Office de la Propriété Intellectuelle du Canada
Un organisme d'Industrie Canada

Canadian Intellectual Property Office
An agency of Industry Canada

CA 2512697 C 2011/11/29

(11)(21) 2 512 697

(12) BREVET CANADIEN
CANADIAN PATENT

(13) C

(22) Date de dépôt/Filing Date: 1999/10/22

(41) Mise à la disp. pub./Open to Public Insp.: 2000/04/22

(45) Date de délivrance/Issue Date: 2011/11/29

(62) Demande originale/Original Application: 2 287 823

(30) Priorité/Priority: 1998/10/22 (US60/105,188)

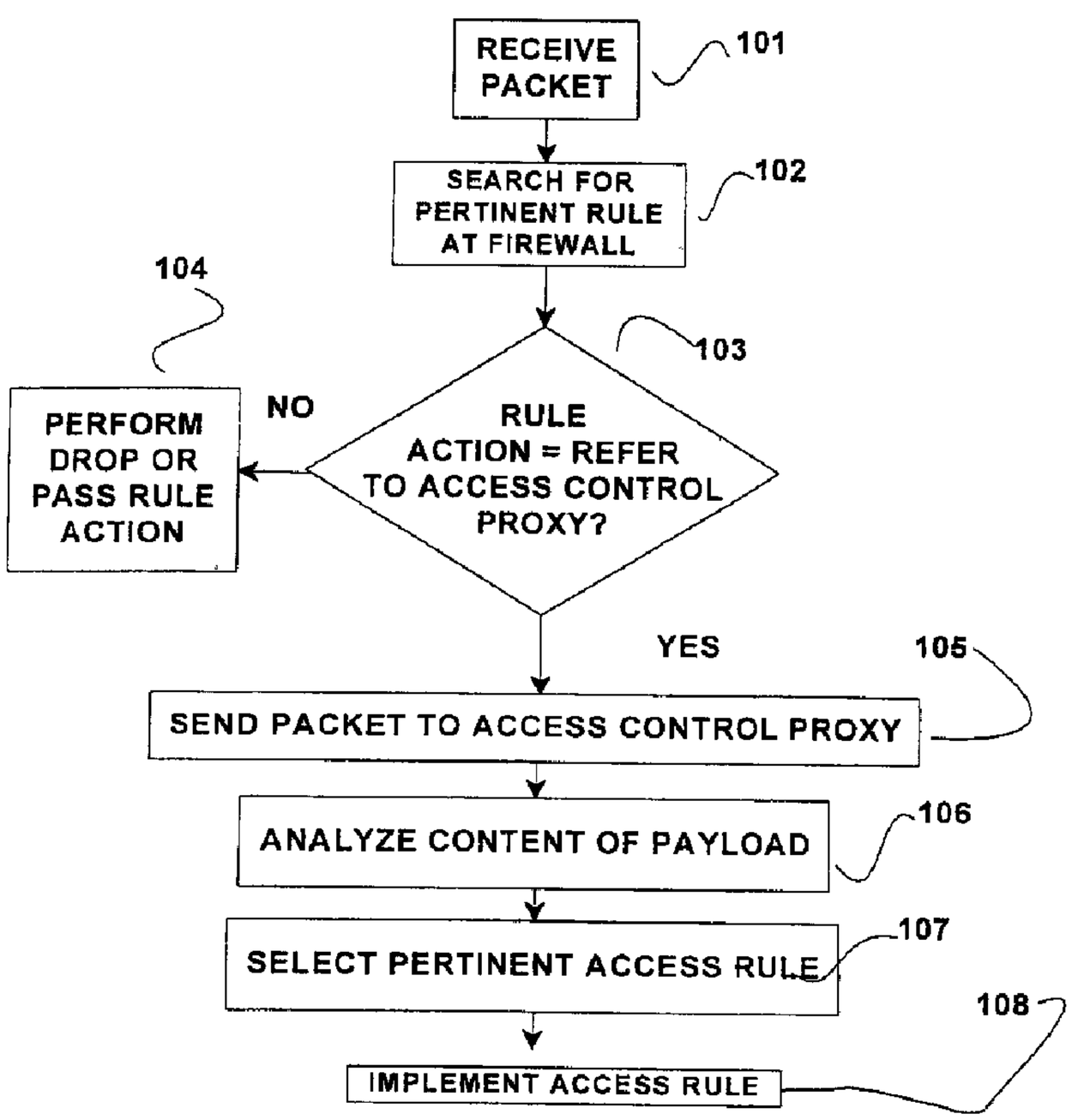
(51) Cl.Int./Int.Cl. *H04L 12/56* (2006.01),
H04L 29/06 (2006.01)

(72) Inventeurs/Inventors:
DUTTA, PARTHA P., US;
KUMAR, MAHESH M., US;
LERNER, MICHAH, US

(73) Propriétaire/Owner:
AT&T INTELLECTUAL PROPERTY II, L.P., US

(74) Agent: KIRBY EADES GALE BAKER

(54) Titre : COMMANDE D'ACCES DE HAUTE PRECISION
(54) Title: HIGH RESOLUTION ACCESS CONTROL



(57) Abrégé/Abstract:
A system and method for high resolution access control for packetized information. A packet is received at a firewall. A rule corresponding to header information in the packet prescribes referring the packet to an access control proxy. The access control proxy analyzes the contents of the packet, and identifies a rule based upon the contents. The rule is implemented at the firewall.

Abstract

A system and method for high resolution access control for packetized information. A packet is received at a firewall. A rule corresponding to header information in the packet prescribes referring the packet to an access control proxy. The access control proxy analyzes the contents of the packet, and identifies a rule based upon the contents. The rule is implemented at the firewall.

HIGH RESOLUTION ACCESS CONTROL

This is a division of co-pending Canadian Patent Application Serial No. 2,287,823, filed October 22, 1999, published on April 22, 2000.

Field of the Invention

The field of the invention is information systems access control, and in particular high resolution filtering of packetized information.

Background of the Invention

A firewall regulates the flow of packetized information. A packet includes a header and a payload. The header includes header parameters, including a source and destination address for the packet, as well as source and destination port numbers and a protocol number. Other examples of header parameters include various flags (e.g., security features implemented with respect to the packet (AUTHENTICATED, ENCRYPTED), quality of service requirements (e.g., HIGH, MEDIUM, LOW) for handling the packet, a priority parameter for handling the packet (e.g., ROUTINE, URGENT, FLASH), etc.). The payload includes the data meant to be conveyed by the packet from its source to its intended destination.

A known firewall is placed between the packet's source and intended destination, where it intercepts the packet. The known firewall filters a packet based upon the packet's header parameters and a rule loaded into the firewall. The rule correlates a pattern in the header of a packet with a prescribed action, either PASS or DROP. The filter identifies the rule that applies to the packet based upon the packet's header, and then implements the rule's prescribed action. When a DROP action is performed, the packet is blocked (deleted), and does not reach its intended destination. When a PASS action is performed, the packet is passed on toward its intended destination. The set of rules loaded into a firewall reflect a security policy, which prescribes what type of information is permissible to pass through the firewall, e.g., from which source, to which destination, for which applications, etc.

The set of rules loaded into a known firewall operate at a low level of resolution. As described above, a firewall rule prescribes a PASS or DROP action based only upon the header parameters of the packet. Packet header parameters alone do not reveal the ultimate target of, for example, a connection request from a sender to a destination host. For example, a HyperText Transfer Protocol (HTTP) connection request to send the file located at http://www.att.com/secret.html is not entirely disclosed in the header of the packet initiating the request. The header reveals the Internet Protocol (IP) address of the proxy corresponding to the domain name att.com. However, information regarding the particular file that is being requested, secret.html, is embedded in the payload of the packet. Since known firewalls only filter packets based upon their header parameters, known filters cannot PASS or DROP a packet on the basis of a particular file at a given destination. The same shortfall in known filters exists for filtering a packet destined for a particular newsgroup, chat session, e-mail address, etc.

Summary of the Invention

The present invention provides high resolution access control for packetized information. In accordance with one embodiment of the present invention, a packet is received at a firewall and referred to an access control proxy. The access control proxy analyzes the contents of the packet, and identifies an access rule based upon the contents. The action prescribed by the access rule is performed with respect to the packet and any related packets. This advantageously provides for filtering a packet based not only upon its header information, as in known firewalls, but upon the information contained in the packet payload.

Certain exemplary embodiments can provide a method for filtering a plurality of packets, comprising: receiving said plurality of packets at a firewall, each of said plurality of packets having a header and a payload; referring at least one packet of said plurality of packets to an access control proxy; selecting at least one access rule based upon said payload of said at least one packet; and implementing said at least one access rule for said at least one packet.

Certain exemplary embodiments can provide a method for filtering at least one packet, comprising: receiving said at least one packet at a receiving node, each of said at least one packet having a header and a payload; sending a request for a pertinent access rule to a library node if said receiving node does not store an access rule corresponding to data contained in said payload of said at least one packet; receiving a pertinent access rule from said library node; and implementing said pertinent access rule for said at least one packet.

Certain exemplary embodiments can provide a system for filtering at least one packet, comprising: a first node for receiving said at least one packet, sending a request for a pertinent access rule if said receiving node does not store an access rule that pertains to data contained in a payload of one or more of said at least one packet; and a library node for receiving said query from said first node, identifying said pertinent access rule from an access rule collection, and providing said pertinent access rule for said at least one packet to said first node.

Brief Description of the Drawings

The present invention, taken in conjunction with the invention described in co-pending Canadian Patent Application Serial No. 2,287,823 filed October 22, 1999, will be described in detail hereinbelow with the aid of the accompanying drawings, in which:

FIG. 1 is a flow chart showing the method in accordance with an embodiment of the present invention.

FIG. 2 shows an apparatus in accordance with an embodiment of the present invention.

FIG. 3 shows a system in accordance with the present invention.

Detailed Description

A flow chart showing the method in accordance with an embodiment of the present invention is shown in FIG. 1. A packet is received at a firewall, step 101. The packet has at least one header parameter and a payload. As discussed above, a packet is a discrete unit of information. In one embodiment of the present invention, a packet includes a header and a payload. The header includes header parameters, such as source address, source port,

destination address, destination port and protocol number. The payload of the packet includes data being conveyed by the packet, e.g., a connection request, document data, etc.

After the packet is received, an access rule is identified that corresponds to at least one header parameter of the packet. In one embodiment, this access rule is stored locally at the firewall. In another embodiment, this access rule is obtained from a node external to the firewall.

In accordance with an embodiment of the present invention, the action prescribed by the rule that corresponds to the received packet's header information indicates that the packet is to be referred to an access control proxy. In one embodiment, the access control proxy is specific to a single protocol, e.g., the file transfer protocol (FTP), the hypertext transfer protocol (HTTP), newsgroup protocol, etc.

The access control proxy selects an access rule based upon the contents of the packet. In one embodiment, the access rule is stored locally at the firewall. In another embodiment, the access rule is retrieved from a node external to the firewall. In one embodiment, the access rule is selected based upon the name of the requested file. In another embodiment, it is selected on the basis of the URL of the requested information. For example, an access rule can be selected based upon the domain name of the requested information, or the nth degree domain name of a URL in a packet payload. The "nth degree domain name" is defined as follows: a domain name is comprised of text strings separated by periods, e.g., a.b.c.d.e. The rightmost string (e.g., "e" in the example) is the first degree domain name, the string immediately to the left on the other side of the period is the second degree domain name (e.g., "d" in the example), and each string further to the left is incremented by one degree. Thus, "c" is the third degree domain name, "b" is the fourth degree, etc.

After selecting the access rule based upon the contents of the packet, the access rule is implemented for that packet and any related packets. A related packet, for example, is another packet in the same session request as the first packet. For example, a session is likely to include many packets. The packet or packets that contain sufficient payload

information for the access proxy to select a corresponding access rule will be PASSED or DROPPED in accordance with the selected access rule, as will any other packets that comprise the connection request.

This process is shown in more detail in FIG. 1. A packet is received, step 101. The set of rules stored at the firewall is searched for a rule that pertains to the header parameters of the packet, step 102. When such a rule is identified, it is determined if the prescribed action of the rule is to refer the packet to an access control proxy, step 103. If the prescribed action is not to refer the packet, the action is to PASS or DROP the packet, which is performed for the packet, step 104. If the prescribed action is to refer the packet, the packet is then sent to the access control proxy, step 105. In one embodiment, the access control proxy analyzes the content of the packet payload to determine details not available from the header parameters as to the information which the payload requests, step 106. In another embodiment, the access control proxy analyzes the contents of a plurality of received packets to determine details pertaining to a request for information that is constituted by the plurality of payloads. The number of packets analyzed is sufficient to select an access rule pertaining to the detailed information request, i.e., to decide whether to PASS or DROP the packets pertinent to the request.

The access control proxy then selects an access rule pertaining to the detailed information request contained in the packet payload, step 107. For example, an access rule prescribes a DROP action for any packet that requests the file located at http://www.att.com/secret.html. On the other hand, an access rule prescribes a PASS action for any packet that requests the file located at http://www.att.com/public.html.

In one embodiment of the present invention, the access control proxy selects an access rule that pertains to the packet based both on an analysis of the payload and the header parameters of the packet. For example, the source address of the packet is included in the header as a header parameter. In one embodiment, the access control proxy selects an access rule that prescribes a DROP action for any packet that requests the file

http://www.att.com/secret.html and whose header indicates the packet is from SOURCE A, whereas another selected access rule prescribes a PASS action for any packet that requests the same file, but whose header indicates the packet is from SOURCE B.

In one embodiment of the present invention, the access control proxy then implements the selected access rule for the packet, performing either a PASS or a DROP action with respect to the packet, in accordance with the access rule, step 108.

An apparatus in accordance with an embodiment of the present invention is shown in FIG. 2. Peer A 201 (the sender) sends a packet of information addressed to destination Peer B 202 (the destination) through filtering device 203. The packet payload includes an identifier of a file (e.g., a filename and directory information) requested by Peer A 201 and stored at Peer B 202. Filtering device 203 comprises a processor 204, a memory 205 that stores rules 206 (e.g., both rules that refer a packet to the access control proxy and access rules that are selected by the access control proxy) and high resolution filtering instructions 207 adapted to be executed by processor 204 to perform steps of the method in accordance with an embodiment of the present invention. The filtering device 203 also includes a first port 208 through which the packet is received from Peer A 201, and a second port 209 through which the packet will pass to Peer B 202 through network 210 if the pertinent rule prescribes a PASS action with respect to the packet.

Peers 201 and 202 are each a computer with a permanent or temporary network address. Network 210 is any information systems network across which the information in the packet can be sent. Examples of network 210 include the Internet, an intranet, a virtual private network, etc.

In one embodiment, processor 204 is a general purpose microprocessor, such as the Pentium II microprocessor manufactured by the Intel Corporation of Santa Clara, California. In another embodiment, processor 204 is an Application Specific Integrated Circuit (ASIC), which has been specifically designed to perform at least some of the steps of the method in accordance with an embodiment of the present invention. ASICs are well-known in the art

for application such as digital signal processing. In an embodiment of the present invention that includes an ASIC, at least part of the high resolution filtering instructions 207 can be implemented in the design of the ASIC.

Memory 205 can be Random Access Memory (RAM), a hard disk, a floppy disk, an optical digital storage medium, or any combination thereof. Memory 205 is meant to encompass any means for storing digital information.

High resolution filtering instructions 207 are adapted to be executed by processor 204 to receive a packet, refer the packet to an access control proxy, select an access rule based upon the contents of the payload of the received packet, and then implement the access rule by performing the action (typically PASS or DROP) prescribed by the selected rule with respect to a packet. The term "high resolution filtering instructions" is meant to include access control proxy instructions. In one embodiment, the access rule is retrieved based upon a combination of the contents and header parameters of the packet. In another embodiment, the access rule is selected based upon the contents of one or several packet payloads.

In one embodiment of the present invention, high resolution filtering instructions 207 include firewall instructions and access control proxy instructions. In one embodiment, the firewall instructions are executed on processor 204 as a firewall process, and the access control proxy instructions are executed on processor 204 as an access control proxy process. When filtering device 203 receives a packet, the firewall process searches for and identifies a rule pertinent to the packet. The rule prescribes an action, either PASS, DROP or to REFER the packet to an access control proxy. In one embodiment of the present invention, there is a distinct access control proxy for each different protocol to which a packet can conform, e.g., HTTP, FTP, e-mail, newsgroup, telnet, etc. The protocol of a packet in one embodiment is indicated as a protocol number in the packet header. An embodiment of the present invention advantageously uses the protocol number in the header to refer a packet to the correct access control proxy process.

When a packet is referred to an access control proxy process, the proxy process analyzes the contents of the packet and selects an access rule based upon the results of the content analysis. In one embodiment, the selected access rule is stored locally. In another

embodiment, the selected access rule is retrieved from an external database. In yet another embodiment, the access rule is dynamically formulated by the proxy. The access rule is implemented at the firewall.

In one embodiment of the present invention, several (more than one) packets are referred to the access control proxy process. The access control proxy process analyzes the contents of the several packets, and selects an access rule based upon the results of this analysis. In one embodiment, the information needed to select an access rule is spread across the contents of the several packets, and may not be contained in any one of the several packets alone. Thus, in one embodiment, the contents of a packet may be represented as:

Packet: SELECT_RULE_1432

This shows that there is sufficient information in the single packet to identify the rule that should be selected. On the other hand, consider four packets that contain the following information:

Packet 1: SELECT_RULE_FIRST_DIGIT_1

Packet 2: SELECT_RULE_SECOND_DIGIT_4

Packet 3: SELECT_RULE_THIRD_DIGIT_3

Packet 4: SELECT_RULE_FOURTH_DIGIT_2

The above example is primarily heuristic. Another example arises when several packets need to be analyzed to determine what type of message is being carried by the packets, and where traffic is regulated through the firewall based upon the type of message being carried.

In one embodiment, there are a plurality of ports to and from numerous destinations. The port or ports that communicate packets to and from filtering device 203 are meant to encompass any number or configuration of ports. The port configuration is expected to vary to suit the particular connectivity required of a filtering device 203 in a given situation, i.e., in a given context or architecture in which parties communicate through filtering device 203.

In various embodiments, the functions of the present invention are performed on separate nodes. In one embodiment shown in FIG. 3, a packet is received from a sender 301 at one of a plurality of receiving nodes 302, which node 302 then refers the packet to a

locally executing access control proxy 303. If the local access control proxy 303 does not store a rule corresponding to the contents of the packet, it sends a query through network 304 to another separate node 305 that can advantageously function as a central library that stores a large number of access rules 306, only some of which may be needed at any one time by the plurality of receiving nodes 302. The library node 305 identifies the pertinent access rule from its collection of access rules 306, and then sends it to the access control proxy at the requesting receiving node 302, which then implements it. This illustrates the advantageous scalability of the present invention. Only relatively few library sites (in relation to the number of receiving nodes) need store large numbers of access rules.

In another embodiment, the firewall is on a receiving node 302, and performs firewall functions, including receiving a packet (using a rule), referring the packet to the access control proxy, and implementing an access rule. The access control proxy is on another node 305, and there performs proxy functions including analyzing the packet and selecting an access rule, which it then sends to the receiving node 302 to implement. In other words, the firewall functions can be performed by a different processor than the processor that performs the proxy functions.

A medium that stores instructions adapted to be executed on a processor, like memory 205, is meant to encompass any medium capable of storing digital information. Examples of a medium that stores instructions include a hard disk, a floppy disk, a Compact Disk Read Only Memory (CD-ROM), magnetic tape, flash memory, etc.

The term "instructions adapted to be executed" is meant to encompass more than machine code. The term "instructions adapted to be executed" is meant to encompass source code, assembler, and any other expression of instructions that may require preprocessing in order to be executed by a processor. For example, also included is code that has been compressed or encrypted, and must be uncompressed and/or unencrypted in order to be executed by a processor.

The present invention advantageously provides a more efficient, flexible and scalable system and method for implementing the rules of a security policy or policies at a filtering device, because a rule is only loaded at the filtering device when the rule is needed.

Claims

1. A method for filtering a plurality of packets, comprising:

receiving said plurality of packets at a firewall, each of said plurality of packets having a header and a payload;

referring at least one packet of said plurality of packets to an access control proxy;

selecting at least one access rule based upon said payload of said at least one packet; and

implementing said at least one access rule for said at least one packet.

2. The method of claim 1, wherein said referring step comprises:

identifying at said firewall a protocol number from said header of said at least one packet; and

sending said at least one packet from said firewall to said access control proxy that corresponds to said protocol number in said header of said at least one packet.

3. The method of claim 2, wherein said protocol number indicates a File Transfer Protocol and said at least one packet is sent to a File Transfer Protocol access control proxy.

4. The method of claim 2, wherein said protocol number indicates a HyperText Transfer Protocol and said at least one packet is sent to a HyperText Transfer Protocol access control proxy.

5. The method of claim 2, wherein said protocol number indicates a NewsGroup Protocol and said at least one packet is sent to a NewsGroup Protocol access control proxy.

6. The method of claim 1, wherein said at least one access rule is selected based upon a directory of a requested file in said payload of said at least one packet.

7. The method of claim 1, wherein said at least one access rule is selected based upon a file name of a requested file in said payload of said at least one packet.

8. The method of claim 1, wherein said at least one access rule is selected based upon an nth degree domain name in said payload of said at least one packet, where n is greater than two.

9. The method of claim 1, wherein said at least one access rule is selected based upon a name of a newsgroup in said payload of said at least one packet.

10. The method of claim 1, wherein said at least one access rule is selected based upon an identifier of a participant in a chat session in said payload of said at least one packet.

11. The method of claim 1, wherein said at least one access rule is selected based upon an e-mail address in said payload of said at least one packet.

12. The method of claim 1, wherein said at least one access rule is selected from a database.

13. The method of claim 1, wherein said at least one access rule is further selected based upon at least one header parameter of said at least one packet.

14. The method of claim 1, wherein said at least one access rule is selected based upon a combination of data contained in a plurality of packets.

15. A method for filtering at least one packet, comprising:

receiving said at least one packet at a receiving node, each of said at least one packet having a header and a payload;

sending a request for a pertinent access rule to a library node if said receiving node does not store an access rule corresponding to data contained in said payload of said at least one packet;

receiving a pertinent access rule from said library node; and

implementing said pertinent access rule for said at least one packet.

16. The method of claim 15, wherein further comprising:

identifying a protocol number from said header of said at least one packet at a firewall located at said receiving node; and

sending said at least one packet from said firewall to an access control proxy that corresponds to said protocol number in said header of said at least one packet.

17. The method of claim 16, wherein said protocol number indicates a File Transfer Protocol and said at least one packet is sent to a File Transfer Protocol access control proxy.

18. A system for filtering at least one packet, comprising:

a first node for receiving said at least one packet, sending a request for a pertinent access rule if said receiving node does not store an access rule that pertains to data contained in a payload of one or more of said at least one packet; and

a library node for receiving said query from said first node, identifying said pertinent access rule from an access rule collection, and providing said pertinent access rule for said at least one packet to said first node.

19. The system of claim 18, wherein said first node comprises a firewall for receiving said at least one packet, referring said at least one packet to said library node, and implementing said pertinent access rule.

20. The system of claim 18, wherein said library node comprises an access control proxy for analyzing said at least one packet and selecting said pertinent access rule.

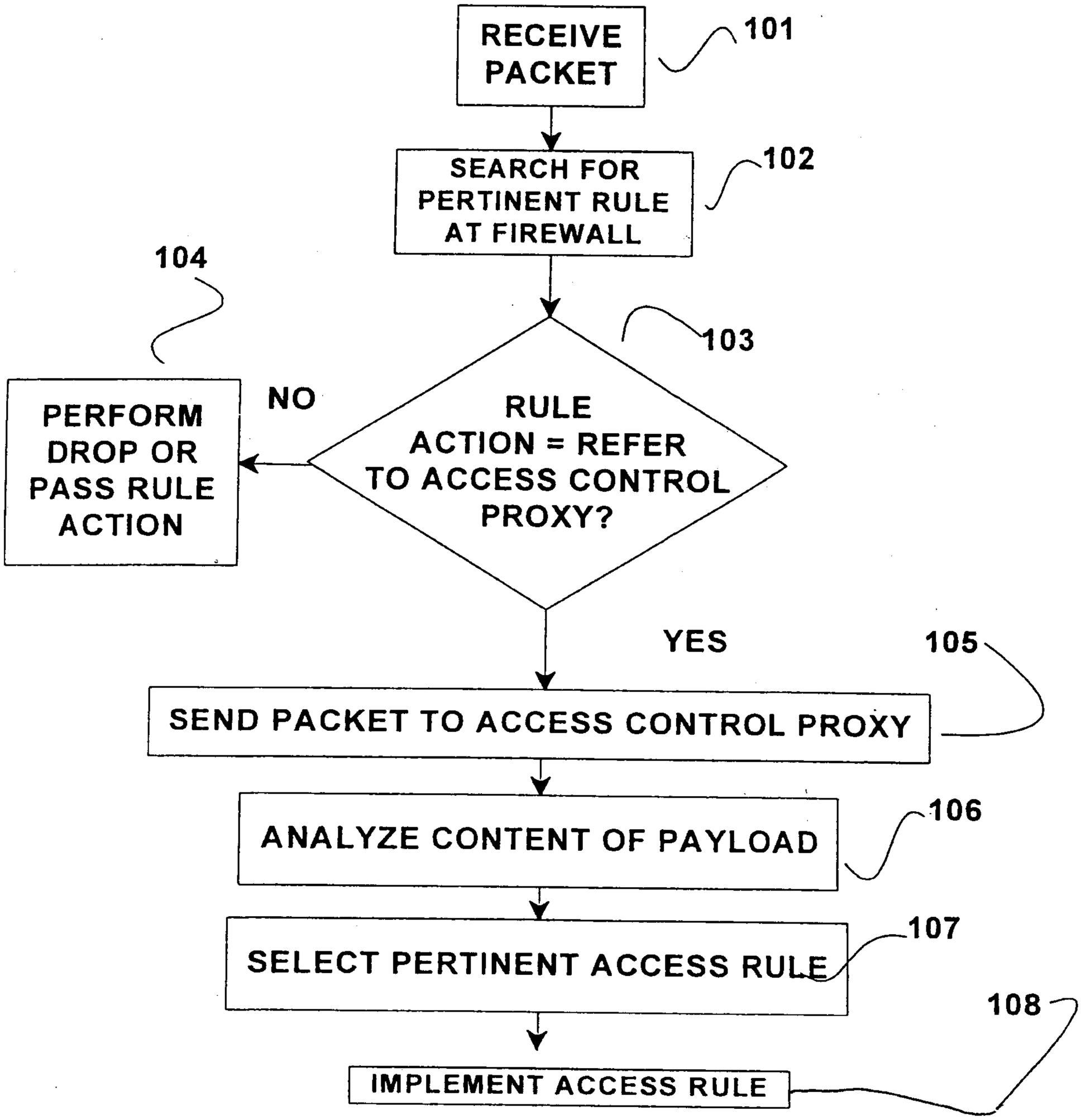

RECEIVE PACKET
101
SEARCH FOR PERTINENT RULE AT FIREWALL
102
104
103
PERFORM DROP OR PASS RULE ACTION
NO
RULE ACTION = REFER TO ACCESS CONTROL PROXY?
YES
105
SEND PACKET TO ACCESS CONTROL PROXY
ANALYZE CONTENT OF PAYLOAD
106
SELECT PERTINENT ACCESS RULE
107
108
IMPLEMENT ACCESS RULE

FIG. 1

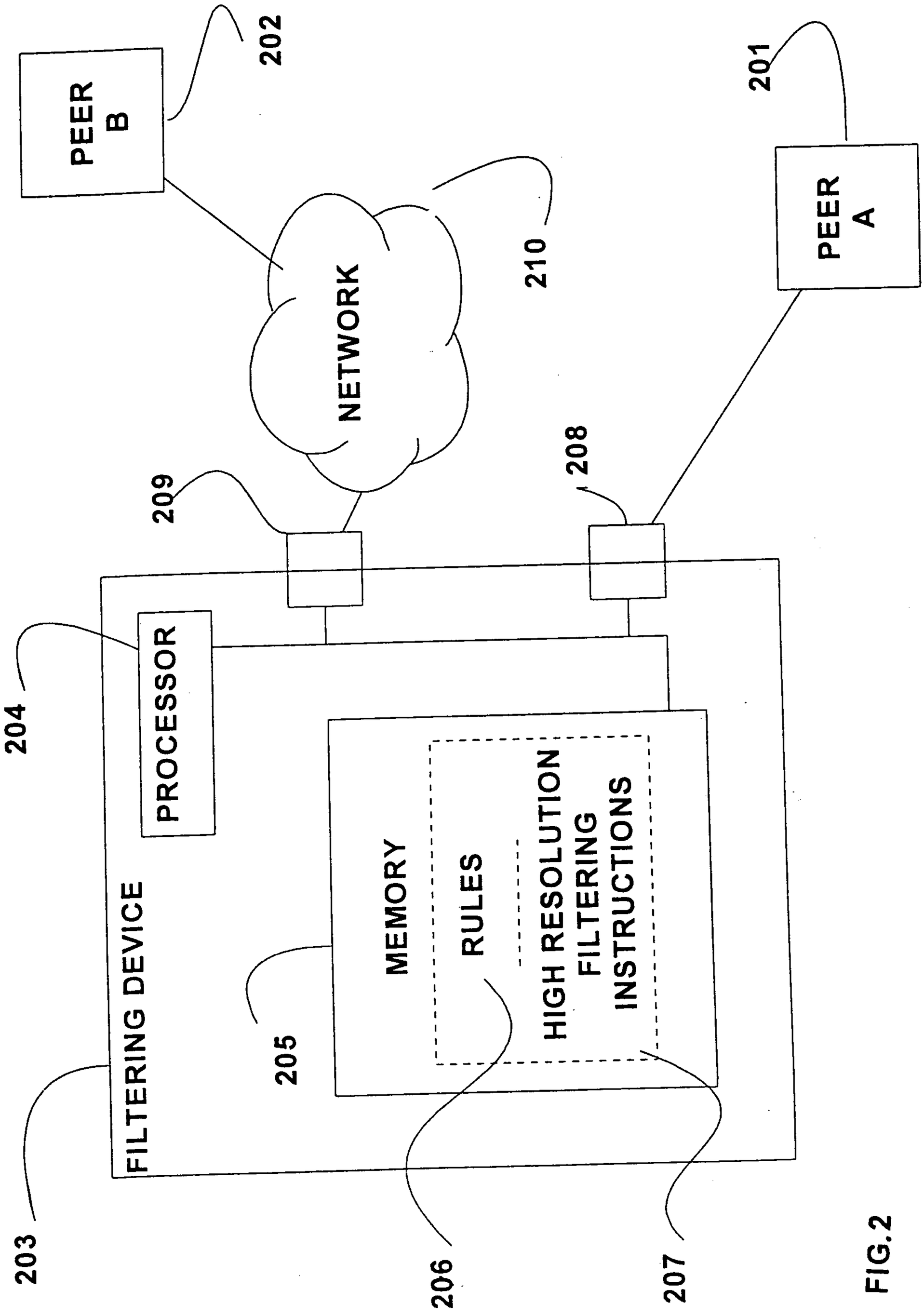

PEER B
202
NETWORK
210
201
PEER A
209
208
PROCESSOR
204
FILTERING DEVICE
203
MEMORY
205
RULES
HIGH RESOLUTION FILTERING INSTRUCTIONS
206
207

FIG.2

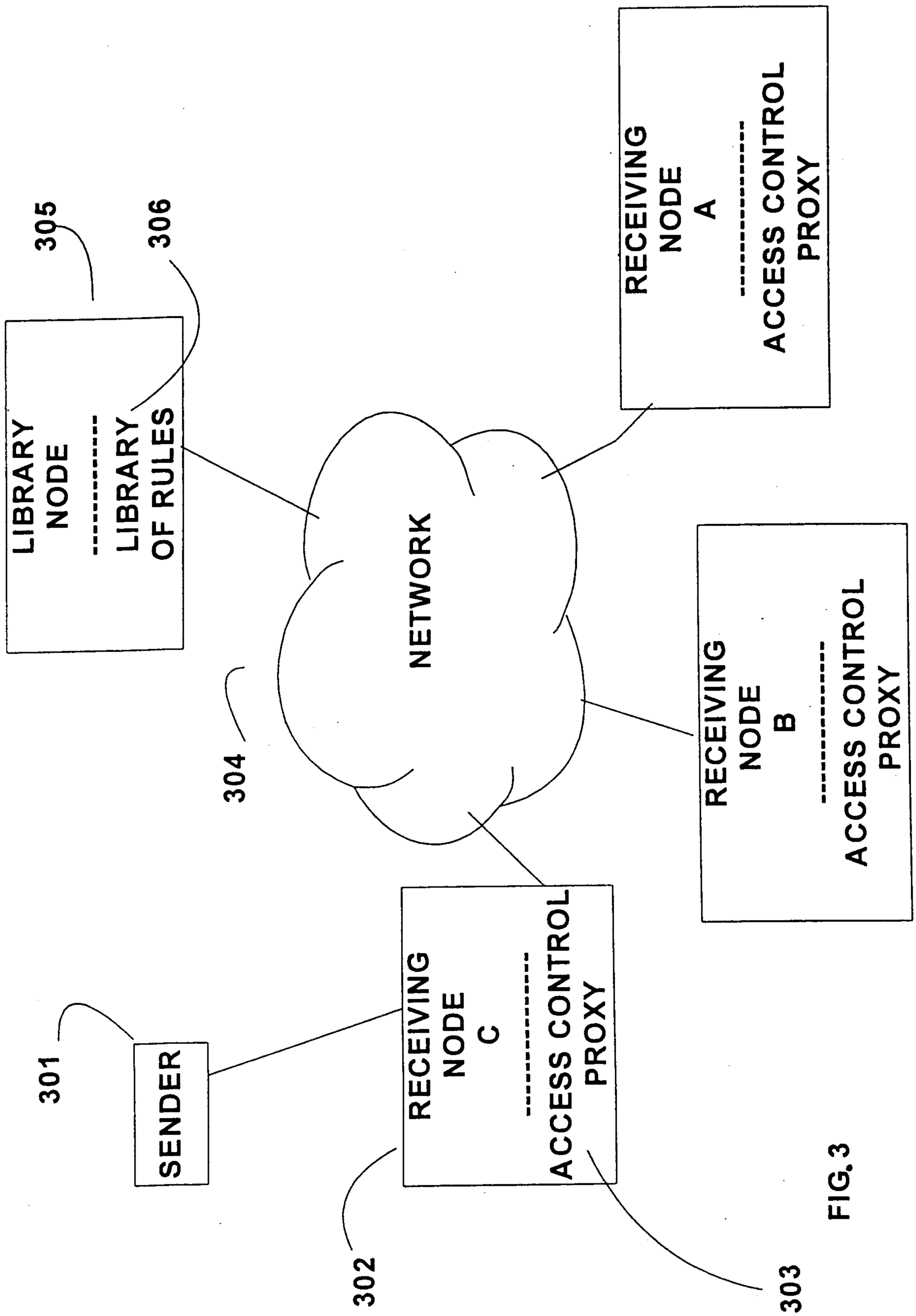
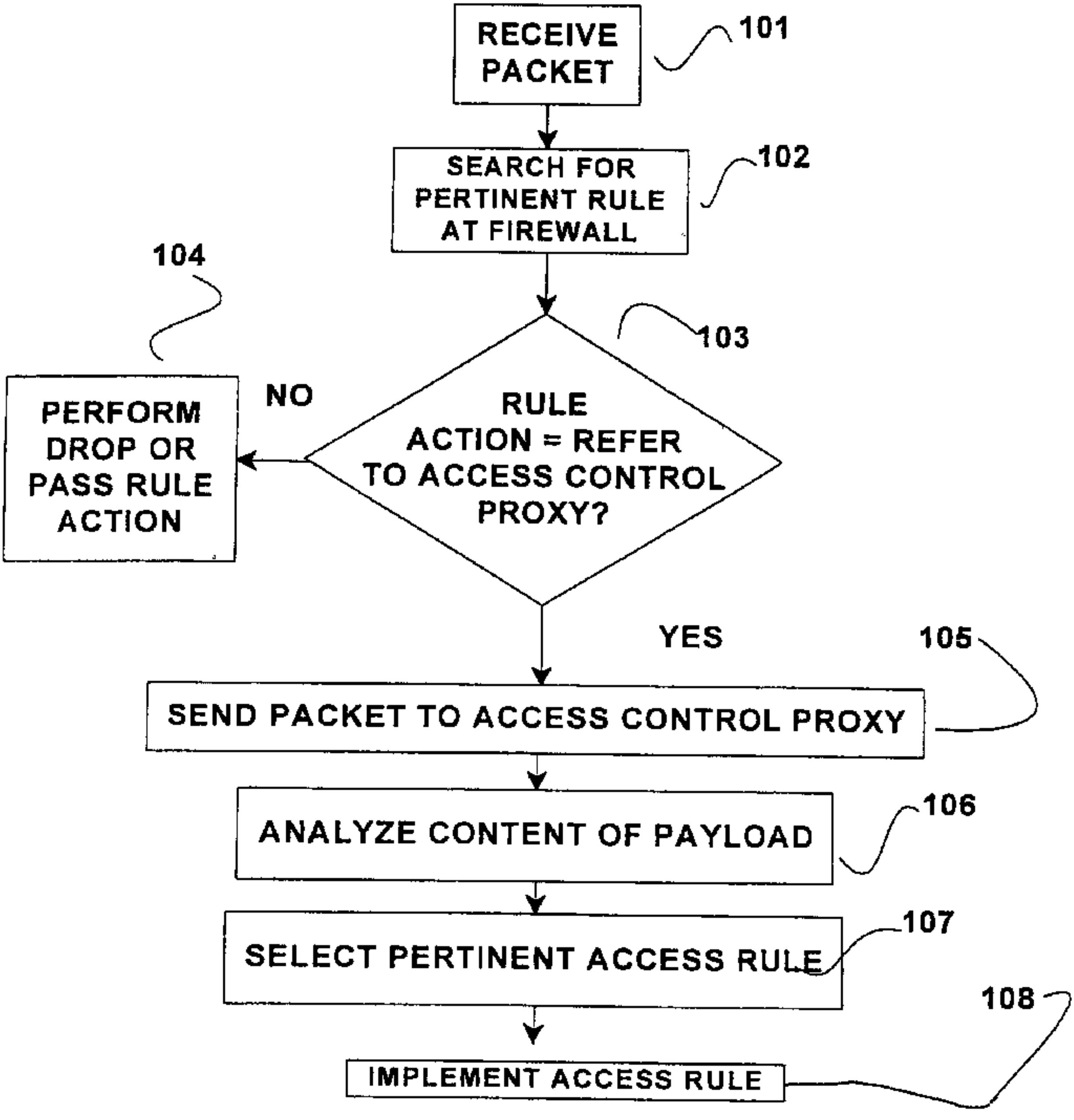

SENDER
301
RECEIVING NODE C
ACCESS CONTROL PROXY
302
303
NETWORK
304
LIBRARY NODE
305
LIBRARY OF RULES
306
RECEIVING NODE A
ACCESS CONTROL PROXY
RECEIVING NODE B
ACCESS CONTROL PROXY

FIG. 3


RECEIVE PACKET
101
SEARCH FOR PERTINENT RULE AT FIREWALL
102
104
103
NO
PERFORM DROP OR PASS RULE ACTION
RULE ACTION = REFER TO ACCESS CONTROL PROXY?
YES
105
SEND PACKET TO ACCESS CONTROL PROXY
ANALYZE CONTENT OF PAYLOAD
106
SELECT PERTINENT ACCESS RULE
107
IMPLEMENT ACCESS RULE
108