

UNITED STATES PATENT AND TRADEMARK OFFICE
BEFORE THE PATENT TRIAL AND APPEAL BOARD

CISCO SYSTEMS, INC.,
Petitioner,
- vs. -
CENTRIPETAL NETWORKS, INC.,
Patent Owner

Case No.: IPR2018-01512

US Patent 9,565,213

PETITIONER'S DEMONSTRATIVE EXHIBITS
PTAB ORAL ARGUMENT
December 18, 2019

more networks. The functionality may be in a single manner, or may be located in a single manner (e.g., a server, a client computer, etc.).

Aspects of the disclosure have been described by way of illustrative embodiments thereof. Numerous modifications, variations, and adaptations will occur to those of ordinary skill in the art from a review of the disclosure. For example, one of ordinary skill in the art will recognize that the steps illustrated in the illustrative figure may be performed in other than the recited order, and steps illustrated may be optional.

What is claimed is:

1. A method comprising:

receiving, by each of a plurality of packet security gateways associated with a security policy management server and from the security policy management server, a dynamic security policy that comprises at least one rule specifying application-layer packet-header information and a packet transformation function comprising a packet digest logging function to be performed on packets comprising the application-layer packet-header information;

receiving, by a packet security gateway of the plurality of packet security gateways, packets associated with a network protected by the packet security gateway;

identifying, by the packet security gateway, from amongst the packets associated with the network protected by the packet security gateway, and on a packet-by-packet basis, one or more packets comprising the application-layer packet-header information;

performing, by the packet security gateway and on a packet-by-packet basis, the packet transformation function on each of the one or more packets comprising the application-layer packet-header information, wherein the performing the packet transformation function comprises

identifying a subset of information specified by the packet digest logging function for each of the one or more packets comprising the application-layer packet-header information;

generating, for each of the one or more packets comprising the application-layer packet-header information, a record comprising the subset of information specified by the packet digest logging function;

reformatting, for each of the one or more packets comprising the application-layer packet-header information, the subset of information specified by the packet digest logging function, the subset of information reformatting to conform with a logging system standard; and

routing, by the packet security gateway and on a packet-by-packet basis, to a monitoring device each of the one or more packets corresponding to the application-layer packet-header information in response to the performing the packet transformation function.

2. The method of claim 1, wherein the performing the packet transformation function on the packets comprising the application-layer packet-header information, and wherein the packet transformation function comprises forwarding the packets to a security gateway, the each of the one or more packets comprising the application-layer packet-header information toward its respective destination.

3. The method of claim 1, wherein the performing the packet transformation function on the packets comprising the application-layer packet-header information, and wherein the performing the packet

transformation function comprises a packet digest logging function to be performed on packets

What is claimed is:

1. A method comprising:

receiving, by each of a plurality of packet security gateways associated with a security policy management server and from the security policy management server, a dynamic security policy that comprises at least one rule specifying application-layer packet-header information and a packet transformation function comprising a packet digest logging function to be performed on packets comprising the application-layer packet-header information;

receiving, by a packet security gateway of the plurality of packet security gateways, packets associated with a network protected by the packet security gateway;

identifying, by the packet security gateway, from amongst the packets associated with the network protected by the packet security gateway, and on a packet-by-packet basis, one or more packets comprising the application-layer packet-header information;

performing, by the packet security gateway and on a packet-by-packet basis, the packet transformation function on each of the one or more packets comprising the application-layer packet-header information, wherein the performing the packet transformation function comprises

identifying a subset of information specified by the packet digest logging function for each of the one or more packets comprising the application-layer packet-header information;

generating, for each of the one or more packets comprising the application-layer packet-header information, a record comprising the subset of information specified by the packet digest logging function; and

reformatting, for each of the one or more packets comprising the application-layer packet-header information, the subset of information specified by the packet digest logging function in accordance with a logging system standard; and

routing, by the packet security gateway and on a packet-by-packet basis, to a monitoring device each of the one or more packets corresponding to the application-layer packet-header information in response to the performing the packet transformation function.

packet transformation function comprising a packet digest logging function to be performed on packets

CISCO EXHIBIT 1001
9565213 Patent

more networks. The functionality may be distributed in any manner, or may be located in a single computing device (e.g., a server, a client computer, etc.).

Aspects of the disclosure have been described in terms of illustrative embodiments thereof. Numerous other embodiments, modifications, and variations within the scope and spirit of the appended claims will occur to persons of ordinary skill in the art from a review of this disclosure. For example, one of ordinary skill in the art will appreciate that the steps illustrated in the illustrative figures may be performed in other than the recited order, and that one or more steps illustrated may be optional.

What is claimed is:

1. A method comprising:

receiving, by each of a plurality of packet security gateways associated with a security policy management server and from the security policy management server, a dynamic security policy that comprises at least one rule specifying application-layer packet-header information and a packet transformation function comprising a packet digest logging function to be performed on packets comprising the application-layer packet-header

transformation function on comprises dropping, by the packet security gateway, the each of the one or more packets comprising the application-layer packet-header information.

4. The method of claim 1, wherein the application-layer packet-header information identifies one or more hypertext transfer protocol (HTTP) packets, and wherein identifying the one or more packets comprising the application-layer packet-header information comprises determining by the packet security gateway that the one or more packets comprising the application-layer packet-header information are amongst the one or more HTTP packets.

5. The method of claim 4, wherein the one or more HTTP packets comprise an HTTP GET method call, and wherein determining that the one or more packets comprising the application-layer packet-header information are amongst the one or more HTTP packets comprises determining that the one or more packets comprising the application-layer packet-header information are associated with the HTTP GET method call.

6. The method of claim 5, wherein the HTTP GET method call specifies a uniform resource identifier (URI), and wherein determining that the one or more packets comprise

What is claimed is:

1. A method comprising:
receiving, by each of a plurality of packet security gateways associated with a security policy management server and from the security policy management server, a dynamic security policy that comprises at least one rule specifying application-layer packet-header information and a packet transformation function compris-

header information, and wherein the performing the packet

digest logging function to be performed on packets

CISCO EXHIBIT 1001
9565213 Patent

25

more networks. The functionality may be distributed in any manner, or may be located in a single computing device (e.g., a server, a client computer, etc.).

Aspects of the disclosure have been described in terms of illustrative embodiments thereof. Numerous other embodiments, modifications, and variations within the scope and spirit of the appended claims will occur to persons of ordinary skill in the art from a review of this disclosure. For example, one of ordinary skill in the art will appreciate that the steps illustrated in the illustrative figures may be performed in other than the recited order, and that one or more steps illustrated may be optional.

What is claimed is:

1. A method comprising:
receiving, by each of a plurality of packet security gateways associated with a security policy management server and from the security policy management server, a dynamic security policy that comprises at least one rule specifying application-layer packet-header information and a packet transformation function comprising a packet digest logging function to be performed on packets comprising the application-layer packet-header information;

receiving, by a packet security gateway of the plurality of

26

transformation function on comprises dropping, by the packet security gateway, the each of the one or more packets comprising the application-layer packet-header information.

4. The method of claim 1, wherein the application-layer packet-header information identifies one or more hypertext transfer protocol (HTTP) packets, and wherein identifying the one or more packets comprising the application-layer packet-header information comprises determining by the packet security gateway that the one or more packets comprising the application-layer packet-header information are amongst the one or more HTTP packets.

5. The method of claim 4, wherein the one or more HTTP packets comprise an HTTP GET method call, and wherein determining that the one or more packets comprising the application-layer packet-header information are amongst the one or more HTTP packets comprises determining that the one or more packets comprising the application-layer packet-header information are associated with the HTTP GET method call.

6. The method of claim 5, wherein the HTTP GET method call specifies a uniform resource identifier (URI), and wherein determining that the one or more packets comprising the application-layer packet-header information are associated with the HTTP GET method call comprises determining that the one or more packets comprising the

mation and a packet transformation function comprising a packet digest logging function to be performed on packets comprising the application-layer packet-header information;

routing, by the packet security gateway and on a packet-by-packet basis, to a monitoring device each of the one or more packets corresponding to the application-layer packet-header information in response to the performing the packet transformation function.

2. The method of claim 1, wherein the at least one rule indicates performing an accept packet transformation function on the packets comprising the application-layer packet-header information, and wherein the performing the packet transformation function comprises forwarding, by the packet security gateway, the each of the one or more packets comprising the application-layer packet-header information toward its respective destination.

3. The method of claim 1, wherein the at least one rule indicates performing a deny packet transformation function on the packets comprising the application-layer packet-header information, and wherein the performing the packet

packets comprising the application-layer packet-header information comprises determining by the packet security gateway that each of the one or more packets comprising the application-layer packet-header information corresponds to at least one of the one or more transport-layer protocols, has a source address within the range of source addresses, a source port within the range of source ports, a destination address within the range of destination addresses, and a destination port within the range of destination ports.

10. A method comprising:
receiving, by each of a plurality of packet security gateways associated with a security policy management server and from the security policy management server, a dynamic security policy that comprises at least one rule specifying packet-identification criteria and a packet transformation function comprising a packet digest logging function to be performed on packets

CISCO EXHIBIT 1001
9565213 Patent

25

more networks. The functionality may be distributed in any manner, or may be located in a single computing device (e.g., a server, a client computer, etc.).

Aspects of the disclosure have been described in terms of illustrative embodiments thereof. Numerous other embodiments, modifications, and variations within the scope and spirit of the appended claims will occur to persons of ordinary skill in the art from a review of this disclosure. For example, one of ordinary skill in the art will appreciate that the steps illustrated in the illustrative figures may be performed in other than the recited order, and that one or more steps illustrated may be optional.

What is claimed is:

1. A method comprising:

receiving, by each of a plurality of packet security gateways associated with a security policy management server and from the security policy management server, a dynamic security policy that comprises at least one rule specifying application-layer packet-header information and a packet transformation function comprising a packet digest logging function to be performed on packets comprising the application-layer packet-header information;

receiving, by a packet security gateway of the plurality of packet security gateways, packets associated with a network protected by the packet security gateway;

identifying, by the packet security gateway, from amongst the packets associated with the network protected by the packet security gateway, and on a packet-by-packet basis, one or more packets comprising the application-layer packet-header information;

performing, by the packet security gateway and on a packet-by-packet basis, the packet transformation function

26

on packets comprising the application-layer packet-header information.

4. The method of claim 1, wherein the application-layer packet-header information identifies one or more hypertext transfer protocol (HTTP) packets, and wherein identifying the one or more packets comprising the application-layer packet-header information comprises determining by the packet security gateway that the one or more packets comprising the application-layer packet-header information are amongst the one or more HTTP packets.

5. The method of claim 4, wherein the one or more HTTP packets comprise an HTTP GET method call, and wherein determining that the one or more packets comprising the application-layer packet-header information are amongst the one or more HTTP packets comprises determining that the one or more packets comprising the application-layer packet-header information are associated with the HTTP GET method call.

6. The method of claim 5, wherein the HTTP GET method call specifies a uniform resource identifier (URI), and wherein determining that the one or more packets comprising the application-layer packet-header information are associated with the HTTP GET method call comprises determining that the one or more packets comprising the application-layer packet-header information originated from or are destined for a network address corresponding to the URI.

7. The method of claim 4, wherein the one or more HTTP packets comprise an HTTP PUT method call, and wherein determining that the one or more packets comprising the application-layer packet-header information are amongst the one or more HTTP packets comprises determining that the

identifying, by the packet security gateway, from amongst the packets associated with the network protected by the packet security gateway, and on a packet-by-packet basis, one or more packets comprising the application-layer packet-header information;

performing, by the packet security gateway and on a packet-by-packet basis, the packet transformation function on the packets comprising the application-layer packet-header information, and wherein the performing the packet

transformation function comprises dropping, by the packet security gateway, the each of the one or more packets comprising the application-layer packet-header information.

CISCO EXHIBIT 1001
9565213 Patent

more networks. The functionality may be distributed in any manner, or may be located in a single computing device (e.g., a server, a client computer, etc.).

Aspects of the disclosure have been described in terms of illustrative embodiments thereof. Numerous other embodiments, modifications, and variations within the scope and

transformation function on comprises dropping, by the packet security gateway, the each of the one or more packets comprising the application-layer packet-header information.

4. The method of claim 1, wherein the application-layer packet-header information identifies one or more hypertext transfer protocol (HTTP) packets, and wherein identifying

identifying a subset of information specified by the packet digest logging function for each of the one or more packets comprising the application-layer packet-header information;

function on each of the one or more packets comprising the application-layer packet-header information, wherein the performing the packet transformation function comprises:

identifying a subset of information specified by the packet digest logging function for each of the one or more packets comprising the application-layer packet-header information;

generating, for each of the one or more packets comprising the application-layer packet-header information, a record comprising the subset of information specified by the packet digest logging function; and reformatting, for each of the one or more packets comprising the application-layer packet-header information, the subset of information specified by the packet digest logging function in accordance with a logging system standard; and routing, by the packet security gateway and on a packet-by-packet basis, to a monitoring device each of the one or more packets corresponding to the application-layer packet-header information in response to the performing the packet transformation function.

2. The method of claim 1, wherein the at least one rule indicates performing an accept packet transformation function on the packets comprising the application-layer packet-header information, and wherein the performing the packet transformation function comprises forwarding, by the packet security gateway, the each of the one or more packets comprising the application-layer packet-header information toward its respective destination.

3. The method of claim 1, wherein the at least one rule indicates performing a deny packet transformation function on the packets comprising the application-layer packet-header information, and wherein the performing the packet

one or more packets comprising the application-layer packet-header information are associated with the HTTP PUT method call.

8. The method of claim 7, wherein the HTTP PUT method call specifies a uniform resource identifier (URI), and wherein determining that the one or more packets comprising the application-layer packet-header information are associated with the HTTP PUT method call comprises determining that the one or more packets comprising the application-layer packet-header information originated from or are destined for a network address corresponding to the URI.

9. The method of claim 1, wherein the at least one rule comprises a five-tuple specifying one or more transport-layer protocols, a range of source addresses, a range of source ports, a range of destination addresses, and a range of destination ports, and wherein identifying the one or more packets comprising the application-layer packet-header information comprises determining by the packet security gateway that each of the one or more packets comprising the application-layer packet-header information corresponds to at least one of the one or more transport-layer protocols, has a source address within the range of source addresses, a source port within the range of source ports, a destination address within the range of destination addresses, and a destination port within the range of destination ports.

10. A method comprising: receiving, by each of a plurality of packet security gateways associated with a security policy management server and from the security policy management server, a dynamic security policy that comprises at least one rule specifying packet-identification criteria and a packet transformation function comprising a packet digest logging function to be performed on packets

more networks. The functionality may be distributed in any manner, or may be located in a single computing device (e.g., a server, a client computer, etc.).

Aspects of the disclosure have been described in terms of illustrative embodiments thereof. Numerous other embodiments, modifications, and variations within the scope and spirit of the appended claims will occur to persons of ordinary skill in the art from a review of this disclosure. For example, one of ordinary skill in the art will appreciate that the steps illustrated in the illustrative figures may be performed in other than the recited order, and that one or more

transformation function on comprises dropping, by the packet security gateway, the each of the one or more packets comprising the application-layer packet-header information.

4. The method of claim 1, wherein the application-layer packet-header information identifies one or more hypertext transfer protocol (HTTP) packets, and wherein identifying the one or more packets comprising the application-layer packet-header information comprises determining by the packet security gateway that the one or more packets comprising the application-layer packet-header information are associated with the HTTP method call.

generating, for each of the one or more packets comprising the application-layer packet-header information, a record comprising the subset of information specified by the packet digest logging function; and

identifying a subset of information specified by the packet digest logging function for each of the one or more packets comprising the application-layer packet-header information.

generating, for each of the one or more packets comprising the application-layer packet-header information, a record comprising the subset of information specified by the packet digest logging function; and reformatting, for each of the one or more packets comprising the application-layer packet-header information, the subset of information specified by the packet digest logging function in accordance with a logging system standard; and routing, by the packet security gateway and on a packet-by-packet basis, to a monitoring device each of the one or more packets corresponding to the application-layer packet-header information in response to the performing the packet transformation function.

2. The method of claim 1, wherein the at least one rule indicates performing an accept packet transformation function on the packets comprising the application-layer packet-header information, and wherein the performing the packet transformation function comprises forwarding, by the packet security gateway, the each of the one or more packets comprising the application-layer packet-header information toward its respective destination.

3. The method of claim 1, wherein the at least one rule indicates performing a deny packet transformation function on the packets comprising the application-layer packet-header information, and wherein the performing the packet

call specifies a uniform resource identifier (URI), and wherein determining that the one or more packets comprising the application-layer packet-header information are associated with the HTTP PUT method call comprises determining that the one or more packets comprising the application-layer packet-header information originated from or are destined for a network address corresponding to the URI.

9. The method of claim 1, wherein the at least one rule comprises a five-tuple specifying one or more transport-layer protocols, a range of source addresses, a range of source ports, a range of destination addresses, and a range of destination ports, and wherein identifying the one or more packets comprising the application-layer packet-header information comprises determining by the packet security gateway that each of the one or more packets comprising the application-layer packet-header information corresponds to at least one of the one or more transport-layer protocols, has a source address within the range of source addresses, a source port within the range of source ports, a destination address within the range of destination addresses, and a destination port within the range of destination ports.

10. A method comprising: receiving, by each of a plurality of packet security gateways associated with a security policy management server and from the security policy management server, a dynamic security policy that comprises at least one rule specifying packet-identification criteria and a packet transformation function comprising a packet digest logging function to be performed on packets

more networks. The functionality may be distributed in any manner, or may be located in a single computing device (e.g., a server, a client computer, etc.).

Aspects of the disclosure have been described in terms of illustrative embodiments thereof. Numerous other embodiments, modifications, and variations within the scope and spirit of the appended claims will occur to persons of ordinary skill in the art from a review of this disclosure. For example, one of ordinary skill in the art will appreciate that the steps illustrated in the illustrative figures may be performed in other than the recited order, and that one or more steps illustrated may be optional.

What is claimed is:

1. A method comprising:
receiving, by each of a plurality of packet security gateways associated with a security policy management server and from the security policy management server, a dynamic security policy that comprises at least one rule specifying application-layer packet-header information.

transformation function on comprises dropping, by the packet security gateway, the each of the one or more packets comprising the application-layer packet-header information.

4. The method of claim 1, wherein the application-layer packet-header information identifies one or more hypertext transfer protocol (HTTP) packets, and wherein identifying the one or more packets comprising the application-layer packet-header information comprises determining by the packet security gateway that the one or more packets comprising the application-layer packet-header information are amongst the one or more HTTP packets.

5. The method of claim 4, wherein the one or more HTTP packets comprise an HTTP GET method call, and wherein determining that the one or more packets comprising the application-layer packet-header information are amongst the one or more HTTP packets comprises determining that the one or more packets comprising the application-layer packet-header information are associated with the HTTP GET method call.

routing, by the packet security gateway and on a packet-by-packet basis, to a monitoring device each of the one or more packets corresponding to the application-layer packet-header information in response to the performing the packet transformation function.

What is claimed is:
routing, by the packet security gateway and on a packet-by-packet basis, to a monitoring device each of the one or more packets corresponding to the application-layer packet-header information in response to the performing the packet transformation function.

2. The method of claim 1, wherein the at least one rule indicates performing an accept packet transformation function on the packets comprising the application-layer packet-header information, and wherein the performing the packet transformation function comprises forwarding, by the packet security gateway, the each of the one or more packets comprising the application-layer packet-header information toward its respective destination.

3. The method of claim 1, wherein the at least one rule indicates performing a deny packet transformation function on the packets comprising the application-layer packet-header information, and wherein the performing the packet

transformation function comprises dropping, by the packet security gateway, the each of the one or more packets comprising the application-layer packet-header information. The method of claim 1, wherein the application-layer packet-header information comprises determining by the packet security gateway that each of the one or more packets comprising the application-layer packet-header information corresponds to at least one of the one or more transport-layer protocols, has a source address within the range of source addresses, a source port within the range of source ports, a destination address within the range of destination addresses, and a destination port within the range of destination ports.

10. A method comprising:
receiving, by each of a plurality of packet security gateways associated with a security policy management server and from the security policy management server, a dynamic security policy that comprises at least one rule specifying packet-identification criteria and a packet transformation function comprising a packet digest logging function to be performed on packets

CISCO EXHIBIT 1001
9565213 Patent

identifying a subset of information specified by the packet digest logging function for each of the one or more packets comprising the application-layer packet-header information;
 40 generating, for each of the one or more packets comprising the application-layer packet-header information, a record comprising the subset of information specified by the packet digest logging function; and
 45 reformatting, for each of the one or more packets comprising the application-layer packet-header information, the subset of information specified by the packet digest logging function in accordance with a logging system standard; and
 50 routing, by the packet security gateway and on a packet-by-packet basis, to a monitoring device each of the one or more packets corresponding to the application-layer packet-header information in response to the performing the packet transformation function.
 55 2. The method of claim 1, wherein the at least one rule indicates performing an accept packet transformation function on the packets comprising the application-layer packet-header information, and wherein the performing the packet transformation function comprises forwarding, by the packet security gateway, the each of the one or more packets
 60 comprising the application-layer packet-header information toward its respective destination.
 3. The method of claim 1, wherein the at least one rule indicates performing a deny packet transformation function on the packets comprising the application-layer packet-header information, and wherein the performing the packet

call specifies a uniform resource identifier (URI), and wherein determining that the one or more packets comprising the application-layer packet-header information are associated with the HTTP PUT method call comprises determining that the one or more packets comprising the application-layer packet-header information originated from or are destined for a network address corresponding to the
 45 URI.
 9. The method of claim 1, wherein the at least one rule comprises a five-tuple specifying one or more transport-layer protocols, a range of source addresses, a range of source ports, a range of destination addresses, and a range of destination ports, and wherein identifying the one or more packets comprising the application-layer packet-header information comprises determining by the packet security gateway that each of the one or more packets comprising the application-layer packet-header information corresponds to
 55 at least one of the one or more transport-layer protocols, has a source address within the range of source addresses, a source port within the range of source ports, a destination address within the range of destination addresses, and a destination port within the range of destination ports.
 10. A method comprising:
 60 receiving, by each of a plurality of packet security gateways associated with a security policy management server and from the security policy management server, a dynamic security policy that comprises at least one rule specifying packet-identification criteria and a packet transformation function comprising a packet digest logging function to be performed on packets

CISCO EXHIBIT 1001
 9565213 Patent

US 9,565,213 B2

27
 corresponding to the packet-identification criteria, wherein the packet-identification criteria comprises a Differentiated Service Code Point (DSCP) selector;
 30 receiving, by a packet security gateway of the plurality of

28
 12. The method of claim 10, comprising:
 35 temporarily storing data in a memory of the packet security gateway, the data comprising the subset of information specified by the packet digest logging function for each of the one or more packets corre-

wherein the packet-identification criteria comprises a Differentiated Service Code Point (DSCP) selector;

more packets corresponding to the packet-identification criteria;
 40 generating, for each of the one or more packets corresponding to the packet-identification criteria, a record comprising the subset of information specified by the packet digest logging function; and
 45 reformatting, for each of the one or more packets corresponding to the packet-identification criteria, the subset of information specified by the packet digest logging function in accordance with a logging system standard; and
 50 routing, by the packet security gateway and on a packet-by-packet basis, to a monitoring device each of the one or more packets corresponding to the packet-identification criteria in response to the performing the packet digest logging function.
 55 11. The method of claim 10, wherein the subset of

comprises reformatting the subset of information specified by the packet digest logging function in accordance with a syslog logging system standard.
 15. The method of claim 10, wherein the packet-identification criteria comprises a five-tuple specifying one or more transport-layer protocols, a range of source addresses, a range of source ports, a range of destination addresses, and a range of destination ports, and wherein identifying the one or more packets corresponding to the packet-identification criteria comprises determining by the packet security gateway that each of the one or more packets corresponding to the packet-identification criteria corresponds to at least one of the one or more transport-layer protocols, has a source address within the range of source addresses, a source port within the range of source ports, a destination address within the range of destination addresses, and a destination port within the range of destination ports.
 35 16. The method of claim 10, wherein the packet-identi-

more networks. The functionality may be distributed in any manner, or may be located in a single computing device (e.g., a server, a client computer, etc.).

Aspects of the disclosure have been described in terms of illustrative embodiments thereof. Numerous other embodiments, modifications, and variations within the scope and spirit of the appended claims will occur to persons of ordinary skill in the art from a review of this disclosure. For example, one of ordinary skill in the art will appreciate that the steps illustrated in the illustrative figures may be performed in other than the recited order, and that one or more steps illustrated may be optional.

What is claimed is:

transformation function on comprises dropping, by the packet security gateway, the each of the one or more packets comprising the application-layer packet-header information.

4. The method of claim 1, wherein the application-layer packet-header information identifies one or more hypertext transfer protocol (HTTP) packets, and wherein identifying the one or more packets comprising the application-layer packet-header information comprises determining by the packet security gateway that the one or more packets comprising the application-layer packet-header information are amongst the one or more HTTP packets.

5. The method of claim 4, wherein the one or more HTTP packets comprise an HTTP GET method call, and wherein

4. The method of claim 1, wherein the application-layer packet-header information identifies one or more hypertext transfer protocol (HTTP) packets, and wherein identifying the one or more packets comprising the application-layer packet-header information comprises determining by the packet security gateway that the one or more packets comprising the application-layer packet-header information are amongst the one or more HTTP packets.

comprising the application-layer packet-header information toward its respective destination.

3. The method of claim 1, wherein the at least one rule indicates performing a deny packet transformation function on the packets comprising the application-layer packet-header information, and wherein the performing the packet

ways associated with a security policy management server and from the security policy management server, a dynamic security policy that comprises at least one rule specifying packet-identification criteria and a packet transformation function comprising a packet digest logging function to be performed on packets

CISCO EXHIBIT 1001
9565213 Patent

more networks. The functionality may be distributed in any manner, or may be located in a single computing device (e.g., a server, a client computer, etc.).

Aspects of the disclosure have been described in terms of illustrative embodiments thereof. Numerous other embodiments, modifications, and variations within the scope and spirit of the appended claims will occur to persons of ordinary skill in the art from a review of this disclosure. For example, one of ordinary skill in the art will appreciate that the steps illustrated in the illustrative figures may be performed in other than the recited order, and that one or more steps illustrated may be optional.

What is claimed is:

1. A method comprising:
receiving, by each of a plurality of packet security gateways associated with a security policy management server and from the security policy management server, a dynamic security policy that comprises at least one rule specifying application-layer packet-header information and a packet transformation function comprising

transformation function on comprises dropping, by the packet security gateway, the each of the one or more packets comprising the application-layer packet-header information.

4. The method of claim 1, wherein the application-layer packet-header information identifies one or more hypertext transfer protocol (HTTP) packets, and wherein identifying the one or more packets comprising the application-layer packet-header information comprises determining by the packet security gateway that the one or more packets comprising the application-layer packet-header information are amongst the one or more HTTP packets.

5. The method of claim 4, wherein the one or more HTTP packets comprise an HTTP GET method call, and wherein determining that the one or more packets comprising the application-layer packet-header information are amongst the one or more HTTP packets comprises determining that the one or more packets comprising the application-layer packet-header information are associated with the HTTP GET method call.

6. The method of claim 5, wherein the HTTP GET method call comprises a uniform resource identifier (URI), and

5. The method of claim 4, wherein the one or more HTTP packets comprise an HTTP GET method call, and wherein determining that the one or more packets comprising the application-layer packet-header information are amongst the one or more HTTP packets comprises determining that the one or more packets comprising the application-layer packet-header information are associated with the HTTP GET method call.

more networks. The functionality may be distributed in any manner, or may be located in a single computing device (e.g., a server, a client computer, etc.).

Aspects of the disclosure have been described in terms of illustrative embodiments thereof. Numerous other embodiments, modifications, and variations within the scope and spirit of the appended claims will occur to persons of ordinary skill in the art from a review of this disclosure. For example, one of ordinary skill in the art will appreciate that the claims illustrated in the illustrative figures may be re-

transformation function on comprises dropping, by the packet security gateway, the each of the one or more packets comprising the application-layer packet-header information.

4. The method of claim 1, wherein the application-layer packet-header information identifies one or more hypertext transfer protocol (HTTP) packets, and wherein identifying the one or more packets comprising the application-layer packet-header information comprises determining by the packet security gateway that the one or more packets comprising the application-layer packet-header information are

7. The method of claim 4, wherein the one or more HTTP packets comprise an HTTP PUT method call, and wherein determining that the one or more packets comprising the application-layer packet-header information are amongst the one or more HTTP packets comprises determining that the one or more packets comprising the application-layer packet-header information are associated with the HTTP PUT method call.

on the packets comprising the application-layer packet-header information, and wherein the performing the packet transformation function comprises forwarding, by the packet security gateway, the each of the one or more packets comprising the application-layer packet-header information toward its respective destination.

3. The method of claim 1, wherein the at least one rule indicates performing a deny packet transformation function on the packets comprising the application-layer packet-header information, and wherein the performing the packet

address within the range of destination addresses, and a destination port within the range of destination ports.

10. A method comprising: receiving, by each of a plurality of packet security gateways associated with a security policy management server and from the security policy management server, a dynamic security policy that comprises at least one rule specifying packet-identification criteria and a packet transformation function comprising a packet digest logging function to be performed on packets

described herein. As used herein, a **dynamic security policy** includes any rule, message, instruction, file, data structure, or the like that specifies criteria corresponding to one or more packets and identifies a packet transformation function to be performed on packets corresponding to the specified criteria. Optionally, a dynamic security policy may further

work traffic. In some embodiments, the list of known network addresses associated with malicious network traffic

may be maintained, various embodiments in which aspects of the disclosure may be practiced. It is to be understood that other embodiments may be utilized, and structural and

FIG. 6 illustrates an exemplary network environment for implementing a monitoring service.

FIG. 7 illustrates an exemplary network environment that includes a secured network having multiple boundaries with unsecured networks.

FIG. 8 illustrates an exemplary network environment that includes multiple distinct secured networks.

FIG. 9 illustrates an exemplary secure LAN environment.

FIG. 10 illustrates an exemplary method for protecting a secured network.

As described herein, a dynamic security policy includes any rule, message, instruction, file, data structure, or the like that specifies criteria corresponding to one or more packets and identifies a packet transformation function to be performed on packets corresponding to the specified criteria. Optionally, a dynamic security policy may further specify one or more additional parameters as described herein.

Network environment 100 may include one or more packet security gateways and one or more security policy

CISCO EXHIBIT 1001
9565213 Patent

US 9,565,213 B2

management servers. For example, network environment 100 may include packet security gateways 112, 114, 116, and 118, and security policy management server 120. One or more security policy management servers may be associated with a protected network. For example, networks A-D 102, 104, 106, and 108 may each be distinct LANs associated with a common organization and may each form part of a protected network associated with security policy management server 120. Many network protocols route packets dynamically, and thus the path a given packet may take cannot be readily predicted. Accordingly it may be advantageous to locate a packet security gateway at each boundary between a protected network and an unprotected network. For example, packet security gateway 112 may be located at the boundary between network A 102 and network E 110. Similarly, packet security gateway 114 may be located at the

examine information associated with packets received by packet security gateway 112 and forward the packets to one or more packet transformation functions based on the examined information. For example, packet filter 214 may examine information associated with packets received by packet security gateway 112 (e.g., packets received from network E 110 via management interface 222 or network interface 206) and forward the packets to one or more of packet transformation functions 1-N 216, 218, and 220 based on the examined information.

As will be described in greater detail below, dynamic security policy 212 may include one or more rules and the configuration of packet filter 214 may be based on one or more of the rules included in dynamic security policy 212. For example, dynamic security policy 212 may include one or more rules specifying that packets having specified information

security policy management server 120's memory 502. For example, an administrator associated with security policy management server 120 may manually construct one or more dynamic security policies offline and then utilize security policy management server 120's management interface 510 to load such dynamic security policies into security policy management server 120's memory 502.

packet security gateway 1 400 may have only been required to compare the packet to five rules and packet security gateway 2 402 may have only been required to compare the packet to seven rules. In a worst case scenario, packet security gateway 1 400 may have only been required to compare the packet to M rules and packet security gateway 2 402 may have only been required to compare the packet to N rules. Moreover, the series configuration may enable packet security gateway 1 400 to begin implementing P₁ with respect to a subsequently received packet, while packet security gateway 2 402 simultaneously implements P₂ with respect to the packet forwarded by packet security gateway 1 400. Furthermore, the memory requirements for this scenario with packet security gateways in series may be comparable to M+N, whereas originally the combinatorially complete set of rules contained in a single packet security gateway may have required memory comparable to N*M.

FIG. 5 illustrates an exemplary security policy management server. Referring to FIG. 5, security policy management server 120 may include processor 500, memory 502, and network interface 504. One or more of processor 500, memory 502, and network interface 504 may be interconnected via data bus 506. Network interface 504 may interface security policy management server 120 with network 110. Memory 502 may include one or more program modules that when executed by processor 500, configure security policy management server 120 to perform functions described herein. It will be appreciated that as used herein

or to load one or more dynamic security policies into security policy management server 120's memory 502. For example, an administrator associated with security policy management server 120 may manually construct one or more dynamic security policies offline and then utilize security policy management server 120's management interface 510 to load such dynamic security policies into security policy management server 120's memory 502.

In some embodiments, security policy management server 120 may be configured to add, remove, or alter one or more dynamic security policies stored in memory 502 based on information received from one or more devices within network environment 100. For example, security policy management server 120's memory 502 may include a dynamic security policy having one or more rules that specify a list of network addresses known to be associated with malicious network traffic. Security policy management server 120 may be configured to automatically create or alter one or more of such rules as new network addresses associated with malicious network traffic are determined. For example, security policy management server 120 may receive updates (e.g., as part of a subscription) from malicious host tracker service 508. Malicious host tracker service 508 may aggregate information associated with malicious network traffic and updates received from malicious host tracker service 508 may include one or more network addresses that have been determined to be associated with malicious network traffic. Security policy management

CISCO EXHIBIT 1001
9565213 Patent

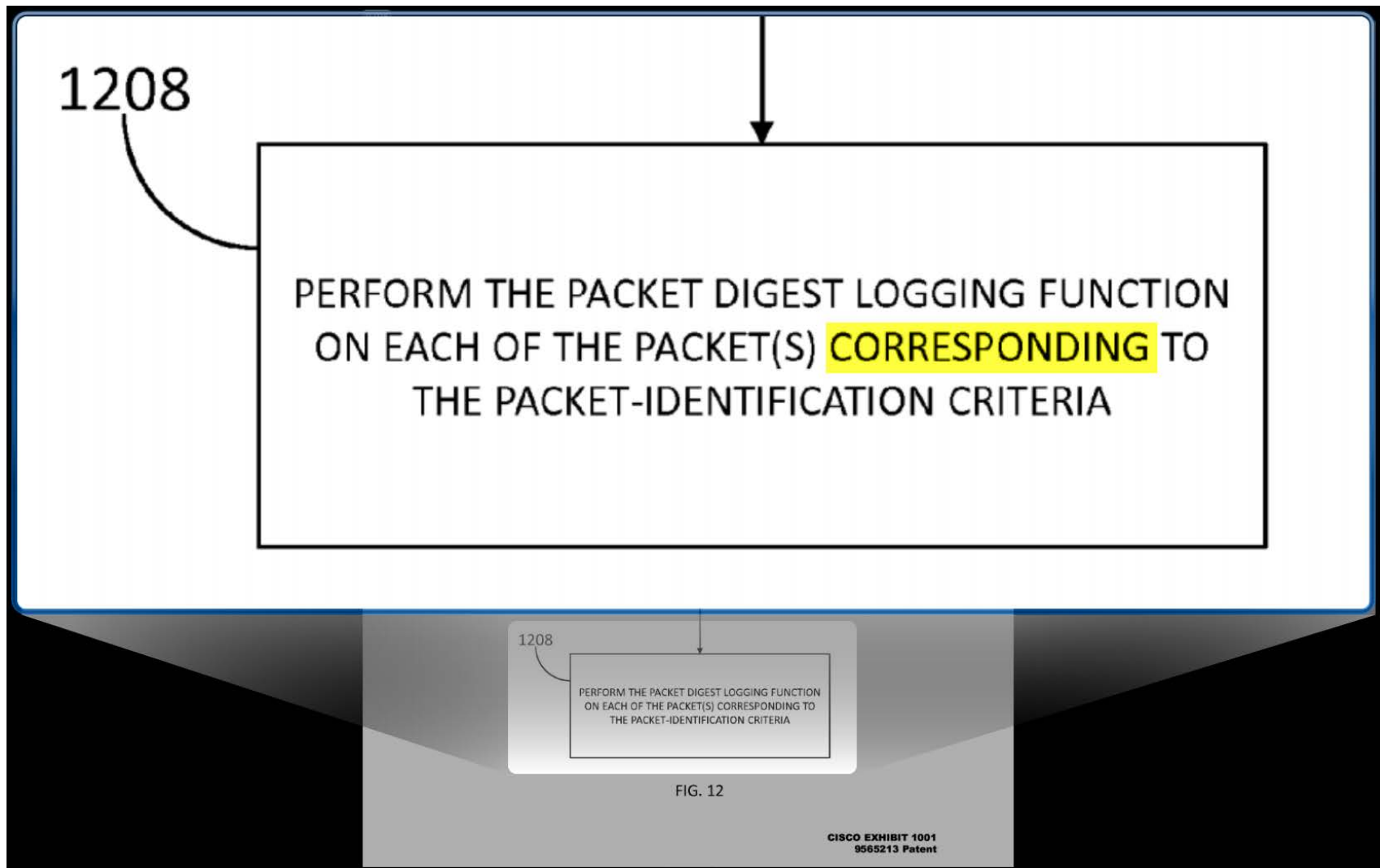
with malicious network traffic. Security policy management server 120 may be configured to automatically create or alter one or more of such rules as new network addresses associated with malicious network traffic are determined. For example, security policy management server 120 may receive updates (e.g. as part of a subscription) from malicious host tracker service 508. Malicious host tracker service 508 may aggregate information associated with malicious network traffic and updates received from malicious host tracker service 508 may include one or more network addresses that have been determined to be associated with malicious network traffic. Security policy management

transformation functions to perform the accept packet transformation function on the packet and forward the packet to network A 102.
It will be appreciated that utilizing multiple packet security

example, an administrator may configure security policy management server 120 in order to associate security policy management server 120 with one or more of packet security gateways 112, 114, 116, and 118. An administrator of

associated with these correlated updates from two or more services within one or more dynamic security policies. Security policy management server 120 may be configured to reduce the size of and/or to de-correlate dynamic security policies from the updates from the services.

or more rules within an existing dynamic security policy. For example, security policy management server 120 may construct a new dynamic security policy that includes a rule specifying one of the extracted network addresses and port numbers.



lost, and specifying that all packets should be blocked. For example, dynamic security policy 300 includes rules 1-4 302, 304, 306, and 308, each of which specifies a set of network addresses for which packets should be allowed, and rule 5 310 which specifies that all packets should be dropped. Thus, if rules 1-5 302, 304, 306, 308, and 310 are executed in order, dynamic security policy 300 will effectuate an allowlist service.

A dynamic security policy may also include one or more rules, the combination of which may effectuate a VoIP firewall service within a network environment. As will be discussed in greater detail below, a security policy management server may receive information associated with VoIP

network in recovering from an attack, by allowing the network to isolate itself from the attack or recover in a controlled manner.

A dynamic security policy may also include one or more rules, the combination of which may effectuate an enqueueing service within a network environment. A dynamic security policy that effectuates an enqueueing service may include one or more rules that specify sets of network addresses and packet transformation functions that queue packets in one or more queues corresponding to the sets. These queues may then be serviced at varying rates. For example, a dynamic security policy may include two rules, each of which specify a set of network addresses. A first of these rules may specify that packets corresponding to its

resources are strained). In some embodiments, one or more rules contained within a dynamic security policy may include an **arbitrary selector** which may correspond to one or more parameters or fields associated with a packet. For example, a dynamic security policy rule may include a **Differentiated Service Code Point (DSCP) selector** that corresponds to a DSCP field in an IP header. Thus, two

specifies a rule or set of rules that specify a larger set of network addresses for which packets should be forwarded (e.g., network addresses corresponding to all network devices that would be allowed under ordinary circumstances). The dynamic security policy may specify that the rules or rule sets should be implemented in time-shifted phases. That is, the dynamic security policy may specify that the first rule or rule set should be executed first, and that the second rule or rule set should be executed at a time after the time at which the first rule or rule set is executed, and the third rule or rule set should be executed at a time after the time at which the second rule or rule set is executed. Such a dynamic security policy may assist a

specifies a rule or set of rules that specify a larger set of network addresses for which packets should be forwarded (e.g., network addresses corresponding to all network devices that would be allowed under ordinary circumstances). The dynamic security policy may specify that the rules or rule sets should be implemented in time-shifted phases. That is, the dynamic security policy may specify that the first rule or rule set should be executed first, and that the second rule or rule set should be executed at a time after the time at which the first rule or rule set is executed, and the third rule or rule set should be executed at a time after the time at which the second rule or rule set is executed. Such a dynamic security policy may assist a

CISCO EXHIBIT 1001
9565213 Patent

ACNS

Page 44

Chapter 1 Understanding the ACNS 5.5 Network

Content Services Overview

Content Pre-Positioning

When pre-positioning content, the administrator can specify a group of content objects to be "pushed" to the remote branch offices in off-peak times, such as at night. This capability is advantageous because after the content is pre-positioned at a nearby Content Engine in the branch office, for example, it can be accessed using a higher-bandwidth connection than the typical low-bandwidth link between branch offices and remote corporate headquarters. Pre-positioning can be particularly advantageous for streaming objects.

For example, the training department at <http://www.bigcorp.com> might prepare an online training course that includes multiple videos and slides. The course might be posted at an internal website. If there is no ACNS network, an employee from a remote office taking the course online must wait a long time for the

- Authenticate, monitor, log, and control web access from end users to implement corporate policies regarding work environment and system security.

Objectives

- Reduce bandwidth usage by caching frequently accessed Internet content.
- Authenticate, monitor, log, and control web access from end users to implement corporate policies regarding work environment and system security.

Both objectives are made possible because of the Content Engine's special position as an "in-line" device between end users and the Internet.

For example, BigCorp Enterprise wants to make sure that the following policy, security, and budgetary objectives are met:

- Only full-time employees can access the web.
- No employee can access objectionable content through the corporate network.
- Viruses such as "Code Red" are prevented from being propagated inside the corporate network.
- Monthly payments to the company's ISP are reduced.

- Viruses such as "Code Red" are prevented from being propagated inside the corporate network.

User Authentication and Content Filtering

Content Engines can be configured to perform a number of authentication and content filtering services. Once the Content Engine receives a request, it does a number of things, including the following tasks:

- Authenticates the client, asks the client to provide a username and password so that it can consult an authentication, authorization, and accounting (AAA) server, and checks whether the client is allowed to access the web.

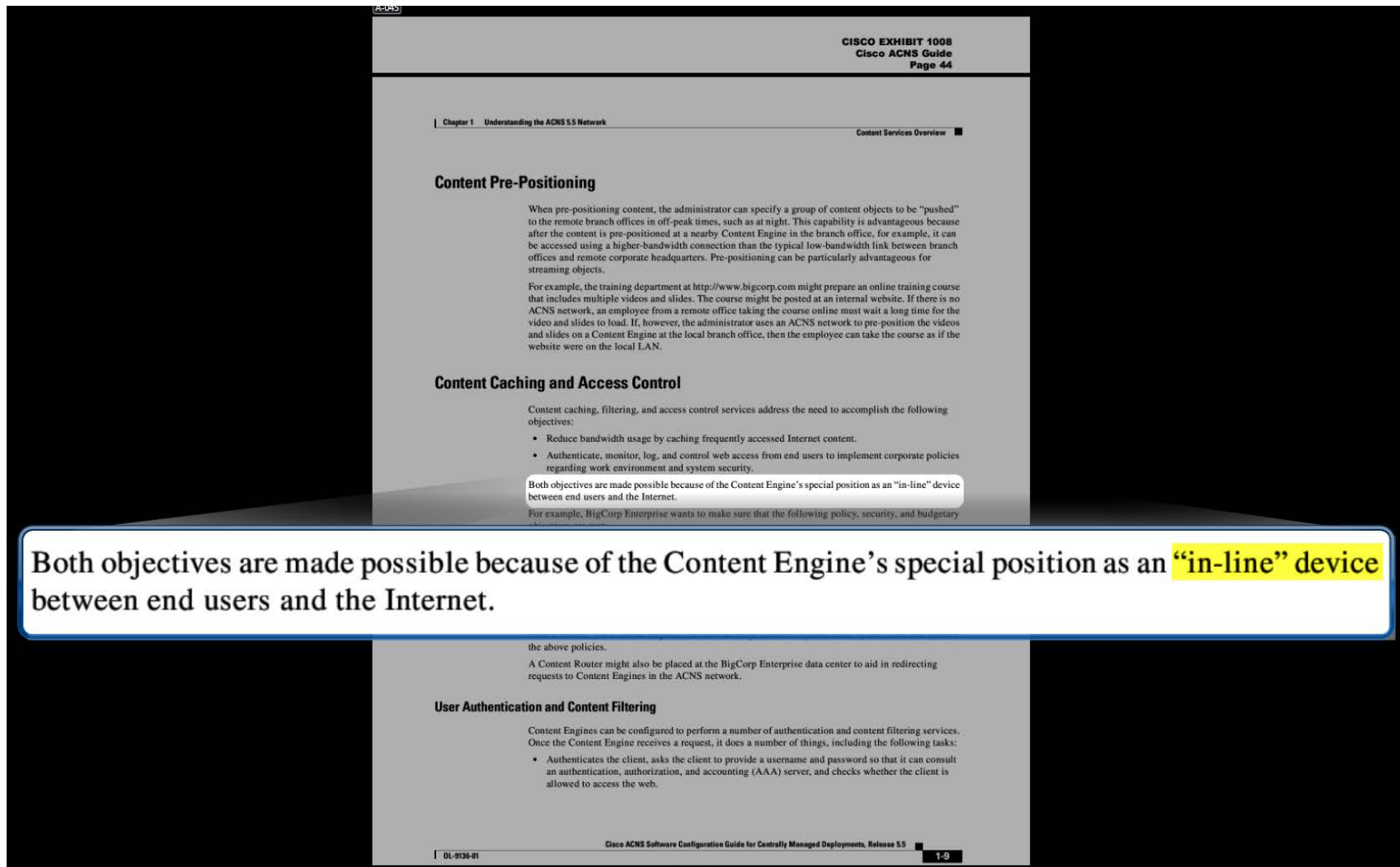
01-9136-01

Cisco ACNS Software Configuration Guide for Centrally Managed Deployments, Release 5.5

1-9

CISCO EXHIBIT 1008

Cisco ACNS Guide





CHAPTER 19

Using the Transaction Logs

This chapter explains how to use the transaction logs and contains the following sections:

- Understanding Transaction Log Formats, page 19-1
- Transaction Logging and NTLM Authentication, page 19-7
- Usage Guidelines for Log Files, page 19-8
- Enabling Transaction Logging with the Content Distribution Manager GUI, page 19-10
- Using WMT Transaction Logging, page 19-15
- Using Real-Time Transaction Logging, page 19-19

Transaction logs allow administrators to view the traffic that has passed through the Content Engine.

Transaction logs are stored in the `logs` directory on the Content Engine.

A Squid log format example looks like this:

```
1012429341.115 100 172.16.100.152 TCP_REFRESH_MISS/304 1100 GET  
http://www.cisco.com/images/homepage/news.gif - DIRECT/www.cisco.com -
```

and contains the following information: bytes received, bytes sent, status code, and request type.

A Squid log format example looks like this:

```
1012429341.115 100 172.16.100.152 TCP_REFRESH_MISS/304 1100 GET  
http://www.cisco.com/images/homepage/news.gif - DIRECT/www.cisco.com -
```

Cisco ACNS Software Configuration Guide for Centrally Managed Deployments, Release 5.5

01-9136-01

19-1

CISCO EXHIBIT 1008
Cisco ACNS Guide
Page 697

01-9136-01

Cisco ACNS Software Configuration Guide for Centrally Managed Deployments, Release 5.5

19-1

CISCO EXHIBIT 1008
Cisco ACNS Guide
Page 697

Chapter 19 Using the Transaction Logs

Understanding Transaction Log Formats

Extended Squid Log Format

The Extended Squid log format logs the associated username for each record in the log file, in addition to the fields logged by the Squid-style format, and is used for billing purposes. In this format the Rfc931 field associated with the Squid format is used to log the authorized user. This field always contains a "-" (dash) if no user information is available.

An Extended Squid-style log format example looks like this:

```
1012429341.115 100 172.16.100.152 TCP_MISS/302 184 GET http://www.cisco.com/cgi-bin/login
myloginname DIRECT/www.cisco.com
```

Apache-Style Transaction Logging

The Apache format is the Common Log File (CLF) format defined by the World Wide Web Consortium (W3C) working group. This format is compatible with many industry-standard log tools. For more information, see the W3C Common Log Format website at the following URL:

<http://www.w3.org/Daemon/User/Config/Logging.html>.

The Apache-style log file format is as follows:

```
remotehost rfc931 authuser date request status bytes
```

An Apache-style log file format example looks like this:

```
172.16.100.152 - - [Wed Jan 30 15:26:26 2002]
"GET/http://www.cisco.com/images/homepage/support.gif HTTP/1.0" 200 632
```

An Apache-style log file format example looks like this:

```
172.16.100.152 - - [Wed Jan 30 15:26:26 2002]
"GET/http://www.cisco.com/images/homepage/support.gif HTTP/1.0" 200 632
```

may not be preceded by an exclamation point (!). The exclamation point is used to negate all of the status codes that follow it, meaning that the value associated with the format token is logged if none of the status codes listed after the ! match the HTTP status code of the request. If any of the status codes listed after the ! match the HTTP status code of the request, then a hyphen (-) is logged.

For example, "%400,501 [User-Agent]!" logs the User-Agent header value on 400 errors and 501 errors (Bad Request, Not Implemented) only, whereas "%(200,304,302 [Referer])!" logs the Referer header value on all requests that did not return a normal status.

A-699

value on all requests that did not return a normal status.

19-2

Cisco ACNS Software Configuration Guide for Centrally Managed Deployments, Release 5.5

OL-9136-01

CISCO EXHIBIT 1008

Cisco ACNS Guide

Page 698

Chapter 19 Using the Transaction Logs

Understanding Transaction Log Formats

The custom format currently supports the following request headers:

- User-Agent
- Referer
- Host
- Cookie

The output of each of the following Request, Referer, and User-Agent format tokens specified in the custom log format string is always enclosed in double quotation marks in the transaction log entry:

```
%r
%{Referer}i
%{User-Agent}i
```

The %{Cookie}i format token is generated without the surrounding double quotation marks because the Cookie value itself can contain double quotes. The Cookie value can contain multiple attribute-value pairs that are separated by spaces. When you use the Cookie format token in a custom format string, we recommend that you position it in the last field of the format string so that the Cookie format token can be more easily parsed by transaction log reporting tools. Alternatively, if you use the format token string "%{Cookie}i", the Cookie header can be surrounded by single quotes.

The following command can be entered to generate the well-known Apache Combined Log Format:

```
transaction-logs format custom "%{d}t/%{b}t/%{Y}t:%{H}t:%{M}t:%{S}t
%{x}t %r %s %b %{Referer}i %{User-Agent}i"
```

The following transaction log entry example in the Apache Combined Format is configured by using the preceding custom format string:

```
[11/Jan/2003:02:12:44 -0800] "GET http://www.cisco.com/swa/i/site_tour_link.gif HTTP/1.1" 200
3436 "http://www.cisco.com/" "Mozilla/4.0 (compatible; MSIE 5.5; Windows NT 5.0)"
```

Table 19-1 Custom Format Log Format String Values

The following transaction log entry example in the Apache Combined Format is configured by using the preceding custom format string:

```
[11/Jan/2003:02:12:44 -0800] "GET http://www.cisco.com/swa/i/site_tour_link.gif HTTP/1.1" 200
3436 "http://www.cisco.com/" "Mozilla/4.0 (compatible; MSIE 5.5; Windows NT 5.0)"
```

%...i	Remote log name. Not implemented on the Content Engine, so a hyphen (-) is logged.
-------	--

Adding or Modifying Additional HTTP Request Methods for the Content Engine

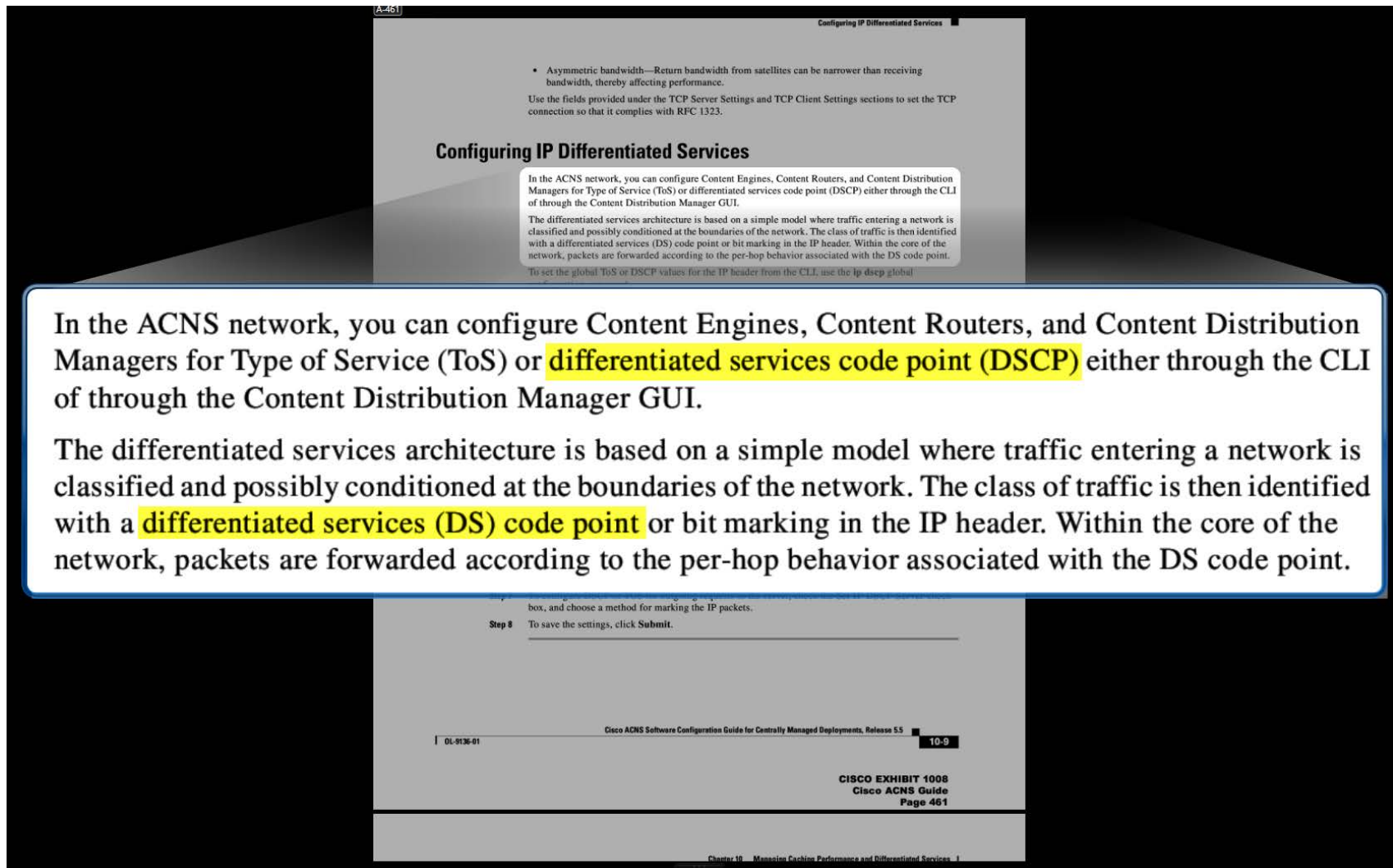
Content Engines accept or reject an HTTP request depending on whether the HTTP request method is supported. HTTP 1.1 request methods (for example, GET, HEAD, or POST) are supported by default. Nonstandard request methods, such as Web-Based Distributed Authoring and Versioning (WebDAV) are not.

When the Content Engine receives an HTTP request from a client using an unsupported method, ACNS software adds the method to the list of unsupported methods and sends an error to the client. You can use the Creating New HTTP Method for Content Engine window to add a method to the list of methods already supported by the Content Engine.

Adding or Modifying Additional HTTP Request Methods for the Content Engine

Content Engines accept or reject an HTTP request depending on whether the HTTP request method is supported. HTTP 1.1 request methods (for example, GET, HEAD, or POST) are supported by default. Nonstandard request methods, such as Web-Based Distributed Authoring and Versioning (WebDAV) are not.

When the Content Engine receives an HTTP request from a client using an unsupported method, ACNS software adds the method to the list of unsupported methods and sends an error to the client. You can use the Creating New HTTP Method for Content Engine window to add a method to the list of methods already supported by the Content Engine.



Appendix C Mapping ACNS Software CLI Commands to the Content Distribution Manager GUI		
CLI Commands Supported in the Content Distribution Manager GUI		
Table C-1 ACNS Software CLI to Content Distribution Manager GUI Map (continued)		
Command	GUI Navigation Path	Procedure Reference
hostname <i>name</i>	Devices > Devices > Device Activation	Modifying Content Engine Properties, page 13-9
http add-method <i>method</i>	Devices > Device Groups > Applications > Web > HTTP > HTTP Method	Adding or Modifying Additional HTTP Request Methods for the Content Engine, page 8-23
http age-multiplier <i>text num</i> <i>binary num</i>	Devices > Device Groups > Applications > Web > HTTP > HTTP Method	Configuring HTTP Cache Freshness, page 8-19
http append x-forwarded-for-header <i>multiple-ip-address</i>		
http append proxy-auth-header (<i>ipaddress</i> <i>hostname</i>)		
http append www-auth-header (<i>hostname</i> <i>ipaddress</i>)		
http authenticate-strip-ntlm		
http authentication cache timeout <i>minutes</i>	Devices > Device Groups > Applications > Web > HTTP > Authenticated HTTP Caching	Configuring Authenticated HTTP Cache Settings, page 8-13
http authentication cache max-entries <i>entries</i>		
http authentication header [401 407]		
http authentication realm <i>line</i>		
http cache-authenticated basic		
http cache-authenticated ntlm		
http cache-authenticated all		
http cache-cookies	Devices > Device Groups > Applications > Web > HTTP > HTTP Caching	Configuring HTTP Cache Settings, page 8-6
http cache-on-abort enable	Devices > Device Groups > Applications > Web > HTTP > Advanced HTTP Caching	Configuring Advanced HTTP Cache Settings, page 8-19
http cache-on-abort max-threshold <i>max_thresh</i>		
http cache-on-abort min-threshold <i>min_thresh</i>		
http cache-on-abort percent <i>percentthresh</i>		
http cache-vary-user-agent sub-string <i>string</i>	Devices > Device Groups > Applications > Web > HTTP > HTTP Cache User Agent	Configuring HTTP Cache Vary User Agent, page 8-25
cache-string <i>string</i>		

(K024)

This general policy-based mirroring allows network administrators to set network mirror policies for various network monitors including, for example, application identification appliances, IDSs, network loggers, analyzers, etc. These policies may be set based on device, user, topology, time-of-day, network events (e.g., triggers) mirrored to device availability, location and load. Given the breadth of dynamic mirroring capabilities, the capabilities exceed administrator manual set-up capabilities. Rapid dynamic mirrors can better optimize monitor and IDS devices and possibly other network devices and remove their physical location as a primary factor in their physical placement in the network topology. This provides the added structure of policy rules for

CISCO EXHIBIT 1009
Kjendal

devices attempting to perform specific portions of the application identification process. IPFIX generally uses the Stream Control Transmission Protocol as its Transport layer protocol, but also allows the use of the TCP and the UDP.

An application identification appliance 180 including the application identification function 200 is represented in FIG. 2 in a simplified way in relation to other devices of the network infrastructure 101 with which it communicates. The appliance 180 exchanges information and command signals with the network system control manager 125, which may be associated with the central policy server 103 and which may be embodied in one or more devices of the network infrastructure 101. An example of the control manager 125 is the NetSight® management system available from Eterasis Networks of Salem, N.H. The control manager 125 may include or at least control a dynamic mirroring engine used to mirror traffic by a network packet forwarding device as described more fully herein. Such a mirroring engine may exist in another device of the network infrastructure 101, including the device that does the mirroring.

The appliance 180 includes a management engine 184 and an application identification engine 186 that performs the

to the management engine 184 of the appliance 180. The configuration engine 174 exchanges information and instruction messages with the appliance 180 for the purposes of checking the status of the appliance 180, to modify the content of an applications signatures library 185 of the appliance 180, to add custom application identification components and to configure the identification engine 186. Messages are exchanged in any way suitable for device management including through, but not limited to MIBs and SNMP. That form of information and/or instruction exchange can be used for other actions described herein. Outputs of the identification engine 186 associated with the characteristics of mirrored frames of packets associated with one or more flows are forwarded to the management engine 184 by any suitable protocol for the network exchange of information. Frames mirrored from the network devices to the identification engine 186 may also be mirrored directly to the management engine 184 for transmission to the control manager 125 or network logging or to other functions and devices. The mirrored frames may also be mirrored to one or more other devices of the network infrastructure 101.

The application identification engine 186 is further represented in FIG. 3. The application identification engine 186

protocol for the network exchange of information. Frames mirrored from the network devices to the identification engine 186 may also be mirrored directly to the management engine 184 for transmission to the control manager 125 or **network logging** or to other functions and devices. The mirrored frames may also be mirrored to one or more other devices of the network infrastructure 101.

TCP/UDP external
port values

DNS, HTTP, SSL,
SMTP, etc.

Applications on non-
standard ports, and
apps that dynamically
select port values (P2P)

CISCO EXHIBIT 1009
Kjendal

lay of the physical network, and as known to those ordinarily skilled in the art, may include any type of VLAN, such as a port-based VLAN, MAC-based VLAN, a protocol-based VLAN, or an ATM VLAN. In a VLAN environment, remotely mirrored network traffic may have a specific IEEE 802.1Q VLAN tag used by device 400 to help direct the mirrored network traffic to a specified location and to isolate it from other traffic. Utilizing the VLAN tag, the frame relay logic of the device 400 typically used for forwarding frames is enhanced. Network traffic mirrored in a VLAN environment may traverse or pass through many network system devices before reaching the mirror destination point 408. Regardless of the particular mechanism for transferring mirrored traffic, the dynamic mirroring function 200 is arranged to configure one or more of the portals 412 to mirror one or more packets, including portions of packets, based on selected criteria. The configuration has several dynamic aspects to it and there are a plurality of criteria used in making

field or selected fields in the packet. The selected field or fields may be one or more of: a) address fields; b) protocol fields; c) length or byte count fields; and d) fields used in determining the value, meaning, placement or inclusion of other fields in the packet. The second selection criterion may also be a selection of the fields in the packet to be mirrored, byte count of a first selected portion of the packet to be mirrored and/or offset of a first selected portion of the packet to be mirrored. Further, the third selection criterion may be: a) a count setting of the frames meeting the first selection criterion; b) created using information contained in the packets mirrored or to be mirrored; c) created using information contained in packets not mirrored to the selected portal where mirroring occurs. The application layer of the packet or series of packets may also be used as either or both of the first and second criteria. The control engine 410 is configured to include one or more inputs used in the analysis of what and when to mirror traffic from the traffic source 404. Inner

mirroring occurs. The application layer of the packet or series of packets may also be used as either or both of the first and second criteria. The control engine 410 is configured to

Examples of such triggering events or conditions have been described above and are incorporated herein but the invention is not limited to those so listed.

The portals 412 may be configured by any means known to those skilled in the art, such as through the use of SNMP, to provide the parameters needed to determine or discover the portal destination, path to it and the transport criteria needed. As an example, a mirror may have been established by the policy-based mirroring function to mirror all traffic from a new attached function that has joined the network system 100. After a period of time based on one or more of policies, on the output of an IDS, the output of the appliance 180 or another network infrastructure device including the application identification function 200, and Network Access and Control (NAC) functions, the particular attached function might have the criteria for mirroring changed to reflect only mirroring, for example, the first 30 frames of a new flow to only the least loaded application identification function 200 of the network infrastructure 101. This might reflect an attached function now having a trusted, known, authenticated user, updated applications and virus detection status, and generally displaying good network behavior.

It is also to be noted that the control engine 410 configured to change the mirroring dynamically may be located in one or more other devices of the network infrastructure 101. Further, the control mechanism of the traffic mirror function may be automatically changed during the mirroring of traffic based on input of information to the control engine 410 that initiates a mirroring change based on network policy, mirroring policies or both. The first selection criterion may be based on a

200 including the engine 186 would need to be deployed throughout the network infrastructure 101 to insure a reasonable detection of a sufficient portion of the traffic flows important to the organization to permit reasonable control. The dynamic traffic mirroring function of the present invention dramatically eases the pain, expense and complexity of engine 186 deployment allowing such application identification engines to be placed almost at will by the network administrator, with the mirrors bringing to it the necessary traffic from any device of interest in the network infrastructure 101. The dynamic mirrors are most logically located in the switches and other devices that are primarily packet forwarding devices, such as switches and routers, as the incremental additional cost is relatively minor and heavily leverages the already existing capabilities these devices have: namely, to classify all incoming traffic, copy traffic as needed, rapidly modify frames and frame headers, filter and forward frames, and to support encapsulations, tunneling and routing technologies. Moreover, their positions in various network topologies make them almost ideal for the task.

Examples of network topology organizational options are shown in FIGS. 7-10. Components of the network that are numbered the same as in FIG. 1 remain the same for purposes of describing the different topology options. FIG. 7 shows a first network topology 700 in which transmissions from attached functions to the network infrastructure 101 enter through edge network devices and all Internet traffic is forced through firewall 118, which may be one or more firewalls scaled as desired based on the size of the network. The firewall 118 performs its gate keeping function and packets that

for that identification. For purposes of the present description, “network services” include, but are not limited to, access, **Quality of Service (QoS)**, bandwidth, priority, computer pro-

networks may be interconnected together to establish internetworks. For purposes of the description of the present invention, the devices and functions that establish the interconnection represent the network infrastructure. The users, computing devices and the like that use that network infrastructure to communicate are referred to herein as attached functions and will be further detailed. The network infrastructure and the attached functions and the network infrastructure will be referred to as a network system.

The process by which the various computing systems of a network or internetwork communicate is generally regulated by agreed-upon signal exchange standards and protocols embodied in network interface cards or circuitry or software, firmware and microcoded algorithms. Such standards and protocols were borne out of the need and desire to provide interoperability among the army of computing systems available from a plurality of suppliers. Two organizations that have been responsible for signal exchange standardization are the Institute of Electrical and Electronic Engineers (IEEE) and the Internet Engineering Task Force (IETF). In particular, the IEEE standards for inter-network interoperability have been established, or are in the process of being established, under the purview of the IEEE 802 committee on Local Area Networks (LANs) and Metropolitan Area Networks (MANs). The IEEE standards include many well defined methods of wired, fiber optic and Radio Frequency (RF or wireless) methods of network communication as well as the methods of network communication (IDS) or Firewall events, application access requests, priority change requests, protocol changes, the addition of a wireless access user, policy changes made, bandwidth changes, routing link changes, changes of monitored conditions, local and remote policy changes and network system changes. More generally for purposes of the description of the present invention, the establishment of the network infrastructure, event, activity, occurrence, information or characteristic identified in a network system by the network administrator as being of interest for the purpose of making a modification to an assigned set of policies. The types of triggers that define usage restrictions and network policy changes are managed by a network administrator. Network policies are generally directed to administration, management, and/or control of access to or usage of network services. A network policy may also be a policy abstraction that is the translation of one or more network policies to a different network environment. A network usage policy may be handled into a higher-level abstract network policy for ease of handling and management; a network policy set is simply a policy composed of one or more policies.

The network policies are typically defined in and regulated through a network policy server device of the network infrastructure controlled by the administrator. The established policies are transmitted to network interface devices of the network infrastructure, referred to herein as packet forwarding devices, for use in forwarding network traffic through the network infrastructure.

networks may be interconnected together to establish internetworks. For purposes of the description of the present invention, the devices and functions that establish the interconnection represent the network infrastructure. The users, computing devices and the like that use that network infrastructure to communicate are referred to herein as attached functions and will be further detailed. The network infrastructure and the attached functions and the network infrastructure will be referred to as a network system.

The process by which the various computing systems of a network or internetwork communicate is generally regulated by agreed-upon signal exchange standards and protocols embodied in network interface cards or circuitry or software, firmware and microcoded algorithms. Such standards and protocols were borne out of the need and desire to provide interoperability among the army of computing systems available from a plurality of suppliers. Two organizations that have been responsible for signal exchange standardization are the Institute of Electrical and Electronic Engineers (IEEE) and the Internet Engineering Task Force (IETF). In particular, the IEEE standards for inter-network interoperability have been established, or are in the process of being established, under the purview of the IEEE 802 committee on Local Area Networks (LANs) and Metropolitan Area Networks (MANs). The IEEE standards include many well defined methods of wired, fiber optic and Radio Frequency (RF or wireless) methods of network communication as well as the methods of network communication (IDS) or Firewall events, application access requests, priority change requests, protocol changes, the addition of a wireless access user, policy changes made, bandwidth changes, routing link changes, changes of monitored conditions, local and remote policy changes and network system changes. More generally for purposes of the description of the present invention, the establishment of the network infrastructure, event, activity, occurrence, information or characteristic identified in a network system by the network administrator as being of interest for the purpose of making a modification to an assigned set of policies. The types of triggers that define usage restrictions and network policy changes are managed by a network administrator. Network policies are generally directed to administration, management, and/or control of access to or usage of network services. A network policy may also be a policy abstraction that is the translation of one or more network policies to a different network environment. A network usage policy uses policies may be bundled into a higher-level abstract network policy for ease of handling and management; a network policy set is simply a policy composed of one or more policies.

The network policies are typically defined in and regulated through a network policy server device of the network infrastructure controlled by the administrator. The established policies are transmitted to network interface devices of the network infrastructure, referred to herein as packet forwarding devices, for use in forwarding data packets through the network.

networks may be interconnected together to establish internetworks. For purposes of the description of the present invention, the devices and functions that establish the interconnection represent the network infrastructure. The users, computing devices and the like that use that network infrastructure to communicate are referred to herein as attached functions and will be further detailed. The network infrastructure and the attached functions and the network infrastructure will be referred to as a network system.

The process by which the various computing systems of a network or internetwork communicate is generally regulated by agreed-upon signal exchange standards and protocols embodied in network interface cards or circuitry or software, firmware and microcoded algorithms. Such standards and protocols were borne out of the need and desire to provide interoperability among the army of computing systems available from a plurality of suppliers. Two organizations that have been responsible for signal exchange standardization are the Institute of Electrical and Electronic Engineers (IEEE) and the Internet Engineering Task Force (IETF). In particular, the IEEE standards for inter-network interoperability have been established, or are in the process of being established, under the purview of the IEEE 802 committee on Local Area Networks (LANs) and Metropolitan Area Networks (MANs). The IEEE standards include many well defined methods of wired, fiber optic and Radio Frequency (RF or wireless) methods of network communication as well as the methods of network communication (IDS) or Firewall events, application access requests, priority change requests, protocol changes, the addition of a wireless access user, policy changes made, bandwidth changes, routing link changes, changes of monitored conditions, local and remote policy changes and network system changes. More generally for purposes of the description of the present invention, the establishment of the network infrastructure, event, activity, occurrence, information or characteristic identified in a network system by the network administrator as being of interest for the purpose of making a modification to an assigned set of policies. The types of triggers that define usage restrictions and network policy changes are managed by a network administrator. Network policies are generally directed to administration, management, and/or control of access to or usage of network services. A network policy may also be a policy abstraction that is the translation of one or more network policies to a different network environment. A network usage policy uses policies may be bundled into a higher-level abstract network policy for ease of handling and management; a network policy set is simply a policy composed of one or more policies.

The network policies are typically defined in and regulated through a network policy server device of the network infrastructure controlled by the administrator. The established policies are transmitted to network interface devices of the network infrastructure, referred to herein as packet forwarding devices, for use in forwarding network data packets.

4	precise in my answer to understand what you are	4	information that could be contained in such a
5	meaning by an application-layer packet.	5	thing. So, for example, if we go to Column 7,
6	Q. Well, how do you understand it to	6	beginning around Line 49, it starts giving
7	mean -- what do you understand the claim to mean	7	examples of things that could be information
8	when it says, Application-layer packet-header?	8	that is in an application-layer packet as
9	A. So that's talking about the	9	defined by the '213 Patent.
10	application-layer packet-header information.	10	Q. Okay. And that refers to

3 Q. Okay. Is it possible such that
4 all of an application-layer HTTP packet, both
5 its headers and its payload would all be
6 embedded within one Layer-3 packet?

7 A. That may occur. That could
8 happen. I don't see a reason why, just sitting
9 here today, that would not be possible.

8 (Pages 26 to 29)

TSG Reporting - Worldwide 877-702-9580 Cisco Ex 1026
Goodwin Deposition August 29, 2019
Cisco v. Centripetal IPR2018-01512

Page 30

1 MICHAEL GOODRICH
2 A. I -- I -- I don't want to try to
3 speculate about variations without having data.

Page 31

1 MICHAEL GOODRICH
2 packet-header -- the application-layer packet-
3 header information that's associated with that

MICHAEL GOODRICH

Q. And you cite Pages 823 to 824 of Exhibit 1008, which is the ACNS guide, correct?

A. Correct.

Q. Where on those two pages does it use the term "reassemble"?

A. On those two pages, it does not use the word "reassemble." Instead, that would be understood by a person of ordinary skill in the art from the words that is used.

Q. But you agree it doesn't use the term "reassemble," correct?

A. Correct.