

QPrivacy USA LLC v. Cisco Systems, Inc., 2:24-cv-00855
Exhibit 7 – Invalidity Claim Chart for U.S. Patent Nos. 11,816,249 and 11,106,824

Cisco Systems, Inc. (“Cisco”) hereby contends that the asserted claims are invalid as anticipated by U.S. Patent No. 8,291,495 (“Yang”) prior art references under various subsections of 35 U.S.C. § 102 and/or as obvious under 35 U.S.C. § 103 in view of the prior art reference alone, combined with the knowledge of a person of ordinary skill in the art, and/or in combination with other references in Cisco’s Initial Invalidity Contentions. The chart below discloses how the prior art reference discloses, either expressly or inherently, and/or renders obvious each of the asserted claims.

The citations provided are exemplary and do not necessarily include each and every disclosure of the limitation in the references. Cisco has endeavored to cite to the most relevant portions of the identified prior art, but other portions of the identified prior art may additionally disclose, either expressly or inherently, and/or render obvious one or more limitations of the asserted claims. Thus, Cisco reserves the right to rely on: (1) uncited portions of the identified prior art; (2) other prior art not identified herein; (3) references that show the state of the art (irrespective of whether such references themselves qualify as prior art to the asserted patents); (4) factual testimony from the inventors or authors of the prior art references, or purveyors of prior art devices; and/or (5) expert testimony, to provide context to or aid in understanding the prior art and the state of the art at the time of the alleged inventions.

The lack of a citation for an element should not be deemed an admission that the element is not disclosed or is not inherent in the reference. When the chart indicates a particular reference discloses or embodies a limitation, the terms “discloses,” “disclosed,” “embodies,” and “embodied” refer to explicit and/or inherent disclosure and/or obvious variations of the actual disclosure. Further, to the extent Cisco asserts that a claim is indefinite, Cisco has used its best efforts to reasonably interpret the claims to fulfill their duties in charting the prior art references.

Where Cisco cites to a particular drawing or figure in the accompanying charts, the citation encompasses the description of the drawing or figure, as well as any text associated with the drawing or figure. Similarly, where Cisco cites to particular text concerning a drawing or figure, the citation encompasses that drawing or figure as well. Certain identified prior art inherently discloses features of the asserted claims. Cisco reserves the right to rely on inherency to demonstrate the invalidity of the asserted claims. Moreover, certain prior art references may inherently disclose certain features of the asserted claims as construed by Plaintiffs. Cisco may rely on cited or uncited portions of the prior art, other documents, factual testimony, and expert testimony to establish the inherency of certain features of the prior art to invalidate the asserted claims.

As set forth in Cisco’s Initial Invalidity Contentions, the below contentions apply the prior art in part in accordance with Cisco’s assumption that QPrivacy contends the claims are not invalid under 35 U.S.C. § 112. However, Cisco’s below contentions do not represent Cisco’s agreement or view as to the meaning, definiteness, written description support for, or enablement of any claim of the ’249 or ’824 patents, or that the ’249 or ’824 patents properly disclose structures corresponding to functions in any claims

***QPrivacy USA LLC v. Cisco Systems Inc.*, 2:24-cv-00855**
Exhibit 7 – Invalidity Claim Chart for U.S. Patent Nos. 11,816,249 and 11,106,824

governed by 35 U.S.C. § 112 ¶ 6. For each dependent claim, the disclosures cited for the claim from which it depends are incorporated by reference.

U.S. Patent No. 11,816,249		
Claim	Relevant Disclosure of Yang	
1. A method of dynamic management of encrypted data during communication between a remote server and a user's device, the method comprising:	To the extent the preamble is limiting, Yang discloses this element and/or renders it obvious either alone or in combination with other references and/or the knowledge of one of ordinary skill in the art. For example, Yang discloses dynamically managing private network communications between an intrusion detection system (IDS) and a user device (<i>e.g.</i> , a “node,” a security management device). The communications may include encrypted communications.	

QPrivacy USA LLC v. Cisco Systems Inc., 2:24-cv-00855
Exhibit 7 – Invalidity Claim Chart for U.S. Patent Nos. 11,816,249 and 11,106,824

U.S. Patent No. 11,816,249	
Claim	Relevant Disclosure of Yang
	The diagram illustrates a network architecture. A cloud labeled 'PUBLIC NETWORK 6' is connected to a bracketed section labeled 'ENTERPRISE'. Within the enterprise, a 'FIREWALL 9' is connected to an 'IDS 10', which is connected to a 'SWITCH 19'. The switch is connected to a group of nodes labeled '18', which includes 'NODE 8A', three dots, 'NODE 8N', and 'SECURITY MANAGEMENT DEVICE 14'. An 'IDS 14' is also connected to the switch. A reference numeral '2' points to the entire enterprise system. FIG. 1 (Yang, Fig. 1.)

U.S. Patent No. 11,816,249	
Claim	Relevant Disclosure of Yang
	<pre> graph TD 70[70 RECEIVE CONFIGURATION INFORMATION] --> 72[72 MONITOR NETWORK TRAFFIC] 72 --> 74[74 IDENTIFY PACKET FLOWS AND UPDATE FLOW TABLE] 74 --> 80[80 REASSEMBLE PACKETS TO FORM APPLICATION-LAYER COMMUNICATIONS] 80 --> 81[81 IDENTIFY APPLICATION] 81 --> 82[82 INVOKE PROTOCOL DECODERS] 82 --> 84[84 OUTPUT TRANSACTION DATA, APPLICATION-LAYER ELEMENTS AND PROTOCOL ANOMALIES] 84 --> 88[88 SELECT ATTACK DEFINITIONS] 88 --> 90[90 APPLY ATTACK DEFINITIONS] 90 --> 91{91 MATCH?} 91 -- YES --> 92[92 OUTPUT ALERT AND TAKE PROGRAMMED RESPONSE] 91 -- NO --> 94[94 FORWARD PACKETS AND RE-CHECK LATER PACKETS] 92 --> END([END]) 94 --> END </pre> FIG. 4 (Yang, Fig. 4.)

QPrivacy USA LLC v. Cisco Systems Inc., 2:24-cv-00855
Exhibit 7 – Invalidity Claim Chart for U.S. Patent Nos. 11,816,249 and 11,106,824

U.S. Patent No. 11,816,249	
Claim	Relevant Disclosure of Yang
	“In general, the invention is directed to techniques for detecting and preventing network attacks, such as Denial of Service Attacks (“DOS”) attacks, network viruses or other malicious activity. More specifically, improved techniques are described herein for identifying the software application and protocol associated with a data stream processed by an intrusion detection system (“IDS”). For example, as described herein, an IDS may analyze both client-to-server and server-to-client packet flows in order to improve the accuracy of the identification. Upon first receiving a first packet flow, for example, a connection request from a client device, the IDS may make an initial assessment as to the type of software application and protocol for the incoming data stream. Upon receiving a response, for example, from a server, the IDS may then attempt to confirm the initial assessment of the application and protocol. If the initial assessment was incorrect, then the IDS may dynamically reclassify the application and protocol associated with the data stream. The IDS may then apply the proper pattern to the data stream based on the reclassification.” (Yang, 1:46-65.) “In another embodiment, the invention is directed to an intrusion detection system (“IDS”) device comprising a flow analysis module to receive a first packet flow from a client and a second packet flow from a server in response to the first packet flow, a forwarding component to send the first packet flow to the server and the second packet flow to the client, and a stateful inspection engine to apply one or more sets of patterns to the first packet flow to determine whether the first packet flow represents a network attack. The IDS further comprises an application identification module to perform an initial identification of a type of software application and communication protocol associated with the first packet flow, and to reevaluate the identification of the type of software application and protocol according to the second packet flow.” (Yang, 2:44-57.) “FIG. 1 is a block diagram illustrating an exemplary system 2 in which enterprise computer network 4 includes intrusion detection system (“IDS”) 10 that identifies applications and protocols in accordance with the principles of the invention. In the example embodiment of FIG. 1, IDS 10 is a single network device. Network 4 also

QPrivacy USA LLC v. Cisco Systems Inc., 2:24-cv-00855
Exhibit 7 – Invalidity Claim Chart for U.S. Patent Nos. 11,816,249 and 11,106,824

U.S. Patent No. 11,816,249	
Claim	Relevant Disclosure of Yang
	includes a private enterprise computing network 5 that is coupled to public network 6, such as the Internet. Public network 6 may include, for example, one or more client computing devices. Firewall 9 protects enterprise network 5 and, in particular, internal computing nodes 8A-8N. Computing nodes 8A-8N (“computing nodes 8”) represent any private computing device within enterprise network 5, including workstations, file servers, print servers, database servers, printers and other devices. In the example of FIG. 1, enterprise network 5 includes IDS 10 that monitors traffic flowing between firewall 9 and internal computing nodes 8. As described herein, IDS 10 may analyze network traffic flowing in both directions (i.e., inbound traffic received from public network 6 as well as outbound traffic destined to the public network) to improve the accuracy in detecting network attacks. That is, IDS 10 may analyze both client-to-server and server-to-client communications between public network 6 and computing nodes 8. IDS 10 may analyze the network traffic to correlate traffic in one direction with traffic in the opposite direction for each communication session detected within the network traffic. For example, for each client-server communication session, IDS may identify a packet flow in one direction (e.g., a client-to-server communication flow for a particular software application on the client) and a corresponding packet flow in the opposite direction (e.g., response communications flowing from the server to the client for that same software application). In accordance with the principles of the invention, IDS 10 integrates pattern matching with application- and protocol-specific anomaly detection to identify sophisticated attack behaviors. In one embodiment, IDS 10 allows the system administrator to specify attack definitions. In one embodiment, the system administrator may specify compound attack definitions. Further details on application of attack definitions, e.g. compound attack definitions, may be found within U.S. patent application Ser. No. 11/045,572, Guruswamy et al., “Compound Attack Detection in a Computer Network,” filed Jan. 27, 2005, assigned to the assignee of the current application, which is incorporated herein by reference in its entirety.” (Yang, 4:12-58.)

QPrivacy USA LLC v. Cisco Systems Inc., 2:24-cv-00855
Exhibit 7 – Invalidity Claim Chart for U.S. Patent Nos. 11,816,249 and 11,106,824

U.S. Patent No. 11,816,249	
Claim	Relevant Disclosure of Yang
	“IDS 10 applies the attack definitions to the elements and the protocol-specific anomalies identified by the protocol decoders to detect and prevent network attacks. For example, a system administrator may specify a compound network attack that includes the protocol anomaly of repeated FTP login failure and a pattern that matches a login username of “root.” In this manner, the system administrator may combine pattern analysis with protocol anomalies to define complex attack definitions. In the event of a network attack, IDS 10 may take one or more programmed actions, such as automatically dropping packet flows associated with the application-layer communications within which the network attack was detected. In accordance with the techniques described herein, IDS 10 may attempt to identify the software application and protocol associated with each communication session, i.e., each corresponding pair of packet flows between a given client and server. IDS 10 uses the identified type of software application and underlying protocol type for the communication session to control selection of any protocol-specific patterns to be applied to the packet flows. IDS 10 may temporarily cache the initial assessment of the software application for later comparison and confirmation.” (Yang, 5:53-6:8.) “These techniques may allow IDS 10 to provide improved attack detection and prevention in environments where some applications no longer use a static port mapping but instead use dynamic or randomized port assignments in an attempt to evade detection and containment. For example, instant messaging programs may initiate communication on one port, but then agree to communicate thereafter on a distinct port. Other programs, such as peer-to-peer (“P2P”) programs and hacker toolkits such as Metasploit, may determine ports dynamically as well. As described, IDS 10 may make an initial determination for the application type of a packet flow from a client device of public network 6 to a server device, e.g. node 8A, based upon characteristics of the communication in addition to specified port number. IDS 10 may then analyze the return communication from node 8A to the client device to verify the initial assumption or to modify the initial determination if necessary.

QPrivacy USA LLC v. Cisco Systems Inc., 2:24-cv-00855
Exhibit 7 – Invalidity Claim Chart for U.S. Patent Nos. 11,816,249 and 11,106,824

U.S. Patent No. 11,816,249	
Claim	Relevant Disclosure of Yang
	In some embodiments, enterprise network 5 may include multiple IDSs 10 and 14 located within different regions (e.g., sub-networks) of enterprise network 5. Security management device 18 may operate as a central device for managing IDSs 10 and 14. Although the example illustrated in FIG. 1 is described in terms of dedicated IDSs 10 and 14, the functionality described herein may be incorporated within other devices, such as firewall 9 or switch 19.” (Yang, 6:45-7:2.) “FIG. 2 is a block diagram illustrating an example embodiment of an IDS 20. In the illustrated example, IDS 20 includes a forwarding plane 22 that transparently monitors inbound network traffic 24 and forwards the network traffic as outbound network traffic 26. In the example illustrated by FIG. 2, forwarding plane 22 includes flow analysis module 25, stateful inspection engine 28, protocol decoders 30, forwarding component 31 and security management module 44. Security management module 44 presents a user interface by which administrator 42 configures IDS 20. For example, administrator 42 may configure IDS 20 to monitor particular subnets of the enterprise network. In addition, security management module 44 presents a user interface by which administrator 42 may specify attack definitions 33, which security management module 44 relays to stateful inspection engine 28. In one embodiment, attack definitions 33 may be compound attack definitions. Moreover, security management module 44 may present a user interface by which administrator 42 may modify assumptions regarding packet flow characteristics, such as the highest priority packet flows for monitoring, port bindings for applications, or other features of determining a type of application and protocol associated with the packet flow. Flow analysis module 25 receives inbound traffic 24 and identifies network flows within the traffic. Each network flow represents a flow of packets in one direction within the network traffic and is identified by at least a source address, a destination address and a

QPrivacy USA LLC v. Cisco Systems Inc., 2:24-cv-00855
Exhibit 7 – Invalidity Claim Chart for U.S. Patent Nos. 11,816,249 and 11,106,824

U.S. Patent No. 11,816,249	
Claim	Relevant Disclosure of Yang
	communication protocol. Flow analysis module 25 may utilize additional information to specify network flows, including source media access control (“MAC”) address, destination MAC address, source port, and destination port. Other embodiments may use other information to identify network flows, such as IP addresses. Flow analysis module 25 maintains data within flow table 35 that describes each active packet flow present within the network traffic. Flow table 35 specifies network elements associated with each active packet flow, i.e., low-level information such as source and destination devices and ports associated with the packet flow. In addition, flow table 35 may identify pairs of packet flows that collectively form a single communication session between a client and server. For example, flow table 35 may designate communication session as pairs of packet flows in opposite directions for flows sharing at least some common network addresses, ports and protocol.” (Yang, 7:16-60.)
1[a]. receiving, by the remote server, a communication comprising encrypted data packets;	Yang discloses this element and/or renders it obvious either alone or in combination with other references and/or the knowledge of one of ordinary skill in the art. For example, Yang’s IDS is configured to receive packets comprising private communications, which may be encrypted.

U.S. Patent No. 11,816,249	
Claim	Relevant Disclosure of Yang
	The diagram illustrates an Intrusion Detection System (20) divided into two main sections by a dashed line labeled 'FORWARDING PLANE' (22). Above the plane, an 'ADMINISTRATOR' (42) is connected to a 'SECURITY MANAGEMENT MODULE' (44). Below the plane, the system includes a 'FLOW ANALYSIS MODULE' (25) which is connected to a 'FLOW TABLE' (35). The 'FLOW ANALYSIS MODULE' (25) sends data (32) to a 'STATEFUL INSPECTION ENGINE' (28). The 'STATEFUL INSPECTION ENGINE' (28) receives data (34) from 'PROTOCOL DECODERS' (30) and sends data (36) to them. The 'STATEFUL INSPECTION ENGINE' (28) also sends data (38) to a 'FORWARDING COMPONENT' (31). The 'FORWARDING COMPONENT' (31) handles 'OUTBOUND TRAFFIC' (26). 'INBOUND TRAFFIC' (24) enters the system from the left. The 'SECURITY MANAGEMENT MODULE' (44) sends data (33) to the 'STATEFUL INSPECTION ENGINE' (28) and receives data (40) from it. FIG. 2 (Yang, Fig. 2.)

U.S. Patent No. 11,816,249	
Claim	Relevant Disclosure of Yang
	<pre> graph TD 70[70 RECEIVE CONFIGURATION INFORMATION] --> 72[72 MONITOR NETWORK TRAFFIC] 72 --> 74[74 IDENTIFY PACKET FLOWS AND UPDATE FLOW TABLE] 74 --> 80[80 REASSEMBLE PACKETS TO FORM APPLICATION-LAYER COMMUNICATIONS] 80 --> 81[81 IDENTIFY APPLICATION] 81 --> 82[82 INVOKE PROTOCOL DECODERS] 82 --> 84[84 OUTPUT TRANSACTION DATA, APPLICATION-LAYER ELEMENTS AND PROTOCOL ANOMALIES] 84 --> 88[88 SELECT ATTACK DEFINITIONS] 88 --> 90[90 APPLY ATTACK DEFINITIONS] 90 --> 91{91 MATCH?} 91 -- YES --> 92[92 OUTPUT ALERT AND TAKE PROGRAMMED RESPONSE] 91 -- NO --> 94[94 FORWARD PACKETS AND RE-CHECK LATER PACKETS] 92 --> END([END]) 94 --> END </pre> FIG. 4 (Yang, Fig. 4.)

QPrivacy USA LLC v. Cisco Systems Inc., 2:24-cv-00855
Exhibit 7 – Invalidity Claim Chart for U.S. Patent Nos. 11,816,249 and 11,106,824

U.S. Patent No. 11,816,249	
Claim	Relevant Disclosure of Yang
1[b]. determining, by the remote server, a content of at least one data packet of the communication in accordance with characteristics of the at least one data packet, and wherein the content of the at least one data packet is not decrypted by the remote server, and the determination of the content is performed by the remote server in real time during a communication session between the remote server and the user's device;	See also 1[pre]. Yang discloses this element and/or renders it obvious either alone or in combination with other references and/or the knowledge of one of ordinary skill in the art. For example, Yang discloses identifying an application or protocol based on a TCP port in a packet header without decrypting the packet. Yang's IDS monitors traffic in real-time. "In general, IDS 10 identifies packet flows in the monitored traffic, and transparently reassembles application-layer communications from the packet flows. A set of protocol-specific decoders within the IDS 10 analyzes the application-layer communications and identifies application-layer transactions. In general, a "transaction" refers to a bounded series of related application-layer communications between peer devices. For example, a single TCP connection can be used to send (receive) multiple HyperText Transfer Protocol (HTTP) requests (responses). As one example, a single web-page comprising multiple images and links to HTML pages may be fetched using a single TCP connection. An HTTP decoder identifies each request/response within the TCP connection as a different transaction. This may be useful to prevent certain attack definitions from being applied across transaction boundaries. In one embodiment, a transaction may be identified according to source and destination IP address, protocol, and source and destination port numbers. Other embodiments may identify a transaction in other ways, for example, by using media access control ("MAC") addresses." (Yang, 5:1-20.) "In addition, IDS 10 monitors for traffic flowing in the opposite direction to intercept any response from the server. That is, IDS 10 identifies any response packet flow from the server, which typically utilizes the same or substantially similar network addresses, ports and/or protocol as the initial client request. Upon intercepting the response packet flow, IDS 10 performs an additional application identification test to confirm whether its initial assessment of the type of software application and protocol for the particular client-server communication session was correct. If not, IDS 10 may reclassify the communication as a different type of software application and protocol, and select and apply additional attack

QPrivacy USA LLC v. Cisco Systems Inc., 2:24-cv-00855
Exhibit 7 – Invalidity Claim Chart for U.S. Patent Nos. 11,816,249 and 11,106,824

U.S. Patent No. 11,816,249	
Claim	Relevant Disclosure of Yang
	definitions based on the updated classification. That is, IDS 10 may retrieve the cached assessment of the application and then use the response packet flow to either confirm or modify the assessment, again caching the result.” (Yang, 6:22-37.) “These techniques may allow IDS 10 to provide improved attack detection and prevention in environments where some applications no longer use a static port mapping but instead use dynamic or randomized port assignments in an attempt to evade detection and containment. For example, instant messaging programs may initiate communication on one port, but then agree to communicate thereafter on a distinct port. Other programs, such as peer-to-peer (“P2P”) programs and hacker toolkits such as Metasploit, may determine ports dynamically as well. As described, IDS 10 may make an initial determination for the application type of a packet flow from a client device of public network 6 to a server device, e.g. node 8A, based upon characteristics of the communication in addition to specified port number. IDS 10 may then analyze the return communication from node 8A to the client device to verify the initial assumption or to modify the initial determination if necessary. In some embodiments, enterprise network 5 may include multiple IDSs 10 and 14 located within different regions (e.g., sub-networks) of enterprise network 5. Security management device 18 may operate as a central device for managing IDSs 10 and 14. Although the example illustrated in FIG. 1 is described in terms of dedicated IDSs 10 and 14, the functionality described herein may be incorporated within other devices, such as firewall 9 or switch 19.” (Yang, 6:45-7:2.) “FIG. 2 is a block diagram illustrating an example embodiment of an IDS 20. In the illustrated example, IDS 20 includes a forwarding plane 22 that transparently monitors inbound network traffic 24 and forwards the network traffic as outbound network traffic 26. In the example illustrated by FIG. 2, forwarding plane 22 includes flow analysis module 25, stateful inspection engine 28, protocol decoders 30, forwarding component 31 and security management module 44.

QPrivacy USA LLC v. Cisco Systems Inc., 2:24-cv-00855
Exhibit 7 – Invalidity Claim Chart for U.S. Patent Nos. 11,816,249 and 11,106,824

U.S. Patent No. 11,816,249	
Claim	Relevant Disclosure of Yang
	Security management module 44 presents a user interface by which administrator 42 configures IDS 20. For example, administrator 42 may configure IDS 20 to monitor particular subnets of the enterprise network. In addition, security management module 44 presents a user interface by which administrator 42 may specify attack definitions 33, which security management module 44 relays to stateful inspection engine 28. In one embodiment, attack definitions 33 may be compound attack definitions. Moreover, security management module 44 may present a user interface by which administrator 42 may modify assumptions regarding packet flow characteristics, such as the highest priority packet flows for monitoring, port bindings for applications, or other features of determining a type of application and protocol associated with the packet flow. Flow analysis module 25 receives inbound traffic 24 and identifies network flows within the traffic. Each network flow represents a flow of packets in one direction within the network traffic and is identified by at least a source address, a destination address and a communication protocol. Flow analysis module 25 may utilize additional information to specify network flows, including source media access control (“MAC”) address, destination MAC address, source port, and destination port. Other embodiments may use other information to identify network flows, such as IP addresses. Flow analysis module 25 maintains data within flow table 35 that describes each active packet flow present within the network traffic. Flow table 35 specifies network elements associated with each active packet flow, i.e., low-level information such as source and destination devices and ports associated with the packet flow. In addition, flow table 35 may identify pairs of packet flows that collectively form a single communication session between a client and server. For example, flow table 35 may designate communication session as pairs of packet flows in opposite directions for flows sharing at least some common network addresses, ports and protocol.” (Yang, 7:16-60.)

QPrivacy USA LLC v. Cisco Systems Inc., 2:24-cv-00855
Exhibit 7 – Invalidity Claim Chart for U.S. Patent Nos. 11,816,249 and 11,106,824

U.S. Patent No. 11,816,249	
Claim	Relevant Disclosure of Yang
	“IDS 20 may use a minimum data size of the reassembled TCP segments, in addition to the signature, in order to identify the types of applications. Certain applications may require a minimum amount of data, so IDS 20 may distinguish malicious packet flows by determining whether the packet flow contains enough data for the identified protocol. Moreover, IDS 20 may not necessarily recognize every application. In one embodiment, when an application is unknown, IDS 20 may simply forward the packet flow. If IDS 20 cannot identify a given application, it may be because that application is not a typical target for a malicious packet flow. Other embodiments may take other actions for unidentified applications, however, such as discarding all packets which target unknown applications or applying a default signature to all packet flows associated with unknown application types. Other embodiments may also utilize other protocols, such as the user datagram protocol (UDP); IDS 20 accordingly may require a minimum data size of UDP segments in order to identify the application associated with the UDP segments. For each packet flow, stateful inspection engine 28 buffers a copy of the packet flow and reassembles the buffered packet flow to form application-layer communications 32. For example, stateful inspection engine 28 may reconstruct TCP segments into application-layer communications 32, which represent protocol-specific messages. Stateful inspection engine 28 invokes the appropriate one of protocol decoders 30 based on the identified type of application determination to analyze the application-layer communications 32. Protocol decoders 30 represent a set of one or more protocol-specific software modules. Each of protocol decoders 30 corresponds to a different communication protocol or service. Examples of communication protocols that may be supported by protocol decoders 30 include the HyperText Transfer Protocol (“HTTP”), the File Transfer Protocol (“FTP”), the Network News Transfer Protocol (“NNTP”), the Simple Mail Transfer Protocol (“SMTP”), Telnet, Domain Name System (“DNS”), Gopher, Finger, the Post Office Protocol (“POP”), the Secure Socket Layer (“SSL”) protocol, the Lightweight Directory Access Protocol (“LDAP”), Secure Shell (“SSH”), Server Message Block (“SMB”) and other protocols.” (Yang, 8:8-48.)

QPrivacy USA LLC v. Cisco Systems Inc., 2:24-cv-00855
Exhibit 7 – Invalidity Claim Chart for U.S. Patent Nos. 11,816,249 and 11,106,824

U.S. Patent No. 11,816,249	
Claim	Relevant Disclosure of Yang
	“Application identification module 51 identifies the type of application and underlying protocol for each intercepted communication session. When stateful inspection engine 28 receives a packet as part of a packet flow, reassembly module 50 buffers the packet in data buffer 55. In one embodiment, data buffer 55 may store data as a sliding window. That is, data buffer 55 may store data until becoming full or reaching a specified required amount of minimum data for identification. When full, data buffer 55 discards certain data to make room for storing new data. In one embodiment, data buffer 55 may store and discard data according to a first-in, first-out (“FIFO”)-like protocol wherein the first data to be stored is the first data to be discarded when data buffer 55 becomes full. In another embodiment, data buffer 55 may discard data according to a least recently used protocol wherein, when data buffer 55 is full, the packet flow which has been least recently used will be discarded to make room for new data to be stored. In one embodiment, reassembly module 50 may associate packets in a packet flow, and packet flows as a communication session, according to the 5-tuple {source IP address, destination IP address, protocol, source port, destination port}. Other embodiments may use other forms of associating packets. For example, in one embodiment, IDS 20 may be part of a network that utilizes virtual local area networks (VLANs). Accordingly, reassembly module 50 may associate packets in a packet flow according to a VLAN identifier, a source address, and a destination address. In any case, reassembly module 50 may utilize the information maintained within flow table 35 (FIG. 2) to reassemble network data, e.g., to form reassembled TCP data. Application identification module 51 analyzes the reassembled data for the packet flow to identify the type of application and protocol associated with the packet flow. If application identification module 51 is unable to identify the type of application and protocol associated with the packet flow, application identification module 51 may use the well-known static port binding as a default application selection. Table 1, below, shows an example static port binding list. Other embodiments may use more, fewer, or different

***QPrivacy USA LLC v. Cisco Systems Inc.*, 2:24-cv-00855**
Exhibit 7 – Invalidity Claim Chart for U.S. Patent Nos. 11,816,249 and 11,106,824

U.S. Patent No. 11,816,249																																									
Claim	Relevant Disclosure of Yang																																								
	entries in a static port table. Moreover, administrator 42 may configure the static port mapping using security management module 44.” (Yang, 9:27-10:3.) TABLE I <table> <tr> <th>PORT</th><th>APPLICATION</th></tr> <tr><td>20</td><td>FTP</td></tr> <tr><td>22</td><td>SSH</td></tr> <tr><td>23</td><td>Telnet</td></tr> <tr><td>25</td><td>SMTP</td></tr> <tr><td>43</td><td>WHOIS</td></tr> <tr><td>53</td><td>DNS</td></tr> <tr><td>67</td><td>BOOTP or DHCP</td></tr> <tr><td>70</td><td>Gopher</td></tr> <tr><td>79</td><td>Finger</td></tr> <tr><td>80</td><td>HTTP</td></tr> <tr><td>109</td><td>POP</td></tr> <tr><td>110</td><td>POP3</td></tr> <tr><td>113</td><td>ident/IRC</td></tr> <tr><td>118</td><td>SQL</td></tr> <tr><td>119</td><td>NNTP</td></tr> <tr><td>194</td><td>IRC</td></tr> <tr><td>443</td><td>HTTPS</td></tr> <tr><td>445</td><td>SMB</td></tr> <tr><td>564</td><td>RTSP</td></tr> </table> (Yang, Table 1.) “In the case where application identification module 51 is unable to identify a type of application and protocol for a packet flow, application identification module 51 may use the default static port mapping to determine an application, causing IDS 20 to respond accordingly. In some cases, application identification module 51 may not be able to	PORT	APPLICATION	20	FTP	22	SSH	23	Telnet	25	SMTP	43	WHOIS	53	DNS	67	BOOTP or DHCP	70	Gopher	79	Finger	80	HTTP	109	POP	110	POP3	113	ident/IRC	118	SQL	119	NNTP	194	IRC	443	HTTPS	445	SMB	564	RTSP
PORT	APPLICATION																																								
20	FTP																																								
22	SSH																																								
23	Telnet																																								
25	SMTP																																								
43	WHOIS																																								
53	DNS																																								
67	BOOTP or DHCP																																								
70	Gopher																																								
79	Finger																																								
80	HTTP																																								
109	POP																																								
110	POP3																																								
113	ident/IRC																																								
118	SQL																																								
119	NNTP																																								
194	IRC																																								
443	HTTPS																																								
445	SMB																																								
564	RTSP																																								

QPrivacy USA LLC v. Cisco Systems Inc., 2:24-cv-00855
Exhibit 7 – Invalidity Claim Chart for U.S. Patent Nos. 11,816,249 and 11,106,824

U.S. Patent No. 11,816,249	
Claim	Relevant Disclosure of Yang
	identify the application and the static port mapping may not have an entry for the requested port number. Various embodiments may treat this situation according to specifications of, for example, a system administrator. For example, in one embodiment, IDS 20 simply forwards packet flows with undetermined application types and protocols that cannot be determined by the static port mapping pass, as an unknown application may indicate that the packet flow is not targeting any type of application known to pose a security threat. In other embodiments, IDS 20 may automatically discard packet flows with unknown application types and protocols that cannot be determined by the static port mapping.” (Yang, 10:25-42.) “Once configured, IDS 20 monitors network traffic 24 (72). In some configurations, stateful inspection engine 28 of forwarding plane 22 may receive network traffic and mirror the network traffic for purposes of analysis. Forwarding component 31 seamlessly forwards the original network traffic. In other embodiments, traffic is not mirrored, rather a line-rate buffering approach is used to analyze the traffic in real-time prior to forwarding.” (Yang, 11:63-12:3.) “4. The method of claim 2, wherein the compared properties comprise at least one of a packet flow pattern, a destination port number, a source port number, and a protocol.” (Yang, 17:3-5.)
1[c]. storing, by the remote server, a preference list;	Yang discloses this element and/or renders it obvious either alone or in combination with other references and/or the knowledge of one of ordinary skill in the art. For example, Yang discloses attack definitions to detect and prevent network attacks and other security risks. These definitions may be stored in the stateful inspection engine. These definitions may be edited.

U.S. Patent No. 11,816,249	
Claim	Relevant Disclosure of Yang
	FIG. 3 (Yang, Fig. 3.)

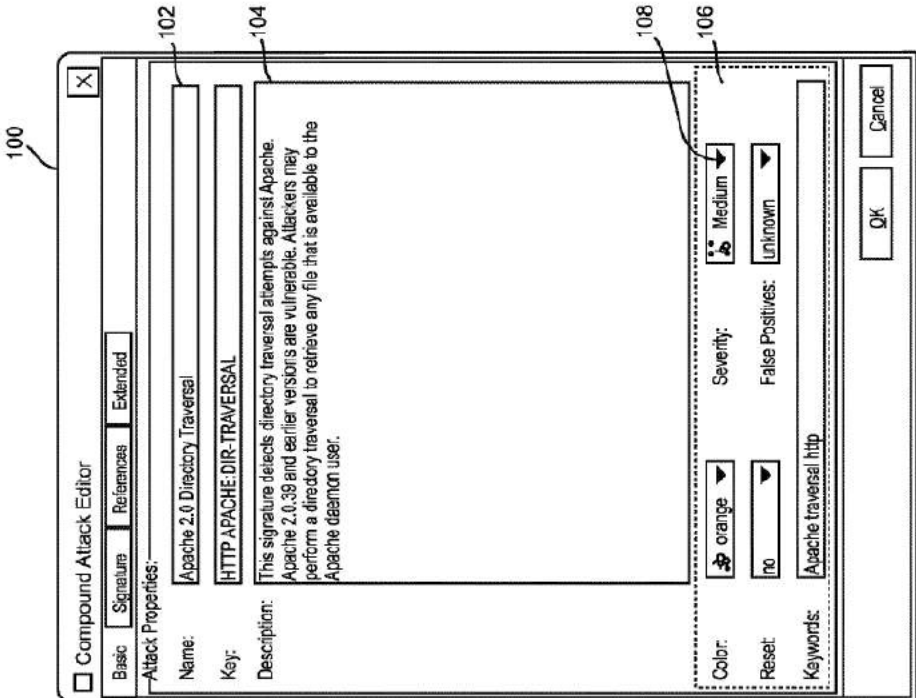
U.S. Patent No. 11,816,249	
Claim	Relevant Disclosure of Yang
	 The screenshot shows a window titled "Compound Attack Editor" with a close button (X) in the top right corner. The window has a tabbed interface with "Basic", "Signature", "References", and "Extended" tabs. The "Basic" tab is selected, displaying "Attack Properties". Fields within the "Attack Properties" section include: Name: Apache 2.0 Directory Traversal (labeled 102) Key: HTTP:APACHE:DIR-TRAVERSAL (labeled 104) Description: This signature detects directory traversal attempts against Apache. Apache 2.0,38 and earlier versions are vulnerable. Attackers may perform a directory traversal to retrieve any file that is available to the Apache daemon user. Color: orange (with a color selection icon) Severity: Medium (with a dropdown arrow) Reset: no (with a dropdown arrow) False Positives: unknown (with a dropdown arrow) Keywords: Apache traversal http At the bottom right of the window are "OK" and "Cancel" buttons. A dashed box labeled 106 encloses the "Color", "Severity", "Reset", "False Positives", and "Keywords" fields. A label 108 points to the "Severity" dropdown.

FIG. 6

(Yang, Fig. 6.)

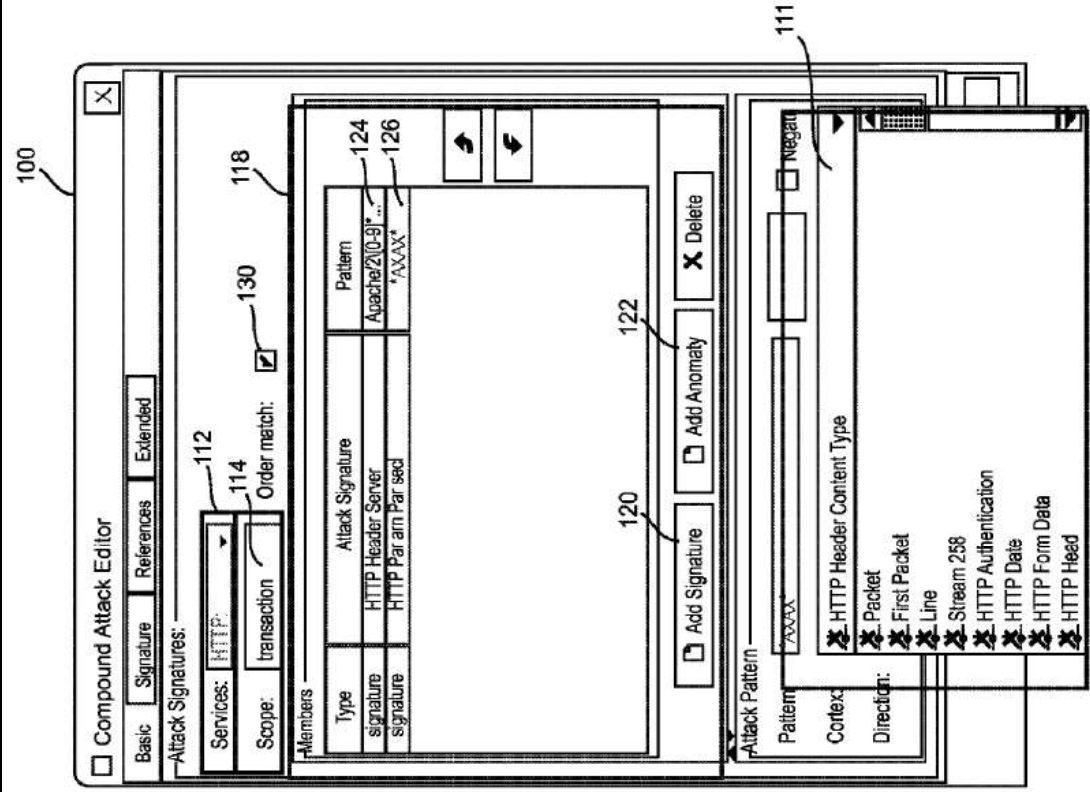


FIG. 7

(Yang, Fig. 7.)

QPrivacy USA LLC v. Cisco Systems Inc., 2:24-cv-00855
Exhibit 7 – Invalidity Claim Chart for U.S. Patent Nos. 11,816,249 and 11,106,824

U.S. Patent No. 11,816,249	
Claim	Relevant Disclosure of Yang
	“In accordance with the principles of the invention, IDS 10 integrates pattern matching with application- and protocol-specific anomaly detection to identify sophisticated attack behaviors. In one embodiment, IDS 10 allows the system administrator to specify attack definitions. In one embodiment, the system administrator may specify compound attack definitions. Further details on application of attack definitions, e.g. compound attack definitions, may be found within U.S. patent application Ser. No. 11/045,572, Guruswamy et al., “Compound Attack Detection in a Computer Network,” filed Jan. 27, 2005, assigned to the assignee of the current application, which is incorporated herein by reference in its entirety. In general, the attack definitions may specify, for example, any combination of textual and non-textual (e.g., binary) patterns and protocol anomalies to define complex attack signatures. Moreover, IDS 10 may associate particular signatures with protocols of certain applications. For a given communication session intercepted by IDS 10, the IDS attempts to identify the application type and underlying protocol for the packet flows of the session in order to select one or more attack signatures to apply to the packet flows.” (Yang, 4:52-67.) “IDS 10 applies the attack definitions to the elements and the protocol-specific anomalies identified by the protocol decoders to detect and prevent network attacks. For example, a system administrator may specify a compound network attack that includes the protocol anomaly of repeated FTP login failure and a pattern that matches a login username of “root.” In this manner, the system administrator may combine pattern analysis with protocol anomalies to define complex attack definitions. In the event of a network attack, IDS 10 may take one or more programmed actions, such as automatically dropping packet flows associated with the application-layer communications within which the network attack was detected.” (Yang, 5:53-65.) “For example, upon first receiving a connection request from a client and destined for a server, IDS 10 may use a static port number list to make an initial assessment of the type

QPrivacy USA LLC v. Cisco Systems Inc., 2:24-cv-00855
Exhibit 7 – Invalidity Claim Chart for U.S. Patent Nos. 11,816,249 and 11,106,824

U.S. Patent No. 11,816,249	
Claim	Relevant Disclosure of Yang
	of software application and protocol associated with the received packet flow. As one example, HTTP packets are typically directed to port 80, while e-mail packets have been directed to port 25. Therefore, IDS 10 may utilize the static port assignments to make an initial assessment for the type of software application and protocol associated with the received packet flow. Based on this initial assessment, IDS selects and applies a first set of patterns, i.e., attack definitions. This initial set may include protocol-specific attack definitions as well as default attack definitions. In addition, IDS 10 monitors for traffic flowing in the opposite direction to intercept any response from the server. That is, IDS 10 identifies any response packet flow from the server, which typically utilizes the same or substantially similar network addresses, ports and/or protocol as the initial client request. Upon intercepting the response packet flow, IDS 10 performs an additional application identification test to confirm whether its initial assessment of the type of software application and protocol for the particular client-server communication session was correct. If not, IDS 10 may reclassify the communication as a different type of software application and protocol, and select and apply additional attack definitions based on the updated classification. That is, IDS 10 may retrieve the cached assessment of the application and then use the response packet flow to either confirm or modify the assessment, again caching the result. Moreover, IDS 10 may apply the newly selected attack definitions to the first packet flow, e.g. the packet flow initially received from the client. That is, IDS 10 may include buffering capability to temporarily store the packet flow on which the initial assessment of application type was based, and apply any newly selected attack definitions to the buffered packet flow.” (Yang, 6:9-44.) “Security management module 44 presents a user interface by which administrator 42 configures IDS 20. For example, administrator 42 may configure IDS 20 to monitor particular subnets of the enterprise network. In addition, security management module 44 presents a user interface by

QPrivacy USA LLC v. Cisco Systems Inc., 2:24-cv-00855
Exhibit 7 – Invalidity Claim Chart for U.S. Patent Nos. 11,816,249 and 11,106,824

U.S. Patent No. 11,816,249	
Claim	Relevant Disclosure of Yang
	which administrator 42 may specify attack definitions 33, which security management module 44 relays to stateful inspection engine 28. In one embodiment, attack definitions 33 may be compound attack definitions. Moreover, security management module 44 may present a user interface by which administrator 42 may modify assumptions regarding packet flow characteristics, such as the highest priority packet flows for monitoring, port bindings for applications, or other features of determining a type of application and protocol associated with the packet flow.” (Yang, 7:24-38.) “Stateful inspection engine 28 receives transaction data 34, application-layer elements 36 and protocol anomaly data 38 from protocol decoders 30. Stateful inspection engine 28 applies attack definitions 33 to protocol-specific application-layer elements 36 and anomaly data 38 to detect and prevent network attacks and other security risks.” (Yang, 8:55-60.) “FIG. 3 is a block diagram that illustrates an example embodiment of stateful inspection engine 28. In the example embodiment, stateful inspection engine 28 includes reassembly module 50, application identification module 51, and attack detection module 52. In addition, stateful inspection engine 28 includes patterns table 54, data buffer 55, anomalies table 56, and attack definitions 33. Reassembly module 50 receives inbound network traffic 24 and reassembles application-layer communications 32 from the packet flows. Reassembly module 50 forwards the reassembled application-layer communications 32 to the appropriate protocol decoders 30 for processing. Stateful inspection engine 28 stores attack definitions 33 received from security management module 44. Attack definitions 33 may be stored, for example, in a computer-readable medium, such as random access memory (“RAM”). Each of attack definitions 33 specifies a combination of one or more patterns specified within patterns

QPrivacy USA LLC v. Cisco Systems Inc., 2:24-cv-00855
Exhibit 7 – Invalidity Claim Chart for U.S. Patent Nos. 11,816,249 and 11,106,824

U.S. Patent No. 11,816,249	
Claim	Relevant Disclosure of Yang
	table 54 and one or more protocol-specific anomalies specified within anomalies table 56.” (Yang, 9:6-26.) “Initially, security management module 44 receives configuration information from administrator 42 and, in response, configures IDS 20 to monitor a network or portions thereof (subnets) of interest (70). During this process, configuration manager 44 may present a user interface by which administrator 42 specifies patterns or other attack definitions 33.” (Yang, 11:56-62.) “FIG. 6 is an example user interface 100 presented by an IDS with which an administrator interacts to create a pattern for detecting attacks. In this example, user interface 100 allows the administrator to define a compound network attack definition. As illustrated in FIG. 6, user interface 100 includes an input 102 by which the administrator may provide a name for the compound network attack. In addition, interface 100 includes an input 104 by which the administrator may provide a brief description for the compound network attack. The example of FIG. 6 illustrates a compound network attack for detecting directory traversal attempts against an Apache web server. In addition, interface 100 includes inputs 106 by which the administrator may provide other data for the compound network attack, such as a severity level 108, keywords for use in indexing and analysis, and other data. FIG. 7 is an example of user interface 110 presented when the user selects the “Signatures” tab of user interface 100 (FIG. 6). As illustrated, user interface 110 includes an input 112 with which the administrator specifies a protocol to which the compound attack definition pertains. In this example, the administrator has selected HTTP as this exemplary compound network attack definition relates to an Apache web server. User interface 110 further includes an input 111 by which the administrator selects from a list of contexts for the particular protocol selected in input 112. For example, input 111 provides a drop-down menu by which the administrator selects HTTP

QPrivacy USA LLC v. Cisco Systems Inc., 2:24-cv-00855
Exhibit 7 – Invalidity Claim Chart for U.S. Patent Nos. 11,816,249 and 11,106,824

U.S. Patent No. 11,816,249	
Claim	Relevant Disclosure of Yang
	contexts, such as HTTP Authorization, HTTP data, HTTP form data and other contexts available from the headers in an HTTP communication. User interface 110 further includes an input 114 by which the administrator selects a scope for the compound attack definition is to be applied. In particular, input 114 allows the administrator to specify whether the attack is session based (i.e., to be applied over the life-time of a session and across all transactions associated with the session) or transaction based (i.e., applied on a per transaction basis and cleared at the end of each transaction). User interface 110 further includes an input area 118 by which the administrator defines the constituent members of the compound attack definition. In particular, the administrator may add patterns by selecting Add button 120 or protocol anomalies by selecting Add button 122. Input area 118 lists the members currently defined for the compound attack definition. In this example, two patterns 124, 126 have been defined in the form of regular expressions. User interface 110 also includes an input 130 by which the administrator controls whether the constituent members must be detected in the order presented within input area 118 or whether the members may be detected in any order within the scope defined by input area 114.” (Yang, 15:11-62.)
1[d]. determining, by the remote server, based on a comparison of the determined content, whether to modify the at least one data packet, and if so, modifying the at least one data packet; and	Yang discloses this element and/or renders it obvious either alone or in combination with other references and/or the knowledge of one of ordinary skill in the art. For example, Yang discloses blocking communications (<i>e.g.</i>, dropping packets) in accordance with a determination of a security risk.

U.S. Patent No. 11,816,249	
Claim	Relevant Disclosure of Yang
	<pre> graph TD 70[RECEIVE CONFIGURATION INFORMATION] --> 72[MONITOR NETWORK TRAFFIC] 72 --> 74[IDENTIFY PACKET FLOWS AND UPDATE FLOW TABLE] 74 --> 80[REASSEMBLE PACKETS TO FORM APPLICATION-LAYER COMMUNICATIONS] 80 --> 81[IDENTIFY APPLICATION] 81 --> 82[INVOKE PROTOCOL DECODERS] 82 --> 84[OUTPUT TRANSACTION DATA, APPLICATION-LAYER ELEMENTS AND PROTOCOL ANOMALIES] 84 --> 88[SELECT ATTACK DEFINITIONS] 88 --> 90[APPLY ATTACK DEFINITIONS] 90 --> 91{MATCH?} 91 -- YES --> 92[OUTPUT ALERT AND TAKE PROGRAMMED RESPONSE] 91 -- NO --> 94[FORWARD PACKETS AND RE-CHECK LATER PACKETS] 92 --> END([END]) 94 --> END </pre> FIG. 4 (Yang, Fig. 4.)

Claim	U.S. Patent No. 11,816,249
	Relevant Disclosure of Yang
	<pre>graph TD START([START]) --> 150[RECEIVE PACKET FROM CLIENT] 150 --> 152[BUFFER PACKETS OF PACKET FLOW] 152 --> 154[DETERMINE TARGET APPLICATION OF PACKET] 154 --> 156{MALICIOUS?} 156 -- NO --> 158[FORWARD PACKETS TO SERVER] 158 --> 160[RECEIVE PACKET FROM SERVER] 160 --> 162[BUFFER PACKETS OF PACKET FLOW] 162 --> 164[RE-EVALUATE TYPE OF APPLICATION AND PROTOCOL] 164 --> 166{SAME?} 166 -- YES --> 168[RECLASSIFY COMMUNICATION SESSION] 166 -- NO --> 170{MALICIOUS?} 168 --> 170 170 -- YES --> 174[OUTPUT ALERT AND TAKE PROGRAMMED RESPONSE] 170 -- NO --> 172[FORWARD PACKETS TO CLIENT] 174 --> 150 172 --> 150</pre> FIG. 5 (Yang, Fig. 5.)

QPrivacy USA LLC v. Cisco Systems Inc., 2:24-cv-00855
Exhibit 7 – Invalidity Claim Chart for U.S. Patent Nos. 11,816,249 and 11,106,824

U.S. Patent No. 11,816,249	
Claim	Relevant Disclosure of Yang
	“IDS 10 applies the attack definitions to the elements and the protocol-specific anomalies identified by the protocol decoders to detect and prevent network attacks. For example, a system administrator may specify a compound network attack that includes the protocol anomaly of repeated FTP login failure and a pattern that matches a login username of “root.” In this manner, the system administrator may combine pattern analysis with protocol anomalies to define complex attack definitions. In the event of a network attack, IDS 10 may take one or more programmed actions, such as automatically dropping packet flows associated with the application-layer communications within which the network attack was detected. In accordance with the techniques described herein, IDS 10 may attempt to identify the software application and protocol associated with each communication session, i.e., each corresponding pair of packet flows between a given client and server. IDS 10 uses the identified type of software application and underlying protocol type for the communication session to control selection of any protocol-specific patterns to be applied to the packet flows. IDS 10 may temporarily cache the initial assessment of the software application for later comparison and confirmation.” (Yang, 5:53-6:8.) “Flow analysis module 25 receives inbound traffic 24 and identifies network flows within the traffic. Each network flow represents a flow of packets in one direction within the network traffic and is identified by at least a source address, a destination address and a communication protocol. Flow analysis module 25 may utilize additional information to specify network flows, including source media access control (“MAC”) address, destination MAC address, source port, and destination port. Other embodiments may use other information to identify network flows, such as IP addresses. Flow analysis module 25 maintains data within flow table 35 that describes each active packet flow present within the network traffic. Flow table 35 specifies network elements associated with each active packet flow, i.e., low-level information such as source and destination devices and ports associated with the packet flow. In addition, flow

QPrivacy USA LLC v. Cisco Systems Inc., 2:24-cv-00855
Exhibit 7 – Invalidity Claim Chart for U.S. Patent Nos. 11,816,249 and 11,106,824

U.S. Patent No. 11,816,249	
Claim	Relevant Disclosure of Yang
	table 35 may identify pairs of packet flows that collectively form a single communication session between a client and server. For example, flow table 35 may designate communication session as pairs of packet flows in opposite directions for flows sharing at least some common network addresses, ports and protocol.” (Yang, 7:39-60.) “To detect an attack or other malicious activity, attack detection module 52 applies attack definitions 33 to application-layer elements 36 and protocol anomaly data 38 received from protocol decoders 30. In particular, for each of attack definitions 33, attack detection module 52 selects the one or more patterns within patterns table 52 specified by the attack definition and determines whether any of application-layer elements 36 match the defined patterns. Each of the patterns may be defined as a respective “regular expression,” which generally refers to a formula that is used to match patterns within data. In addition to determining whether the defined patterns are present, attack detection module 52 may determine whether any protocol anomalies detected by protocol decoders 30 match the protocol anomalies specified by attack definitions 33. Attack detection module 52 determines that the corresponding packet flow matches one of attack definitions 33 when both the patterns and protocol anomalies specified by the compound attack definition are detected within a given communication session. Further, each of attack definitions 33 may specify whether the pattern matching and protocol anomalies must be satisfied on a per-transaction basis or over the lifetime of the communication session.” (Yang, 11:20-30.) “Upon receiving data from protocol decoders 30, stateful inspection engine 28 selects the attack definitions 33 that are defined for the corresponding protocol, and optionally a set of default attack definitions when no particular type of application has been identified (88). Stateful inspection engine 28 then applies the selected compound attack definitions to determine whether the communication session represents a security risk (90). When applying a given compound attack definition, stateful inspection engine 28 determines whether all of the specified patterns and protocol anomalies are satisfied for any given communication session between peers, either on a per-transaction basis or over the lifetime

QPrivacy USA LLC v. Cisco Systems Inc., 2:24-cv-00855
Exhibit 7 – Invalidity Claim Chart for U.S. Patent Nos. 11,816,249 and 11,106,824

U.S. Patent No. 11,816,249	
Claim	Relevant Disclosure of Yang
	of the communication session, as specified by the compound attack definition. Moreover, if required by the compound attack definition, stateful inspection engine 28 may determine whether the specified patterns and protocol anomalies are satisfied in a required order. In the event a security risk (i.e., match) is detected (91), stateful inspection engine 28 outputs alert 40 to security management module 44 for logging and further analysis (92). In addition, stateful inspection engine 28 may take any of a number of programmed responses, such as dropping the packets associated with the communication session, automatically closing the communication session or other action. If no security risk is detected for a given application-layer communication session, forwarding component 31 forwards the packet flow to the destination (94). IDS 20 then waits for a response packet flow and reanalyzes the packet flow in light of the response to check the initial determination.” (Yang, 12:51-13:12.) See also claim element 1[c].
1[e]. sharing, by the remote server, the modified communication.	Yang discloses this element and/or renders it obvious either alone or in combination with other references and/or the knowledge of one of ordinary skill in the art. For example, Yang discloses blocking and allowing communications according to whether a security risk is determined. “IDS 10 applies the attack definitions to the elements and the protocol-specific anomalies identified by the protocol decoders to detect and prevent network attacks. For example, a system administrator may specify a compound network attack that includes the protocol anomaly of repeated FTP login failure and a pattern that matches a login username of “root.” In this manner, the system administrator may combine pattern analysis with protocol anomalies to define complex attack definitions. In the event of a network attack, IDS 10 may take one or more programmed actions, such as automatically dropping packet flows associated with the application-layer communications within which the network attack was detected.

QPrivacy USA LLC v. Cisco Systems Inc., 2:24-cv-00855
Exhibit 7 – Invalidity Claim Chart for U.S. Patent Nos. 11,816,249 and 11,106,824

U.S. Patent No. 11,816,249	
Claim	Relevant Disclosure of Yang
	In accordance with the techniques described herein, IDS 10 may attempt to identify the software application and protocol associated with each communication session, i.e., each corresponding pair of packet flows between a given client and server. IDS 10 uses the identified type of software application and underlying protocol type for the communication session to control selection of any protocol-specific patterns to be applied to the packet flows. IDS 10 may temporarily cache the initial assessment of the software application for later comparison and confirmation.” (Yang, 5:53-6:8.) “Flow analysis module 25 receives inbound traffic 24 and identifies network flows within the traffic. Each network flow represents a flow of packets in one direction within the network traffic and is identified by at least a source address, a destination address and a communication protocol. Flow analysis module 25 may utilize additional information to specify network flows, including source media access control (“MAC”) address, destination MAC address, source port, and destination port. Other embodiments may use other information to identify network flows, such as IP addresses. Flow analysis module 25 maintains data within flow table 35 that describes each active packet flow present within the network traffic. Flow table 35 specifies network elements associated with each active packet flow, i.e., low-level information such as source and destination devices and ports associated with the packet flow. In addition, flow table 35 may identify pairs of packet flows that collectively form a single communication session between a client and server. For example, flow table 35 may designate communication session as pairs of packet flows in opposite directions for flows sharing at least some common network addresses, ports and protocol.” (Yang, 7:39-60.) See also claim elements 1[pre] and 1[d].
2. The method of claim 1, wherein determining a content of the at least one data packet comprises	Yang discloses this element and/or renders it obvious either alone or in combination with other references and/or the knowledge of one of ordinary skill in the art.

QPrivacy USA LLC v. Cisco Systems Inc., 2:24-cv-00855
Exhibit 7 – Invalidity Claim Chart for U.S. Patent Nos. 11,816,249 and 11,106,824

U.S. Patent No. 11,816,249	
Claim	Relevant Disclosure of Yang
determining at least one data type of the at least one data packet of the communication in accordance with characteristics of the at least one data packet, and wherein determining whether to modify the at least one data packet is based on the determined data type.	For example, Yang discloses identifying an application or protocol based on a TCP port in a packet header without decrypting the packet and modifying a packet based on packet characteristics. “IDS 20 may use a minimum data size of the reassembled TCP segments, in addition to the signature, in order to identify the types of applications. Certain applications may require a minimum amount of data, so IDS 20 may distinguish malicious packet flows by determining whether the packet flow contains enough data for the identified protocol. Moreover, IDS 20 may not necessarily recognize every application. In one embodiment, when an application is unknown, IDS 20 may simply forward the packet flow. If IDS 20 cannot identify a given application, it may be because that application is not a typical target for a malicious packet flow. Other embodiments may take other actions for unidentified applications, however, such as discarding all packets which target unknown applications or applying a default signature to all packet flows associated with unknown application types. Other embodiments may also utilize other protocols, such as the user datagram protocol (UDP); IDS 20 accordingly may require a minimum data size of UDP segments in order to identify the application associated with the UDP segments. For each packet flow, stateful inspection engine 28 buffers a copy of the packet flow and reassembles the buffered packet flow to form application-layer communications 32. For example, stateful inspection engine 28 may reconstruct TCP segments into application-layer communications 32, which represent protocol-specific messages. Stateful inspection engine 28 invokes the appropriate one of protocol decoders 30 based on the identified type of application determination to analyze the application-layer communications 32. Protocol decoders 30 represent a set of one or more protocol-specific software modules. Each of protocol decoders 30 corresponds to a different communication protocol or service. Examples of communication protocols that may be supported by protocol decoders 30 include the HyperText Transfer Protocol (“HTTP”), the File Transfer Protocol (“FTP”), the Network News Transfer Protocol (“NNTP”), the Simple

QPrivacy USA LLC v. Cisco Systems Inc., 2:24-cv-00855
Exhibit 7 – Invalidity Claim Chart for U.S. Patent Nos. 11,816,249 and 11,106,824

U.S. Patent No. 11,816,249	
Claim	Relevant Disclosure of Yang
	Mail Transfer Protocol (“SMTP”), Telnet, Domain Name System (“DNS”), Gopher, Finger, the Post Office Protocol (“POP”), the Secure Socket Layer (“SSL”) protocol, the Lightweight Directory Access Protocol (“LDAP”), Secure Shell (“SSH”), Server Message Block (“SMB”) and other protocols.” (Yang, 8:8–48.) See also claim elements 1[b] and 1[d].
3. The method of claim 2, wherein the preference list includes a list of the of forbidden and/or allowed data packet types for sharing during communication.	Yang discloses this element and/or renders it obvious either alone or in combination with other references and/or the knowledge of one of ordinary skill in the art. For example, Yang discloses attack definitions to detect and prevent network attacks and other security risks. These definitions may be edited. See also claim elements 1[c] and 4[c].
4. The method of claim 1, further comprising:	See claim 1.
4[a]. determining, by the remote server, at least one communication data pattern in accordance with a behavior range of the data packet, and wherein the content of the at least one data packet is not decrypted by the remote server, and the determination of the at least one communication data pattern is performed by the remote server in real time during a communication session between the remote server and the user's device;	Yang discloses this element and/or renders it obvious either alone or in combination with other references and/or the knowledge of one of ordinary skill in the art. For example, Yang discloses protocol-specific anomaly analysis, which includes using anomaly information to detect sophisticated attack behaviors.

U.S. Patent No. 11,816,249	
Claim	Relevant Disclosure of Yang
	The diagram illustrates an Intrusion Detection System (20) divided into two main sections by a dashed line labeled 'FORWARDING PLANE' (22). On the left side, 'INBOUND TRAFFIC' (24) enters the system. It passes through a 'FLOW ANALYSIS MODULE' (25), which is connected to a 'FLOW TABLE' (35). The traffic then moves to a 'STATEFUL INSPECTION ENGINE' (28). This engine is connected to 'PROTOCOL DECODERS' (30) via a bidirectional arrow (32). The engine also has a bidirectional connection (34) to a 'SECURITY MANAGEMENT MODULE' (44). The 'SECURITY MANAGEMENT MODULE' (44) is further connected to an 'ADMINISTRATOR' (42) via a bidirectional arrow (40). The 'STATEFUL INSPECTION ENGINE' (28) sends data to a 'FORWARDING COMPONENT' (31) via a bidirectional arrow (36). Finally, 'OUTBOUND TRAFFIC' (26) exits the system from the forwarding component. The entire system is labeled 'INTRUSION DETECTION SYSTEM' (20) at the top. FIG. 2 (Yang, Fig. 2.)

Claim	U.S. Patent No. 11,816,249
	Relevant Disclosure of Yang
	FIG. 5 (Yang, Fig. 5.)

QPrivacy USA LLC v. Cisco Systems Inc., 2:24-cv-00855
Exhibit 7 – Invalidity Claim Chart for U.S. Patent Nos. 11,816,249 and 11,106,824

U.S. Patent No. 11,816,249	
Claim	Relevant Disclosure of Yang
	“In the event a security risk is detected, stateful inspection engine 28 outputs alert 40 to security management module 44 (FIG. 2) for logging and further analysis. Stateful inspection engine 28 may also direct forwarding component 31 to automatically drop the packet flow associated with the application-layer communications within which the network attack was detected. In this manner, stateful inspection engine 28 combines pattern matching with protocol-specific anomaly analysis to detect sophisticated attack behaviors.” (Yang, 11:43-51.) “Upon receiving data from protocol decoders 30, stateful inspection engine 28 selects the attack definitions 33 that are defined for the corresponding protocol, and optionally a set of default attack definitions when no particular type of application has been identified (88). Stateful inspection engine 28 then applies the selected compound attack definitions to determine whether the communication session represents a security risk (90). When applying a given compound attack definition, stateful inspection engine 28 determines whether all of the specified patterns and protocol anomalies are satisfied for any given communication session between peers, either on a per-transaction basis or over the lifetime of the communication session, as specified by the compound attack definition. Moreover, if required by the compound attack definition, stateful inspection engine 28 may determine whether the specified patterns and protocol anomalies are satisfied in a required order.” (Yang, 12:55-67.) See also claim elements 1[b] and 1[c].
4[b]. wherein the determining whether to modify the at least one data packet, by the remote server, is further based on a comparison of the determined data pattern and the preference list, whether to modify the at least one data packet, and if so, modifying the at least one data packet; and	Yang discloses this element and/or renders it obvious either alone or in combination with other references and/or the knowledge of one of ordinary skill in the art. For example, Yang discloses blocking communications (<i>e.g.</i>, dropping packets) in accordance with a determination of a security risk (<i>e.g.</i>, a matching pattern). “IDS 10 applies the attack definitions to the elements and the protocol-specific anomalies identified by the protocol decoders to detect and prevent network attacks. For example, a

QPrivacy USA LLC v. Cisco Systems Inc., 2:24-cv-00855
Exhibit 7 – Invalidity Claim Chart for U.S. Patent Nos. 11,816,249 and 11,106,824

U.S. Patent No. 11,816,249	
Claim	Relevant Disclosure of Yang
	system administrator may specify a compound network attack that includes the protocol anomaly of repeated FTP login failure and a pattern that matches a login username of “root.” In this manner, the system administrator may combine pattern analysis with protocol anomalies to define complex attack definitions. In the event of a network attack, IDS 10 may take one or more programmed actions, such as automatically dropping packet flows associated with the application-layer communications within which the network attack was detected. In accordance with the techniques described herein, IDS 10 may attempt to identify the software application and protocol associated with each communication session, i.e., each corresponding pair of packet flows between a given client and server. IDS 10 uses the identified type of software application and underlying protocol type for the communication session to control selection of any protocol-specific patterns to be applied to the packet flows. IDS 10 may temporarily cache the initial assessment of the software application for later comparison and confirmation.” (Yang, 5:53-6:8.) “FIG. 2 is a block diagram illustrating an example embodiment of an IDS 20. In the illustrated example, IDS 20 includes a forwarding plane 22 that transparently monitors inbound network traffic 24 and forwards the network traffic as outbound network traffic 26. In the example illustrated by FIG. 2, forwarding plane 22 includes flow analysis module 25, stateful inspection engine 28, protocol decoders 30, forwarding component 31 and security management module 44. Security management module 44 presents a user interface by which administrator 42 configures IDS 20. For example, administrator 42 may configure IDS 20 to monitor particular subnets of the enterprise network. In addition, security management module 44 presents a user interface by which administrator 42 may specify attack definitions 33, which security management module 44 relays to stateful inspection engine 28. In one embodiment, attack definitions 33 may be compound attack definitions. Moreover, security management

QPrivacy USA LLC v. Cisco Systems Inc., 2:24-cv-00855
Exhibit 7 – Invalidity Claim Chart for U.S. Patent Nos. 11,816,249 and 11,106,824

U.S. Patent No. 11,816,249	
Claim	Relevant Disclosure of Yang
	module 44 may present a user interface by which administrator 42 may modify assumptions regarding packet flow characteristics, such as the highest priority packet flows for monitoring, port bindings for applications, or other features of determining a type of application and protocol associated with the packet flow. Flow analysis module 25 receives inbound traffic 24 and identifies network flows within the traffic. Each network flow represents a flow of packets in one direction within the network traffic and is identified by at least a source address, a destination address and a communication protocol. Flow analysis module 25 may utilize additional information to specify network flows, including source media access control (“MAC”) address, destination MAC address, source port, and destination port. Other embodiments may use other information to identify network flows, such as IP addresses. Flow analysis module 25 maintains data within flow table 35 that describes each active packet flow present within the network traffic. Flow table 35 specifies network elements associated with each active packet flow, i.e., low-level information such as source and destination devices and ports associated with the packet flow. In addition, flow table 35 may identify pairs of packet flows that collectively form a single communication session between a client and server. For example, flow table 35 may designate communication session as pairs of packet flows in opposite directions for flows sharing at least some common network addresses, ports and protocol.” (Yang, 7:16-60.) See also claim elements 1[c] and 1[d].
4[c]. wherein the preference list includes a list of forbidden and/or allowed data packet types and communication data patterns for sharing during communication.	Yang discloses this element and/or renders it obvious either alone or in combination with other references and/or the knowledge of one of ordinary skill in the art. For example, Yang discloses details about attack definitions, indicating patterns and protocol anomalies that may be specified as occurring in a required order.

Claim	U.S. Patent No. 11,816,249
	Relevant Disclosure of Yang
	FIG. 5 (Yang, Fig. 5.)

QPrivacy USA LLC v. Cisco Systems Inc., 2:24-cv-00855
Exhibit 7 – Invalidity Claim Chart for U.S. Patent Nos. 11,816,249 and 11,106,824

U.S. Patent No. 11,816,249	
Claim	Relevant Disclosure of Yang
	“Upon receiving data from protocol decoders 30, stateful inspection engine 28 selects the attack definitions 33 that are defined for the corresponding protocol, and optionally a set of default attack definitions when no particular type of application has been identified (88). Stateful inspection engine 28 then applies the selected compound attack definitions to determine whether the communication session represents a security risk (90). When applying a given compound attack definition, stateful inspection engine 28 determines whether all of the specified patterns and protocol anomalies are satisfied for any given communication session between peers, either on a per-transaction basis or over the lifetime of the communication session, as specified by the compound attack definition. Moreover, if required by the compound attack definition, stateful inspection engine 28 may determine whether the specified patterns and protocol anomalies are satisfied in a required order. In the event a security risk (i.e., match) is detected (91), stateful inspection engine 28 outputs alert 40 to security management module 44 for logging and further analysis (92). In addition, stateful inspection engine 28 may take any of a number of programmed responses, such as dropping the packets associated with the communication session, automatically closing the communication session or other action. If no security risk is detected for a given application-layer communication session, forwarding component 31 forwards the packet flow to the destination (94). IDS 20 then waits for a response packet flow and reanalyzes the packet flow in light of the response to check the initial determination.” (Yang, 12:51-13:12.) See also claim element 1[c].
5. The method of claim 1, wherein the modification of the group consisting of: selected from the group consisting of: blocking the at least one data packet that is not compatible with the preference list, data nullification, data randomization, content modification,	Yang discloses this element and/or renders it obvious either alone or in combination with other references and/or the knowledge of one of ordinary skill in the art. For example, Yang discloses blocking communications (<i>e.g.</i>, dropping packets) in accordance with a determination of a security risk (<i>e.g.</i>, a matching pattern). See claim elements 1[c] and 1[d].

QPrivacy USA LLC v. Cisco Systems Inc., 2:24-cv-00855
Exhibit 7 – Invalidity Claim Chart for U.S. Patent Nos. 11,816,249 and 11,106,824

U.S. Patent No. 11,816,249	
Claim	Relevant Disclosure of Yang
change of encoding, change of file template, change of header, change of footer, addition of a predetermined data packet, and encryption.	
6. The method of claim 2, wherein the data type of the at least one data packet is determined based at least in part on metadata of at least one data packet of the communication.	Yang discloses this element and/or renders it obvious either alone or in combination with other references and/or the knowledge of one of ordinary skill in the art. For example, Yang discloses identifying an application or protocol based on a TCP port in a packet header, which is metadata. See claim element 1[b] and claim 2.
7. The method of claim 4, wherein the data pattern is determined based on communication behavior, wherein the communication behavior consists of data packet frequency, data packet size, data packet speed, data packet count, data packet ratio compared to other data type flow, data packet repetition, or data packet order.	Yang discloses this element and/or renders it obvious either alone or in combination with other references and/or the knowledge of one of ordinary skill in the art. For example, Yang discloses protocol-specific anomaly analysis, which includes using anomaly information to detect sophisticated attack behaviors. Attack definition for the analysis indicates patterns, which may be a required order. See claim elements 1[b], 1[c], 1[d], and claim 4.
8. The method of claim 1, further comprising, the remote server maintaining communication between the remote server and the user's device by sharing of the modified communication.	Yang discloses this element and/or renders it obvious either alone or in combination with other references and/or the knowledge of one of ordinary skill in the art. For example, Yang discloses blocking and allowing communications according to whether a security risk is determined. See claim elements 1[pre], 1[b], and 1[e].

QPrivacy USA LLC v. Cisco Systems Inc., 2:24-cv-00855
Exhibit 7 – Invalidity Claim Chart for U.S. Patent Nos. 11,816,249 and 11,106,824

U.S. Patent No. 11,816,249	
Claim	Relevant Disclosure of Yang
9. The method of claim 1, wherein the data type is determined based on data packet characteristics selected from a group consisting of: data packet header, data packet footer, version number, IP (Internet Protocol) address, HTTPS (HyperText Transfer Protocol Secure), file extension, encryption method, encoding method, keywords, driver, and communication protocol.	Yang discloses this element and/or renders it obvious either alone or in combination with other references and/or the knowledge of one of ordinary skill in the art. For example, Yang discloses identifying an application or protocol based on a TCP port in a packet header. See claim element 1[b] and claim 2.
10. The method of claim 1, wherein the remote server is a gatekeeper server.	Yang discloses this element and/or renders it obvious either alone or in combination with other references and/or the knowledge of one of ordinary skill in the art. For example, Yang's IDS intercepts packet flows and filters flows associated with security risks. Yang's IDS manages communications in both directions. See claim elements 1[pre] and 1[d].
11. The method of claim 2, further comprising, the remote server issuing an alert to the user device based on the comparison of the determined data type and the preference list.	Yang discloses this element and/or renders it obvious either alone or in combination with other references and/or the knowledge of one of ordinary skill in the art. For example, Yang discloses outputting an alert in the event of a match corresponding to a security risk. See claim element 1[d].
12. A method of dynamic management of encrypted data	See claim element 1[pre].

QPrivacy USA LLC v. Cisco Systems Inc., 2:24-cv-00855
Exhibit 7 – Invalidity Claim Chart for U.S. Patent Nos. 11,816,249 and 11,106,824

U.S. Patent No. 11,816,249	
Claim	Relevant Disclosure of Yang
during communication between a remote server and a user's device, the method comprising:	
12[a]. receiving, by the remote server, a communication comprising encrypted data packets;	See claim element 1[a].
12[b]. determining, by the remote server, at least one communication data pattern of at least one data packet of the communication in accordance with a behavior range of the data packet, and wherein content of the at least one data packet is not decrypted by the remote server, and the determination of the at least one data pattern is performed by the remote server in real time during a communication session between the remote server and the user's device;	See claim elements 1[b] and 4[a].
12[c]. storing, by the remote server, a preference list;	See claim element 1[c].
12[d]. determining, by the remote server, based on a comparison of the determined data pattern and the preference list, whether to modify the at least one data packet, and if so, modifying the at least one data packet; and	See claim elements 1[d] and 4[b].

***QPrivacy USA LLC v. Cisco Systems Inc.*, 2:24-cv-00855**
Exhibit 7 – Invalidity Claim Chart for U.S. Patent Nos. 11,816,249 and 11,106,824

U.S. Patent No. 11,816,249	
Claim	Relevant Disclosure of Yang
12[e]. sharing, by the remote server, the modified communication.	See claim element 1[e].
13. The method of claim 12, wherein the preference list includes a list of forbidden and/or allowed communication data patterns for sharing during communication.	See claim 4[c].
14. The method of claim 12, wherein the modification of the communication is selected from the group consisting of: blocking the at least one data packet that is not compatible with the preference list, data nullification, data randomization, content modification, change of encoding, change of file template, change of header, change of footer, addition of a predetermined data packet, and encryption.	See claim 5.
15. The method of claim 12, wherein the communication data pattern of the at least one data packet is determined based on communication behavior, wherein the communication behavior consists of data packet frequency, data packet size, data packet speed, data packet count, data packet ratio compared to other data	See claim 7.

QPrivacy USA LLC v. Cisco Systems Inc., 2:24-cv-00855
Exhibit 7 – Invalidity Claim Chart for U.S. Patent Nos. 11,816,249 and 11,106,824

U.S. Patent No. 11,816,249	
Claim	Relevant Disclosure of Yang
type flow, data packet repetition, or data packet order.	
16. The method of claim 12, further comprising, the remote server maintaining communication between the remote server and the user's device by sharing of the modified communication.	See claim 8.
17. The method of claim 12, wherein the remote server is a gatekeeper server.	See claim 10.
18. The method of claim 12, further comprising, the remote server issuing an alert to the user device based on the comparison of the determined data pattern and the preference list.	See claims 4 and 11.
19. A system for dynamic management of encrypted data during communication between a remote server and a user's device, the system comprising:	See claim element 1[pre].
19[a]. a memory;	Yang discloses this element and/or renders it obvious either alone or in combination with other references and/or the knowledge of one of ordinary skill in the art. For example, Yang's IDS comprises a memory. See Yang, 9:19-26.

***QPrivacy USA LLC v. Cisco Systems Inc.*, 2:24-cv-00855**
Exhibit 7 – Invalidity Claim Chart for U.S. Patent Nos. 11,816,249 and 11,106,824

U.S. Patent No. 11,816,249	
Claim	Relevant Disclosure of Yang
19[b]. a communication data type database comprising at least one communication data type corresponding to sharing of at least one data packet with the user's device;	See claim element 1[b] and claim 2.
19[c]. a preference list database;	See claim element 1[c].
19[d]. a communication module configured to allow communication between the remote server and the at least one user's device, and	See claim elements 1[pre] and 1[b].
19[e]. a processor, coupled to the preference list database, wherein the processor is configured to instruct the remote server to:	Yang discloses this element and/or renders it obvious either alone or in combination with other references and/or the knowledge of one of ordinary skill in the art. For example, Yang's system a processor coupled to a memory for storing the preference list. See Yang, 2:58-3:17, 9:19-26, 19:36-20:60. See also claim element 1[c].
19[f]. determine a content of at least one data packet of the communication in accordance with characteristics of the at least one data packet, and wherein the content of the at least one data packet is not decrypted by the remote server, and the determination of the content is performed by the remote server in real time during a	See claim element 1[b].

***QPrivacy USA LLC v. Cisco Systems Inc.*, 2:24-cv-00855**
Exhibit 7 – Invalidity Claim Chart for U.S. Patent Nos. 11,816,249 and 11,106,824

U.S. Patent No. 11,816,249	
Claim	Relevant Disclosure of Yang
communication session between the remote server and the user's device,	
19[g]. based on a comparison of the determined content and the preference list, determine whether to modify the at least one data packet, and if so, modify the at least one data packet; and	See claim elements 1[c], 1[d], and 4[b].
19[h]. share the modified communication.	See claim element 1[e].
20. The system of claim 19, wherein determining a content of the at least one data packet comprises determining at least one data type of the at least one data packet of the communication in accordance with characteristics of the at least one data packet, and wherein determining whether to modify the at least one data packet is based on the determined data type.	See claim 2.
21. The system of claim 20, wherein the preference list database comprises a list of forbidden and/or allowed data packet types for sharing during communication.	See claim 3.

***QPrivacy USA LLC v. Cisco Systems Inc.*, 2:24-cv-00855**
Exhibit 7 – Invalidity Claim Chart for U.S. Patent Nos. 11,816,249 and 11,106,824

U.S. Patent No. 11,816,249	
Claim	Relevant Disclosure of Yang
22. The system of claim 19, wherein, the processor is further configured to determine at least one communication data pattern in accordance with a behavior range of the data packet, and wherein the content of the at least one data packet is not decrypted by the remote server, and the determination of the at least one communication data pattern is performed by the remote server in real time during a communication session between the remote server and the user's device; and wherein the determining whether to modify the at least one data packet, by the remote server, is further based on a comparison of the determined data pattern and the preference list, whether to modify the at least one data packet, and if so, modifying the at least one data packet; and wherein the preference list includes a list of forbidden and/or allowed data packet types and communication data patterns for sharing during communication.	See claim 4.
23. The system of claim 20, wherein the data type of the at least one data packet is determined based at least in	See claim 6.

***QPrivacy USA LLC v. Cisco Systems Inc.*, 2:24-cv-00855**
Exhibit 7 – Invalidity Claim Chart for U.S. Patent Nos. 11,816,249 and 11,106,824

U.S. Patent No. 11,816,249	
Claim	Relevant Disclosure of Yang
part on metadata of at least one data packet of the communication.	
24. The system of claim 22, wherein the at least one communication data pattern is determined in accordance with a behavior range of the data packet, wherein the behavior range is selected from a group consisting of: data packet frequency, data packet size, data packet speed, data packet count, data packet ratio compared to other data type flow, data packet repetition, and data packet order.	See claim 7.
25. The system of claim 19, wherein the modification of the communication is selected from the group consisting of blocking the at least one data packet that is not compatible with the preference list, data nullification, data randomization, content modification, change of encoding, change of file template, change of header, change of footer, addition of a predetermined data packet and encryption.	See claim 5.
26. The system of claim 19, wherein the remote server is configured to maintain communication between the	See claim 8.

QPrivacy USA LLC v. Cisco Systems Inc., 2:24-cv-00855
Exhibit 7 – Invalidity Claim Chart for U.S. Patent Nos. 11,816,249 and 11,106,824

U.S. Patent No. 11,816,249	
Claim	Relevant Disclosure of Yang
remote server and the user's device by sharing of the modified communication.	
27. The system of claim 20, wherein the data type is determined based on data packet characteristics selected from a group consisting of: data packet header, data packet footer, version number, IP (Internet Protocol) address, HTTPS (HyperText Transfer Protocol Secure), file extension, encryption method, encoding method, keywords, driver, and communication protocol.	See claim 9.
28. The system of claim 19, wherein the remote server is a gatekeeper server.	See claim 10.
29. The system of claim 20, wherein the processor is configured to issue an alert to the user device based on the comparison of the determined data type and the preference list.	See claim 11.
30. The system of claim 22, wherein the preference list includes a list of forbidden and/or allowed data packet types and communication data patterns for sharing during communication, and wherein, if the at	Yang discloses this element and/or renders it obvious either alone or in combination with other references and/or the knowledge of one of ordinary skill in the art. For example, Yang discloses modifying communications without comparing a data pattern, and that communications may be modified by comparing only data types.

***QPrivacy USA LLC v. Cisco Systems Inc.*, 2:24-cv-00855**
Exhibit 7 – Invalidity Claim Chart for U.S. Patent Nos. 11,816,249 and 11,106,824

U.S. Patent No. 11,816,249	
Claim	Relevant Disclosure of Yang
least one data type of the at least one data packet is included in the list of forbidden and/or allowed data packets, the processor is configured to determine whether to modify the at least one data packet based on the determined data type and the preference list without performing a comparison of a determined data pattern and the preference list.	See claim elements 1[d] and 4[c], and claims 2 and 3.

***QPrivacy USA LLC v. Cisco Systems Inc.*, 2:24-cv-00855**
Exhibit 7 – Invalidity Claim Chart for U.S. Patent Nos. 11,816,249 and 11,106,824

U.S. Patent No. 11,106,824	
Claim	Relevant Disclosure of Yang
1. A method of dynamic management of private data during communication between a remote server and a user's device, the method comprising:	To the extent the preamble is limiting, Yang discloses this element and/or renders it obvious either alone or in combination with other references and/or the knowledge of one of ordinary skill in the art. For example, Yang discloses dynamically managing private network communications between an intrusion detection system (IDS) and a user device (<i>e.g.</i>, a “node,” a security management device). The communications may include encrypted communications.

U.S. Patent No. 11,106,824	
Claim	Relevant Disclosure of Yang
	The diagram illustrates a network architecture. A cloud labeled 'PUBLIC NETWORK 6' is connected to a bracketed section labeled 'ENTERPRISE'. Within the enterprise, a 'FIREWALL 9' is connected to an 'IDS 10', which is connected to a 'SWITCH 19'. The switch (19) is connected to a group of nodes labeled '8A', '8N', and '8N' (with three dots between 8N and 8N). These nodes are connected to a 'SECURITY MANAGEMENT DEVICE 14' and an 'IDS 14'. A bracket labeled '18' groups the nodes and the security management device. A bracket labeled '4' groups the firewall, IDS, and switch. A bracket labeled '5' groups the nodes and the security management device. A bracket labeled '2' groups the entire enterprise system. FIG. 1 (Yang, Fig. 1.)

U.S. Patent No. 11,106,824	
Claim	Relevant Disclosure of Yang
	<pre>graph TD 70[70 RECEIVE CONFIGURATION INFORMATION] --> 72[72 MONITOR NETWORK TRAFFIC] 72 --> 74[74 IDENTIFY PACKET FLOWS AND UPDATE FLOW TABLE] 74 --> 80[80 REASSEMBLE PACKETS TO FORM APPLICATION-LAYER COMMUNICATIONS] 80 --> 81[81 IDENTIFY APPLICATION] 81 --> 82[82 INVOKE PROTOCOL DECODERS] 82 --> 84[84 OUTPUT TRANSACTION DATA, APPLICATION-LAYER ELEMENTS AND PROTOCOL ANOMALIES] 84 --> 88[88 SELECT ATTACK DEFINITIONS] 88 --> 90[90 APPLY ATTACK DEFINITIONS] 90 --> 91{91 MATCH?} 91 -- YES --> 92[92 OUTPUT ALERT AND TAKE PROGRAMMED RESPONSE] 91 -- NO --> 94[94 FORWARD PACKETS AND RE-CHECK LATER PACKETS] 92 --> END([END]) 94 --> END</pre> (Yang, Fig. 4.) FIG. 4

QPrivacy USA LLC v. Cisco Systems Inc., 2:24-cv-00855
Exhibit 7 – Invalidity Claim Chart for U.S. Patent Nos. 11,816,249 and 11,106,824

U.S. Patent No. 11,106,824	
Claim	Relevant Disclosure of Yang
	“In general, the invention is directed to techniques for detecting and preventing network attacks, such as Denial of Service Attacks (“DOS”) attacks, network viruses or other malicious activity. More specifically, improved techniques are described herein for identifying the software application and protocol associated with a data stream processed by an intrusion detection system (“IDS”). For example, as described herein, an IDS may analyze both client-to-server and server-to-client packet flows in order to improve the accuracy of the identification. Upon first receiving a first packet flow, for example, a connection request from a client device, the IDS may make an initial assessment as to the type of software application and protocol for the incoming data stream. Upon receiving a response, for example, from a server, the IDS may then attempt to confirm the initial assessment of the application and protocol. If the initial assessment was incorrect, then the IDS may dynamically reclassify the application and protocol associated with the data stream. The IDS may then apply the proper pattern to the data stream based on the reclassification.” (Yang, 1:46-65.) “In one embodiment, the invention is directed to a method comprising receiving, with a network device, a first packet flow within a network from a client to a server, performing an initial identification of a type of software application and communication protocol associated with the first packet flow, applying a first set of patterns to the first packet flow to determine whether the first packet flow represents a network attack, and forwarding the first packet flow to a server. The method further comprises receiving, in response, a second packet flow from the server, associating the first packet flow and the second packet flow as a communication session between the client and the server using the first packet flow and the second packet flow, reevaluating the initial identification of the type of software application and protocol associated with the communication session, selecting a second set of patterns based on the reevaluation of the initial identification of the software application and protocol, and applying the second set of patterns to the first packet flow to re-determine whether the first packet flow represents a network attack.” (Yang, 2:24-43.)

QPrivacy USA LLC v. Cisco Systems Inc., 2:24-cv-00855
Exhibit 7 – Invalidity Claim Chart for U.S. Patent Nos. 11,816,249 and 11,106,824

U.S. Patent No. 11,106,824	
Claim	Relevant Disclosure of Yang
	“In another embodiment, the invention is directed to an intrusion detection system (“IDS”) device comprising a flow analysis module to receive a first packet flow from a client and a second packet flow from a server in response to the first packet flow, a forwarding component to send the first packet flow to the server and the second packet flow to the client, and a stateful inspection engine to apply one or more sets of patterns to the first packet flow to determine whether the first packet flow represents a network attack. The IDS further comprises an application identification module to perform an initial identification of a type of software application and communication protocol associated with the first packet flow, and to reevaluate the identification of the type of software application and protocol according to the second packet flow.” (Yang, 2:44-57.) “FIG. 1 is a block diagram illustrating an exemplary system 2 in which enterprise computer network 4 includes intrusion detection system (“IDS”) 10 that identifies applications and protocols in accordance with the principles of the invention. In the example embodiment of FIG. 1, IDS 10 is a single network device. Network 4 also includes a private enterprise computing network 5 that is coupled to public network 6, such as the Internet. Public network 6 may include, for example, one or more client computing devices. Firewall 9 protects enterprise network 5 and, in particular, internal computing nodes 8A-8N. Computing nodes 8A-8N (“computing nodes 8”) represent any private computing device within enterprise network 5, including workstations, file servers, print servers, database servers, printers and other devices. In the example of FIG. 1, enterprise network 5 includes IDS 10 that monitors traffic flowing between firewall 9 and internal computing nodes 8. As described herein, IDS 10 may analyze network traffic flowing in both directions (i.e., inbound traffic received from public network 6 as well as outbound traffic destined to the public network) to improve the accuracy in detecting network attacks. That is, IDS 10 may analyze both client-to-server and server-to-client communications between public network 6 and computing nodes 8. IDS 10 may analyze the network traffic to correlate traffic in one direction with traffic in the opposite direction for each communication session detected

QPrivacy USA LLC v. Cisco Systems Inc., 2:24-cv-00855
Exhibit 7 – Invalidity Claim Chart for U.S. Patent Nos. 11,816,249 and 11,106,824

U.S. Patent No. 11,106,824	
Claim	Relevant Disclosure of Yang
	within the network traffic. For example, for each client-server communication session, IDS may identify a packet flow in one direction (e.g., a client-to-server communication flow for a particular software application on the client) and a corresponding packet flow in the opposite direction (e.g., response communications flowing from the server to the client for that same software application). In accordance with the principles of the invention, IDS 10 integrates pattern matching with application- and protocol-specific anomaly detection to identify sophisticated attack behaviors. In one embodiment, IDS 10 allows the system administrator to specify attack definitions. In one embodiment, the system administrator may specify compound attack definitions. Further details on application of attack definitions, e.g. compound attack definitions, may be found within U.S. patent application Ser. No. 11/045,572, Guruswamy et al., “Compound Attack Detection in a Computer Network,” filed Jan. 27, 2005, assigned to the assignee of the current application, which is incorporated herein by reference in its entirety.” (Yang, 4:12-58.) “IDS 10 applies the attack definitions to the elements and the protocol-specific anomalies identified by the protocol decoders to detect and prevent network attacks. For example, a system administrator may specify a compound network attack that includes the protocol anomaly of repeated FTP login failure and a pattern that matches a login username of “root.” In this manner, the system administrator may combine pattern analysis with protocol anomalies to define complex attack definitions. In the event of a network attack, IDS 10 may take one or more programmed actions, such as automatically dropping packet flows associated with the application-layer communications within which the network attack was detected. In accordance with the techniques described herein, IDS 10 may attempt to identify the software application and protocol associated with each communication session, i.e., each corresponding pair of packet flows between a given client and server. IDS 10 uses the identified type of software application and underlying protocol type for the communication

QPrivacy USA LLC v. Cisco Systems Inc., 2:24-cv-00855
Exhibit 7 – Invalidity Claim Chart for U.S. Patent Nos. 11,816,249 and 11,106,824

U.S. Patent No. 11,106,824	
Claim	Relevant Disclosure of Yang
	session to control selection of any protocol-specific patterns to be applied to the packet flows. IDS 10 may temporarily cache the initial assessment of the software application for later comparison and confirmation.” (Yang, 5:53-6:8.) “These techniques may allow IDS 10 to provide improved attack detection and prevention in environments where some applications no longer use a static port mapping but instead use dynamic or randomized port assignments in an attempt to evade detection and containment. For example, instant messaging programs may initiate communication on one port, but then agree to communicate thereafter on a distinct port. Other programs, such as peer-to-peer (“P2P”) programs and hacker toolkits such as Metasploit, may determine ports dynamically as well. As described, IDS 10 may make an initial determination for the application type of a packet flow from a client device of public network 6 to a server device, e.g. node 8A, based upon characteristics of the communication in addition to specified port number. IDS 10 may then analyze the return communication from node 8A to the client device to verify the initial assumption or to modify the initial determination if necessary. In some embodiments, enterprise network 5 may include multiple IDSs 10 and 14 located within different regions (e.g., sub-networks) of enterprise network 5. Security management device 18 may operate as a central device for managing IDSs 10 and 14. Although the example illustrated in FIG. 1 is described in terms of dedicated IDSs 10 and 14, the functionality described herein may be incorporated within other devices, such as firewall 9 or switch 19.” (Yang, 6:45-7:2.) “FIG. 2 is a block diagram illustrating an example embodiment of an IDS 20. In the illustrated example, IDS 20 includes a forwarding plane 22 that transparently monitors inbound network traffic 24 and forwards the network traffic as outbound network traffic 26. In the example illustrated by FIG. 2, forwarding plane 22 includes flow analysis module 25, stateful inspection engine 28, protocol decoders 30, forwarding component 31 and security management module 44.