a central terminal at least partially dedicated to that purpose.

This is particularly the case for private networks connected to one or more public networks via a central terminal of the firewall type managing the entire incoming or outgoing stream in or from the private network.

According to the existing methods, in order to detect port scans, the central terminal analyzes all of the connections initiated by same terminal identified by its IP address, toward a large number of terminals and/or ports in the private network.

Thus, for example, when the number of these connections exceeds a predetermined threshold, the central terminal either raises an alarm to warn network administrators of potential port scanning or filters subsequent connections from that address.

However, the implementation of the existing methods requires that the central terminal dedicate a large quantity of available hardware resources to storing and analyzing the information relating to the set of connections initiated toward all of the other terminals depending on that central terminal. This generally does not make it possible to keep a long enough history of these connections in particular to detect port scans spread out over time. These methods often also do not make it possible to detect "distributed" scans, i.e., coming from several terminals.

The present invention aims to propose a method for detecting unauthorized port scanning making it possible to resolve these drawbacks.

To that end, the invention relates to a method for detecting unauthorized port scanning in a computer network in particular to avoid a computer attack, the computer network being made up of a plurality of communication terminals each defining a plurality of communication ports and able to communicate with one another via these communication ports, each terminal being associated with a unique terminal identifier in the network, each port being associated with a unique port identifier within the corresponding terminal, at least some of the terminals, called initiating terminals, being able to initiate a computer connection with at least one other terminal, called receiving terminal, via at least one port of the receiving terminal, each receiving terminal being able to reject a connection initiated by an initiating terminal.

The method is implemented by terminals forming an identified group of terminals and including the following steps:

- detection, by each receiving terminal in the group, of each connection initiated by an initiating terminal via at least one port of that receiving terminal;

5 - identification, by each receiving terminal of the group, of rejected connections among all of the initiated connections;

10 - for each rejected connection, generating security information comprising a first datum and a second datum respectively corresponding to the identifiers of the receiving terminal and one of its ports corresponding to this rejected connection, and adding the generated security information in a security database shared between the terminals in the group, and

- analyzing the shared security information database to determine unauthorized port scans.

15 According to other advantageous aspects of the invention, the method comprises one or more of the following features, considered alone or according to all technically possible combinations:

- each rejected connection corresponds to an initiated connection toward a closed port or toward a port comprising a filter refusing the connection with the corresponding initiating terminal;

20 - each terminal of the group further includes a storage unit able to store at least part of the security information database;

- the security information database is shared in the group using so-called "peer-to-peer"-type technology;

25 - the security information database comprises a correspondence graph whose nodes are provided by the first data and the second data from each added item of security information;

30 - each added item of security information corresponds to an edge between two nodes formed by the first datum and the second datum coming from the security information, a set of nodes connected via one or more edges forming a connected component of the correspondence graph;

35 - the addition of an item of security information comprises the following sub-steps: creating a new node formed by the first datum of this security information if such a node does not exist in the correspondence graph; creating a new node formed by the second datum of this security information if such a node does not exist in the correspondence graph; identifying a first connected component of the correspondence graph containing a node formed by the first datum; identifying a second connected component of the

correspondence graph containing a node formed by the second datum; and, if the first connected component and the second connected component are separate, merging the first connected component with the second connected component by creating an edge between the nodes formed by the first datum and the second datum;

5 - the analysis of the security information database comprises determining the number of nodes of at least one connector component of the correspondence graph;

 - the at least one connector component for which the number of nodes is determined corresponds to a component obtained after merging the first connected component and the second connected component;

10 - port scanning is determined when the number of identified nodes exceeds a predetermined threshold;

 - each security information item further comprises the identifier of the initiating terminal having initiated the rejected connection corresponding to the security information;

15 - the security information database further comprises the number of occurrences of each security information item in that database;

 - the step for updating the security database making it possible to erase the oldest security information contained in that database.

The present invention also relates to a computer program product including software instructions which, when implemented by a computer, carry out this method.

20 The present invention also relates to an analysis device for detecting unauthorized port scanning in a computer network in particular to prevent a computer attack, the computer network being made up of a plurality of communication terminals each defining a plurality of communication ports and able to communicate with one another via these communication ports, each terminal being associated with a unique terminal identifier in
25 the network, each port being associated with a unique port identifier within the corresponding terminal, at least some of the terminals, called initiating terminals, being able to initiate a computer connection with at least one other terminal, called receiving terminal, via at least one port of this receiving terminal, each receiving terminal being able to reject the connection initiated by an initiating terminal.

30 The analysis device is able to detect each connection initiated by an initiating terminal via at least one port of this receiving terminal; identify rejected connections among the set of initiated connections; for each rejected connection, generate security information comprising a first datum and a second datum respectively corresponding to the identifiers of the receiving terminal and one of its ports corresponding to this rejected
35 connection, and add the generated security information in a shared security database;

and analyze the shared security information database to determined unauthorized port scans.

The invention will be better understood upon reading the following description, provided solely as an example, and done in reference to the appended drawings, in which:

- figure 1 is a diagrammatic view of a computer network including a plurality of communication terminals, at least some of these terminals implementing the port scanning detection method according to the invention;

- figure 2 is a detailed diagrammatic view of a terminal of figure 1;

- figure 3 is a detailed diagrammatic view of another terminal of figure 1, the terminal including at least part of a security information database;

- figure 4 is a diagrammatic view of the information database of figure 3;

- figure 5 is a flowchart of the method according to the invention; and

- figure 6 is a flowchart of one step of the method of figure 5 according to one advantageous aspect of the invention.

The computer network 10 includes a plurality of communication terminals 11, 12A to 12E connected to one another by wired or wireless computer links, and able to communicate with one another by sharing computer data.

In the example embodiment shown in figure 1, the computer network 10 comprises a private network 16 formed by the terminals 12A to 12E, and a public network 18 connected to the private network 16 and in particular including the terminal 11.

The public network 18 is for example the Internet. Such a public network 18 includes a plurality of other communication terminals not illustrated in figure 1.

The private network 16 is for example a local network connecting the terminals 12A to 12E to connect each of them to the public network 18.

The private network 16 advantageously includes means for protecting the terminals 12A to 12E against computer attacks coming from a terminal of the public network 18, for example the terminal 11.

However, the present invention is applicable to any other technically possible structure of the computer network, and figure 1 shows only one example of such a structure.

Thus, according to one example embodiment (not illustrated), the private network 16 is connected to the public network 18 via a central terminal monitoring all of the data streams leaving or entering the private network 16. This central terminal advantageously includes protection means making it possible to detect and stop at least some of the

computer attacks coming from a terminal of the public network 18 toward one of the terminals 12A to 12D of the private network.

According to still another example embodiment (not illustrated), all of the terminals 11, 12A to 12E are part of a same public or private network.

Each terminal 11, 12A to 12E is associated with an identifier ID_1 making it possible to distinguish it uniquely and certainly within the network 10. Such an identifier ID_1 for example corresponds to the IP (Internet Protocol) address of the corresponding terminal 11, 12A to 12E assigned during its connection to the network 10, or to the MAC (Media Access Control) physical address determined from the electronic components forming that terminal.

The terminals 11, 12A to 12E are for example computers all having a substantially similar set of components. Thus, these similar components will be explained below using one example embodiment of the terminal 12A illustrated in figure 2.

As illustrated in this figure 2, the terminal 12A includes a computing unit 22 such as a processor, a storage unit 24 and a connecting unit 26 for connecting to the network 10.

The connecting unit 26 in particular makes it possible to share computer data between the terminal 12A and other terminals 11, 12B to 12E via the network 10. The connecting unit 26 for example comprises a network card known in itself.

The storage unit 24 is able to store computer data, and more particularly, software that can be executed by the computing unit 22.

In particular, the storage unit 24 is able to store network software 31 to 33 capable of connecting with other network software of the terminals 11, 12B to 12E.

The storage unit 24 is further able to store a driver 36 for connections with the other terminals 11, 12B to 12E of the network 10.

The driver software 36 makes it possible to establish a connection between one of the network programs 31 to 33 with at least one other network program of at least one terminal 11, 12B to 12E to exchange computer data between these network programs via the network 10.

To that end, the driver 36 is able to define a communication port P_1 to P_3 for each network program 31 to 33.

Such a communication port is associated with an identifier ID_2 making it possible to identify it uniquely and certainly within the corresponding terminal 12A. This identifier for example comprises a number expressed over 16 bits and therefore varying between 0 and 65,535.

Thus, when a connection is requested by a remote network program for example executed by the computing unit of the terminal 11, toward one of the network programs 31 to 33, for example toward the program 31, the remote program is able to initiate a connection by emitting computer data toward the network program 31 and incorporating the identifier ID_2 associated with the port P_1 corresponding to that program 31.

In this case, the terminal 11 is an initiating terminal, and the terminal 12A is a receiving terminal.

The connection is initiated according to a communication protocol known in itself, such as TCP/IP or UDP.

The computer data are transmitted for example in the form of data packets. Each packet comprises a header in particular including information on the receiving terminal such as the identifier ID_1 of that terminal or the identifier ID_2 of the port for which that packet is intended.

Upon receiving computer data, the driver 36 is able to recognize the identifier ID_2 integrated in that data, and to send and receive data toward the network program 31 via the port P_1 corresponding to that identifier ID_2 .

The driver 36 is further able to define the operating state for each port P_1 to P_3 among the open, closed or filtered states.

When the port P_1 to P_3 is in its open state, the driver 36 is able to establish a connection initiated by the remote program between this remote software and the software 31 to 33 corresponding to this port P_1 to P_3 .

When a port P_1 to P_3 is in its closed state, the driver 36 is able to reject any connection initiated by remote software toward this port P_1 to P_3 . This is for example the case when no software is associated with the port P_1 to P_3 toward which the connection has been initiated.

When a port P_1 to P_3 is in its filtered state, it is in the open state only for some terminals of the network 10, for example according to a list of identifiers ID_2 of these predetermined terminals. For connections initiated from other terminals, a port in the filtered state does not send any response. Thus, for example, the port P_2 is in the open state for all of the connections initiated from the terminals 12B to 12E of the private network 16, and does not send any response to the connections initiated from the terminals of the public network 18.

Each of the terminals 11, 12B to 12E includes a set of components similar to those of the terminal 12A and previously described, such as the computing unit 22, the storage unit 24, the connecting unit 26 or the network programs 31 to 33 and the connection driver 36.

In particular, the connection driver 36 of each of the terminals 11, 12B to 12E is able to define a plurality of communication ports P_1 to P_3 and to associate each of these ports P_1 to P_3 with one of the network programs 31 to 33 stored in the storage unit 24.

As in the previous case, each of the ports P_1 to P_3 is associated with a unique identifier ID_2 within the corresponding terminal 11, 12B to 12E. At least some of the identifiers ID_2 associated with the network software 31 to 33 having similar functions in the different terminals 11, 12A to 12E are identical.

Unlike the terminal 12A, each of the terminals 12B to 12E further includes a device 40B to 40E for analyzing initiated connections, and a shared memory 42B allocated in the storage unit 24. This is shown in figure 3, illustrating an example embodiment of the terminal 12B.

The set of devices for analyzing initiated connections 40B to 40E implements a method for detecting unauthorized port scanning according to the invention.

Each device 40B to 40E for example assumes the form of a program stored in the corresponding storage unit 24 or in the form of an electronic component for example integrated in the corresponding connecting unit 26.

Each device 40B to 40E is able to detect each connection initiated by an initiating terminal 11, 12A to 12E, toward one of the ports P_1 to P_3 of the terminal 12B to 12E corresponding to that device 40B to 40E.

Each device 40B to 40E is further able to identify, among the set of initiated connections, each connection rejected by at least one port P_1 to P_3 of the corresponding terminal 12B to 12E.

The connection is rejected when it is initiated toward a port P_1 to P_3 being in the closed state or a port P_1 to P_3 being in the filtered state whose filter does not accept the connections from the corresponding initiating terminal.

For each rejected connection, each device 40B to 40E is able to generate security information I. This information I in particular comprises a first datum D_1 corresponding to the identifier ID_1 of the terminal 12B to 12E corresponding to this device 40B to 40E, and a second datum D_2 corresponding to the identifier ID_2 of the port P_1 to P_3 toward which the corresponding connection has been initiated.

Lastly, each device 40B to 40E is able to add the generated security information I in a security information database BI and analyze that database BI as will be explained below.

The shared memory 42B to 42E of each of the terminals 12B to 12E is able to store computer data accessible to the set of terminals 12B to 12E via the network 10.

Thus, each of these terminals 12B to 12E is able to emit computer data toward the terminal 12B to 12E corresponding to those memory 42B to 42E to store these computer data therein.

Furthermore, each of the terminals 12B to 12E is able to access this shared memory 42B to 42E to read and/or process previously stored computer data.

The communication between the terminals 12B to 12E to access the shared memory 42B to 42E of one of them for example uses peer-to-peer (P2P) technology known in itself. In this case, the set of shared memories 42B to 42E forms a shared memory for the terminals 12B to 12E, also known under the term "distributed cache".

The shared memory 42B to 42E of each of the terminals 12B to 12E in particular makes it possible to store at least part of the security information database BI made up of the security information I added by the analysis devices 40B to 40E.

The security information database BI is diagrammatically illustrated in figure 4 and in particular comprises a correspondence graph G. This graph G includes nodes of the first type N_1 and nodes of the second type N_2 , and edges connecting nodes of different types.

The nodes of the first type N_1 correspond to the first data D_1 of the set of security information I added by the devices 40B to 40E, and the nodes of the second type N_2 correspond to the second data D_2 of this information.

Each edge between a node of the first type N_1 and a node of the second type N_2 corresponds to the added security information I containing the first D_1 and second D_2 data respectively associated with these nodes of the first N_1 and second N_2 types.

Thus, in figure 4, the nodes $ID_1(12B)$, $ID_1(12C)$, $ID_1(12D)$ and $ID_1(12E)$ of the first type N_1 correspond to the identifiers ID_1 of the terminals 12B to 12E, the nodes $ID_2(P_1)$, $ID_2(P_2)$ and $ID_2(P_3)$ of the second type N_2 correspond to the identifiers ID_2 of the ports P_1 to P_3 , and the edges 51 to 54 correspond to the security information I respectively associated with a rejected connection toward the port P_2 of the terminal 12B, a rejected connection toward the port P_1 of the terminal 12A, a rejected connection toward the port P_3 of the terminal 12D and a rejected connection toward the same port of the terminal 12E.

Each set of nodes connected via one or more edges of the correspondence graph G forms a connected component of that graph G. An isolated node also forms a connected component.

Thus, in figure 4, the nodes $ID_1(12B)$, $ID_1(12C)$, $ID_2(P_1)$ and $ID_2(P_2)$ form the connected component 61, and the nodes $ID_1(12D)$, $ID_1(12E)$ and $ID_2(P_3)$ form the connected component 62.

Additionally, the security information database BI further comprises metadata 58 related to the rejected connections. These metadata 58 for example comprise a number of occurrences of a same item of security information I in the information database BI, the addition date of each occurrence of this security information I and/or the identifier ID₁ of the initiating terminal of the rejected connection corresponding to each occurrence of the security information I.

Also additionally, the metadata 58 further comprises a header of the packets corresponding to the rejected connections.

Figure 4 shows a flowchart of the main steps of the method for detecting unauthorized port scanning 100.

Initially, each terminal of the set of terminals 11, 12A to 12E is connected to the network 10 and able to initiate a connection with one of the other terminals 11, 12A to 12E.

During step 110, each of the analysis devices 40B to 40E detects each connection initiated by an initiating terminal 11, 12A to 12E, toward one of the ports P₁ to P₃ of the terminal 11, 12A to 12E corresponding to this analysis device 40B to 40E.

Thus, when for example the connection is initiated under the TCP/IP protocol by the initiating terminal 11, 12A to 12E, for example by the terminal 11, the latter emits, toward a port P₁ to P₃, for example toward the port P₃, of a receiving terminal 12B to 12E, for example the terminal 12C, computer data including a packet of the "SYN" type.

This packet is received by the driver 36 of the terminal 12B, which then notifies the analysis device 40B of an initiated connection.

During the following step 120, each of the analysis devices 40B to 40E identifies, among the set of connections initiated toward the corresponding terminal 12B to 12E, each connection rejected by at least one of the ports P₁ to P₃ of this terminal.

Thus, in the example mentioned above, when the port P₃ of the receiving terminal 12C is in the closed state, the driver 38 sends the initiating terminal 11 computer data including a packet of the RST (reset) type indicating the rejection of the initiated connection and notifies the analysis device 40C.

Alternatively, the driver 38 does not answer the initiating terminal 11, which also indicates rejection of the initiated connection.

During the following step 130, for each rejected connection, the corresponding analysis device 40B to 40E generates an item of security information I. This information I in particular comprises a first datum D₁ corresponding to the identifier ID₁ of the receiving terminal 12B to 12E and a second datum D₂ corresponding to the identifier ID₂ of the port P₁ to P₃ toward which the corresponding connection has been initiated.

Additionally, the security information further comprises a third datum D_3 corresponding to metadata related to the rejected connection. Thus for example, these metadata comprise the identifier ID_1 of the initiating terminal or the header of the data packet corresponding to this rejected connection.

5 In the example mentioned above, the analysis device 40C therefore generates an item of security information I comprising a first datum D_1 corresponding to the identifier ID_1 of the terminal 12C and a second datum D_2 corresponding to the identifier ID_2 of the port P_3 .

10 During the following step 140, the corresponding analysis device 40B to 40E adds the generated security information I into the security information database BI .

To that end, step 140 comprises several sub-steps described below.

During sub-step 141, the corresponding analysis device 40B to 40E determines whether the correspondence graph G already contains a node of the first type N_1 corresponding to the first datum D_1 of the generated security information I . If not, the corresponding analysis device 40B to 40E adds a new node of the first type N_1 corresponding to the first datum D_1 in the graph G during step 142.

20 During the following sub-step 143, the corresponding analysis device 40B to 40E determines whether the correspondence graph already contains a node of the second type N_2 corresponding to the second datum D_2 of the generated security information item I . If not, the corresponding analysis device 40B to 40E adds a new node of the second type N_2 corresponding to the second datum D_2 in the graph G during step 144.

During the following sub-step 145, the analysis device 40B to 40E identifies a first connector component C_1 of the correspondence graph G containing a node of the first type N_1 formed by the first datum D_1 .

25 During the following sub-step 146, the analysis device 40B to 40E identifies a second connector component C_2 of the correspondence graph G containing a node of the second type N_2 formed by the second datum D_2 .

During the following sub-step 147, the analysis device 40B to 40E compares the first C_1 and second C_2 identified connected components.

30 If the components C_1 and C_2 are different, the analysis device 40B to 40E merges these connected components by creating an edge between the nodes formed by the first datum D_1 and the second datum D_2 during step 148.

Otherwise, during step 149, the analysis device 40B to 40E creates an edge between the nodes formed by the first datum D_1 and the second datum D_2 if it does not already exist.

Additionally, the analysis device 40B to 40E increments the number of occurrences of the security information item I in the metadata 58 by one and additionally adds the third datum D_3 corresponding to the identifier ID_1 of the initiating terminal 11, 12A to 12E and the addition date of that occurrence.

Thus, in the aforementioned example combined with the example of the current state of the security information database BI illustrated in figure 4, the analysis device 40C identifies the connected components 61 containing the node $ID_1(12C)$ corresponding to the first datum D_1 and the connected components 62 containing the node $ID_2(P_3)$ corresponding to the second datum D_2 . Since the components 61 and 62 are different, the analysis device 40C merges the components 61 and 62 by creating a new edge A.

Additionally, the analysis device 40C adds, into the metadata 58, the third datum D_3 corresponding to the identifier ID_1 of the terminal 11.

During the following step 150, the corresponding analysis device 40B to 40E analyzes the security information database BI modified during step 148 or step 149 to determine unauthorized port scans in the network 10.

This analysis in particular comprises comparing the total number of nodes in the connected component obtained after merging the components C_1 and C_2 during step 148 if this has been done, with a predetermined threshold.

Thus, when the total number of nodes in this connected component exceeds the predetermined threshold, the analysis device 40B to 40E raises an alarm indicating a potential unauthorized port scan in the network 10.

Alternatively, this analysis comprises comparing the total number of edges in the connected component obtained after merging the components C_1 and C_2 during step 148 or modified during step 149, with a predetermined threshold.

Alternatively or additionally, this analysis comprises computing an alarm value and comparing this value with a predetermined threshold. The alarm value depends on the total number of edges and/or nodes in the connected component obtained during step 148 or modified during step 149. Additionally, the alarm value also depends on at least some of the data contained in the metadata 58, for example the identifiers ID_1 of the initiating terminals.

The method 100 is implemented by each of the analysis devices 40B to 40E in real-time or at predetermined time intervals.

According to one advantageous aspect of the invention, the method 100 further comprises a step 160 for updating the security information database BI in particular making it possible to erase the oldest security information I contained in that database.

This step 160 is implemented by one of the analysis devices 40B to 40E and comprises several sub-steps described below. The flowchart of step 160 is shown in figure 6.

During sub-step 161, the corresponding analysis device 40B to 40E identifies the occurrence of the oldest security information item I contained in the information database I. To that end, the analysis device 40B to 40E analyzes the addition dates of the occurrences of each security information item I from metadata 58.

During the following sub-step 162, if the number of occurrences of the security information I corresponding to the oldest occurrence identified during step 161 is greater than 1, the corresponding analysis device 40B to 40E decrements this occurrence number by one and erases all of the metadata related to the oldest occurrence during step 163.

Otherwise, the identified oldest occurrence corresponds only to the occurrence of the corresponding security information item I, and the analysis device 40B to 40E therefore erases the security information I from the information database BI.

To that end, during sub-step 164, the analysis device 40B to 40E identifies the connected component containing a node of the first type N_1 and a node of the second type N_2 corresponding respectively to the first D_1 and second D_2 data of this security information item I.

In the following sub-step 165, the corresponding analysis device 40B to 40E eliminates the edge corresponding to this security information I.

During the following sub-step 166, the corresponding analysis device 40B to 40E analyzes the connection of the nodes in the connected component after eliminating the edge during the previous step.

If each node of this component is still connected to each node by one or more edges, the corresponding analysis device 40B to 40E keeps the connector component unmodified during step 167.

Otherwise, during step 168, the corresponding analysis device 40B to 40E creates two new connected components, one of which contains the node corresponding to the first datum D_1 of the security information item I and the other of which contains the node corresponding to the second datum D_2 of the security information item I. When one or the other connected component newly created is made up of a single node, the analysis device 40B to 40E eliminates that node from the correspondence graph G.

One can then see that the present invention includes a certain number of advantages.

Thus, according to the invention, the hardware resources used to detect unauthorized port scans are distributed between several terminals of the network to be protected.

5 This in particular makes it possible to store a large security information database in the shared memories, and thus to detect scans spread out over time.

Furthermore, the computing resources distributed between these terminals make it possible to analyze this information database particularly quickly and easily.

10 Furthermore, the method according to the invention makes it possible to detect unauthorized port scans coming from one terminal of the network toward other terminals of the same network. Thus, in the example of figure 1, the method implemented by the analysis devices 40B to 40E of the terminals 12B to 12E makes it possible to detect scans coming from the terminal 12A that is part of the same private network 16.

15 Lastly, the method according to the invention makes it possible to complete the protection means already present in a computer network and to increase their effectiveness in detecting computer attacks.

15
CLAIMS

1.- A method (100) for detecting unauthorized port scanning in a computer network (10) in particular to avoid a computer attack, the computer network (10) being made up of a plurality of communication terminals (11, 12A,...,12E) each defining a plurality of communication ports ($P_1, \dots, P_3$) and able to communicate with one another via these communication ports ($P_1, \dots, P_3$), each terminal (11, 12A,...,12E) being associated with a unique terminal identifier (ID_1) in the network (10), each port ($P_1, \dots, P_3$) being associated with a unique port identifier (ID_2) within the corresponding terminal (11, 12A,...,12E),

at least some of the terminals (11, 12A,...,12E), called initiating terminals, being able to initiate a computer connection with at least one other terminal (11, 12A,...,12E), called receiving terminal, via at least one port ($P_1, \dots, P_3$) of the receiving terminal (11, 12A,...,12E), each receiving terminal (11, 12A,...,12E) being able to reject a connection initiated by an initiating terminal (11, 12A,...,12E),

the method (100) being implemented by terminals (12B,...,12E) forming an identified group of terminals and including the following steps:

- detection (110), by each receiving terminal (12B,...,12E) in the group, of each connection initiated by an initiating terminal (11, 12A,...,12E) via at least one port ($P_1, \dots, P_3$) of that receiving terminal (12B,...,12E);

- identification (120), by each receiving terminal (12B,...,12E) of the group, of rejected connections among all of the initiated connections;

- for each rejected connection, generating (130) security information (I) comprising a first datum (D_1) and a second datum (D_2) respectively corresponding to the identifiers (ID_1, ID_2) of the receiving terminal (12B,...,12E) and one of its ports ($P_1, \dots, P_3$) corresponding to this rejected connection, and adding (140) the generated security information (I) in a security database (BI) shared between the terminals (12B,...,12E) in the group; and

- analyzing (150) the shared security information database (BI) to determine unauthorized port scans.

2.- The method (100) according to claim 1, wherein each rejected connection corresponds to an initiated connection toward a closed port ($P_1, \dots, P_3$) or toward a port ($P_1, \dots, P_3$) comprising a filter refusing the connection with the corresponding initiating terminal (11, 12A,...,12E).

3.- The method (100) according to claim 1 or 2, wherein each terminal (12A,...,12E) of the group further includes a storage unit (24) able to store at least part of the security information database (BI).

5 4.- The method (100) according to claim 3, wherein the security information database (BI) is shared in the group using so-called "peer-to-peer" (P2P) type technology.

10 5.- The method (100) according to any one of the preceding claims, wherein the security information database (BI) comprises a correspondence graph (G) whose nodes are provided by the first data (D_1) and the second data (D_2) from each added item of security information (I).

15 6.- The method (100) according to claim 5, wherein each added item of security information (I) corresponds to an edge (51,...,54) between two nodes formed by the first datum (D_1) and the second datum (D_2) coming from the security information (I), a set of nodes connected via one or more edges (C_1 , C_2 , 61, 62) forming a connected component of the correspondence graph (G).

20 7.- The method (100) according to claim 6, wherein the addition of an item of security information (I) comprises the following sub-steps:

- creating (142) a new node formed by the first datum (D_1) of this security information item (I) if such a node does not exist in the correspondence graph (G);
- creating (144) a new node formed by the second datum (D_2) of this security information item (I) if such a node does not exist in the correspondence graph (G);
- 25 - identifying (145) a first connector component (C_1) of the correspondence graph (G) containing a node formed by the first datum (D_1);
- identifying (146) a second connector component (C_2) of the correspondence graph (G) containing a node formed by the second datum (D_2);
- 30 - if the first component (C_1) and the second connected component (C_2) are different, merging (148) the first connected component (C_1) with the second connected component (C_2) by creating an edge (A) between the nodes formed by the first datum (D_1) and the second datum (D_2).

35 8.- The method (100) according to claim 6 or 7, wherein the analysis of the security information database (BI) comprises determining the number of nodes of at least one connector component of the correspondence graph (G).

9.- The method (100) according to claim 7 and 8, wherein the at least one connector component for which the number of nodes is determined corresponds to a component obtained after merging the first connected component (C_1) and the second connected component (C_2).

10.- The method (100) according to claim 8 or 9, wherein port scanning is determined when the number of identified nodes exceeds a predetermined threshold.

11.- The method (100) according to any one of the preceding claims, wherein each security information item (I) further comprises the identifier (ID_1) of the initiating terminal (11, 12A,...,12E) having initiated the rejected connection corresponding to the security information (I).

12.- The method (100) according to any one of the preceding claims, wherein the security information database (BI) further comprises the number of occurrences of each security information item (I) in that database (BI).

13.- The method (100) according to any one of the preceding claims, further comprising a step (160) for updating the security information database (BI) in particular making it possible to erase the oldest security information (I) contained in that database.

14.- A computer program product comprising software instructions which, when implemented by a computer, carry out the method according to any one of the preceding claims.

15.- An analysis device (40B,...,40E) for detecting unauthorized port scanning in a computer network (10) in particular to avoid a computer attack, the computer network (10) being made up of a plurality of communication terminals (11, 12A,...,12E) each defining a plurality of communication ports ($P_1,...,P_3$) and able to communicate with one another via these communication ports ($P_1,...,P_3$), each terminal (11, 12A,...,12E) being associated with a unique terminal identifier (ID_1) in the network (10), each port ($P_1,...,P_3$) being associated with a unique port identifier (ID_2) within the corresponding terminal (11, 12A,...,12E),

at least some of the terminals (11, 12A,...,12E), called initiating terminals, being able to initiate a computer connection with at least one other terminal (11, 12A,...,12E), called receiving terminal, via at least one port ($P_1,...,P_3$) of the receiving terminal (11,

12A,...,12E), each receiving terminal (11, 12A,...,12E) being able to reject a connection initiated by an initiating terminal (11, 12A,...,12E),

the analysis device (40B,...,40E) being able to:

- detect each connection initiated by an initiating terminal (11, 12A,...,12E) via at least one port ($P_1, \dots, P_3$) of that receiving terminal (12B,...,12E);

- identifying connections among all of the initiated connections;

- for each rejected connection, generating security information (I) comprising a first datum (D_1) and a second datum (D_2) respectively corresponding to the identifiers (ID_1, ID_2) of the receiving terminal (12B,...,12E) and one of its ports ($P_1, \dots, P_3$) corresponding to this rejected connection, and adding the generated security information (I) in a shared security database (BI); and

- analyzing the shared security information database (BI) to determine unauthorized port scans.