

Exhibit A85 - U.S. Patent No. 9,917,856

US Patent No. 7,185,356 to Buruganahalli¹

Claim	U.S. Patent No. 9,917,856	Disclosure from Buruganahalli
1.0	A method comprising:	<i>See, e.g., Buruganahalli at Abstract:</i> Techniques for destination domain extraction for secure protocols are disclosed. In some embodiments, destination domain extraction for secure protocols includes monitoring network communications between a client and a remote server; determining if the client sends a request to create a secure connection with the remote server (e.g., in which the network communications are initiating a setup for a secure protocol-based connection); and extracting a destination domain from the request to create the secure connection with the remote server. In some embodiments, the secure protocol is a secure sockets layer (SSL) protocol or transport layer security (TLS) protocol, and the destination domain is extracted from the server name indication (SNI) of a client hello message sent from the client to the remote server. In some embodiments, destination domain extraction for secure protocols further includes applying a policy (e.g., a security policy) based on the destination domain to filter traffic using a security device. <i>See, e.g., Buruganahalli at claim 11:</i> A method for destination domain extraction for secure protocols,

¹ Buruganahalli was filed on July 25, 2013 and issued on August 16, 2016, both prior to the earliest priority claim of the '856 patent. Buruganahalli is therefore, prior art under post-AIA 35 U.S.C. §§ 102(a)(1) and (2).

Claim	U.S. Patent No. 9,917,856	Disclosure from Buruganahalli
		comprising: monitoring network communications between a client and a remote server using a security device; determining if the client sends a request to create a secure connection with the remote server, wherein the secure connection utilizes a secure protocol, the secure protocol being a secure sockets layer (SSL) protocol or transport layer security (TLS) protocol; and extracting a destination domain from the request to create the secure connection with the remote server, comprising: before the secure connection is created, extracting the destination domain from a server name indication (SNI) of a client hello message sent from the client to the remote server, the secure connection including an encrypted message, wherein the extracting of the destination domain from the SNI of the client hello message sent from the client to the remote server comprises: identifying the SNI from the client hello message during a handshaking process for setting the secure connection between the client and the remote server; extracting a domain identified in a public certificate sent from the remote server to the client; comparing the destination domain and the domain identified in the public certificate; and in the event that the destination domain matches the domain

Claim	U.S. Patent No. 9,917,856	Disclosure from Buruganahalli
		identified in the public certificate, applying a security policy based on the destination domain to filter traffic at the security device, wherein the security policy includes a whitelist/blacklist policy.
1.1	receiving, by a packet-filtering system comprising a hardware processor and a memory and configured to filter packets in accordance with a plurality of packet-filtering rules, data indicating a plurality of network-threat indicators, wherein at least one of the plurality of network-threat indicators comprises a domain name identified as a network threat;	<i>See, e.g., Buruganahalli at Abstract:</i> Techniques for destination domain extraction for secure protocols are disclosed. In some embodiments, destination domain extraction for secure protocols includes monitoring network communications between a client and a remote server; determining if the client sends a request to create a secure connection with the remote server (e.g., in which the network communications are initiating a setup for a secure protocol-based connection); and extracting a destination domain from the request to create the secure connection with the remote server. In some embodiments, the secure protocol is a secure sockets layer (SSL) protocol or transport layer security (TLS) protocol, and the destination domain is extracted from the server name indication (SNI) of a client hello message sent from the client to the remote server. In some embodiments, destination domain extraction for secure protocols further includes applying a policy (e.g., a security policy) based on the destination domain to filter traffic using a security device. <i>See, e.g., Buruganahalli at 1:1-11:</i> The invention can be implemented in numerous ways, including as

Claim	U.S. Patent No. 9,917,856	Disclosure from Buruganahalli
		a process; an apparatus; a system; a composition of matter; a computer program product embodied on a computer readable storage medium; and/or a processor, such as a processor configured to execute instructions stored on and/or provided by a memory coupled to the processor. <i>See, e.g., Buruganahalli at 5:14-41:</i> In some embodiments, the SNI standard extension is used as disclosed herein as a mechanism to facilitate destination domain extraction for secure protocols without requiring decryption. For example, a firewall can intercept and monitor data communications from a client and a remote server in order to extract the hostname data from the SNI field (e.g., server_name field of the SNI extension), which is sent as an unencrypted data communication from the client (e.g., a web browser executing on the client) to the remote server (e.g., at the start of the handshaking process for setting up a secure TLS communication channel/session between a client and a remote server). This extracted hostname data (e.g., destination domain) can then be used by the firewall to perform an action(s) based on a firewall policy (e.g., the domain name can be extracted by parsing the client hello message to extract the host name from the server_name field of the SNI extension, which can then be used to apply firewall policies or take responsive actions based on this information without having to wait for data to be decrypted), as further described herein with respect to various embodiments. Also, such SNI information is available in the first packet of this handshaking process, such that a policy (e.g., a security or firewall policy using the hostname information extracted from the SNI information) can be applied sooner based on this early extraction of the hostname from the SNI information such that the rest of the stream can be forwarded without any

Claim	U.S. Patent No. 9,917,856	Disclosure from Buruganahalli
		decryption resulting in less state to track thereby enhancing performance (e.g., of the security device, such as a firewall). <i>See, e.g.,</i> Buruganahalli at 6:66-7:15: In some embodiments, destination domain extraction for secure protocols includes monitoring network communications between a client and a remote server; determining if the client sends a request to create a secure connection with the remote server (e.g., in which the network communications are initiating a setup for a secure protocol-based connection); and extracting a destination domain from the request to create the secure connection with the remote server. In some embodiments, the secure protocol is a secure sockets layer (SSL) protocol or transport layer security (TLS) protocol, and the destination domain is extracted from the server name indication (SNI) of a client hello message sent from the client to the remote server (e.g., the client hello message is an unencrypted communication). In some embodiments, destination domain extraction for secure protocols further includes applying a policy (e.g., a security policy) based on the destination domain to filter traffic at a security device. <i>See, e.g.,</i> Buruganahalli at 7:62-8:8: In some embodiments, destination domain extraction for secure protocols further includes decrypting the monitored encrypted network communications of a session associated with the network communications between the client and the remote server to verify that the destination domain name extracted from a server name indication matches the domain associated with a URL for the session associated with the network communications between the client and the remote server. For example, this technique can be

Claim	U.S. Patent No. 9,917,856	Disclosure from Buruganahalli
		used in order to detect malware that attempts to fool a hosting site by including false destination domain information (e.g., in the SNI field in a client hello message of an initial set-up of a TLS session). <i>See, e.g., Buruganahalli at 8:9-28:</i> In some embodiments, destination domain extraction for secure protocols further includes intercepting a request to establish an encrypted session from the client to the remote server; sending a request to establish the encrypted session on behalf of the client to the remote server; sending an encrypted session response to the client on behalf of the remote server using a session key associated with the firewall device; decrypting encrypted traffic between the client and the remote server to monitor for a request from the client to create a tunnel using the first protocol with the remote server; allowing the request to create the tunnel; monitoring decrypted session traffic between the client and the remote server over the tunnel based on one or more firewall policies; and blocking the session traffic if a violation of a first firewall policy is determined, in which the first firewall policy includes a policy for verifying that the destination domain name extracted from a server name indication matches the domain associated with a URL for a session associated with the network communications between the client and the remote server. <i>See, e.g., Buruganahalli at 9:59-64:</i> In some embodiments, the results of the various traffic monitoring techniques using known protocol decoder engine 112, identified traffic engine 114, and unknown protocol decoder engine 116 described above are provided to report and enforce policies engine 120 (e.g., network/routing policies, security policies, and/or

Claim	U.S. Patent No. 9,917,856	Disclosure from Buruganahalli
		firewall policies). <i>See, e.g.,</i> Buruganahalli at 11:50-12:11: FIG. 3B is another functional block diagram illustrating the use of a firewall for destination domain extraction for secure protocols in accordance with some embodiments. As shown, a client 312 attempts to establish an SSL session with a remote server 316. The client 312 can attempt to use, for example, a secure tunnel with the server 316 to access a network service 318, which can be a network service activity that is in violation of one or more firewall policies/rules implemented by a firewall device 314. However, in this case, the SSL session request is intercepted and detected by the firewall 314. In response, the firewall 314 performs a trusted man-in-the-middle technique by effectively splitting the SSL session between the client 312 and the remote server 316 into two half sessions shown as Session A and Session B in FIG. 38. In Session A, the firewall 314 acts as the remote server 316 such that it is transparent to the client 312 that it is not communicating directly with the remote server 316. Session A traffic is encrypted using the session key 51 associated with the firewall device. In Session B, the firewall 314 acts as the client 312 such that it is transparent to the remote server 316 that it is not communicating directly with the client 312. Session B traffic is encrypted using the session key S2 associated with the firewall device (e.g., the firewall device can store the fingerprint from the remote server in association with that remote server IP address). After the session set-up handshaking is completed for each of Session A and Session B, any data that is communicated from the client 312 to the firewall 314 is decrypted using a session key 51 and is then inspected by the firewall 314.

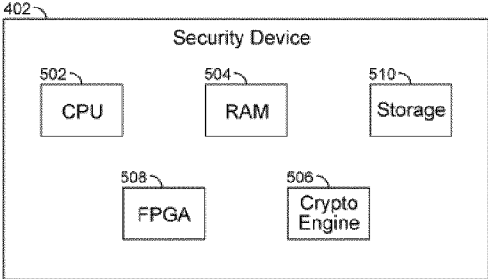
Claim	U.S. Patent No. 9,917,856	Disclosure from Buruganahalli
		<i>See, e.g.,</i> Buruganahalli at 14:45-62: FIG. 7 is a flow diagram for providing destination domain extraction for secure protocols in accordance with some embodiments. At 702, monitoring network communications between a client and a remote server is performed. At 704, determining if the client sends a request to create a secure connection with the remote server is performed (e.g., in which the network communications are initiating a setup for a secure protocol-based connection). At 706, extracting a destination domain from the request to create the secure connection with the remote server is performed. In some embodiments, the secure protocol is a secure sockets layer (SSL) protocol or transport layer security (TLS) protocol, and the destination domain is extracted from the server name indication (SNI) of a client hello message sent from the client to the remote server. In some embodiments, destination domain extraction for secure protocols further includes applying a policy (e.g., a security policy) based on the destination domain to filter traffic using a security device. <i>See, e.g.,</i> Buruganahalli at claim 1: A system for destination domain extraction for secure protocols, comprising: a security device; a processor configured to: monitor network communications between a client and a remote server;

Claim	U.S. Patent No. 9,917,856	Disclosure from Buruganahalli
		determine if the client sends a request to create a secure connection with the remote server, wherein the secure connection utilizes a secure protocol, the secure protocol being a secure sockets layer (SSL) protocol or transport layer security (TLS) protocol; and extract a destination domain from the request to create the secure connection with the remote server, comprising: before the secure connection is created, extract the destination domain from a server name indication (SNI) of a client hello message sent from the client to the remote server, the secure connection including an encrypted message, wherein the extracting of the destination domain from the SNI of the client hello message sent from the client to the remote server comprises to: identify the SNI from the client hello message during a handshaking process for setting the secure connection between the client and the remote server; extract a domain identified in a public certificate sent from the remote server to the client; compare the destination domain and the domain identified in the public certificate; and in the event that the destination domain matches the domain identified in the public certificate, apply a security policy based on the destination domain to filter traffic at the security device, wherein the security policy includes a whitelist/blacklist policy; and a memory coupled to the processor and configured to provide the

Claim	U.S. Patent No. 9,917,856	Disclosure from Buruganahalli
		processor with instructions. <i>See, e.g.,</i> Buruganahalli at claim 3: The system recited in claim 1, wherein the security policy includes a malware detection policy. <i>See, e.g.,</i> Buruganahalli at claim 4: The system recited in claim 1, wherein the security policy includes a uniform resource locator (URL)/category filtering related policy <i>See, e.g.,</i> Buruganahalli at claim 7: The system recited in claim 1, wherein a session associated with the network communications between the client and the remote server are decrypted to verify that the destination domain extracted from a server name indication (SNI) matches a domain associated with a uniform resource locator (URL) for the session associated with the network communications between the client and the remote server. <i>See, e.g.,</i> Buruganahalli at claim 8: The system recited in claim 1, wherein the processor is further configured to: decrypt a monitored encrypted network communications of a session associated with the network communications between the client and the remote server to verify that the destination domain extracted from a server name indication (SNI) matches a domain associated with a uniform resource locator (URL) for the session associated with the network communications between the client and the remote server.

Claim	U.S. Patent No. 9,917,856	Disclosure from Buruganahalli
		<i>See, e.g.,</i> Buruganahalli at claim 11: A method for destination domain extraction for secure protocols, comprising: monitoring network communications between a client and a remote server using a security device; determining if the client sends a request to create a secure connection with the remote server, wherein the secure connection utilizes a secure protocol, the secure protocol being a secure sockets layer (SSL) protocol or transport layer security (TLS) protocol; and extracting a destination domain from the request to create the secure connection with the remote server, comprising: before the secure connection is created, extracting the destination domain from a server name indication (SNI) of a client hello message sent from the client to the remote server, the secure connection including an encrypted message, wherein the extracting of the destination domain from the SNI of the client hello message sent from the client to the remote server comprises: identifying the SNI from the client hello message during a handshaking process for setting the secure connection between the client and the remote server; extracting a domain identified in a public certificate sent from the remote server to the client; comparing the destination domain and the domain identified in the

Claim	U.S. Patent No. 9,917,856	Disclosure from Buruganahalli
		public certificate; and in the event that the destination domain matches the domain identified in the public certificate, applying a security policy based on the destination domain to filter traffic at the security device, wherein the security policy includes a whitelist/blacklist policy. See, e.g., Buruganahalli at Fig. 1: <pre> graph TD 102[Start] --> 104[Check IP/Port] 104 --> 106[Policy Check] 106 --> 108[App Signature Check] 108 --> 110[Decrypt (SSL/SSH)] 108 --> 112[Known Protocol Decoder] 110 --> 114[Identified Traffic (No Decoding)] 112 --> 116[Unknown Protocol Decoder] 114 --> 122[Content-ID] 116 --> 122 122 --> 120[Report and Enforce Policy] </pre> FIG. 1 See, e.g., Buruganahalli at Fig. 5:

Claim	U.S. Patent No. 9,917,856	Disclosure from Buruganahalli
		 FIG. 5
1.2	identifying packets comprising unencrypted data;	<i>See, e.g., Buruganahalli at 4:24-35:</i> While performing various decryption techniques as described herein can facilitate applying filtering based on a policy (e.g., a security policy, such as a firewall policy), it would be desirable and efficient to further facilitate policy- based filtering on such data communication flows (e.g., sessions) prior to the set-up of the encrypted data communication for such flows (e.g., to monitor the initial unencrypted/ clear text data communications exchanged as part of an initial handshake to setup a secure connection for a new session between a client and a remote server, which is prior to the establishment of the secure communication channel/tunnel after which any data communications intercepted between the client and the remote server would be encrypted). <i>See, e.g., Buruganahalli at 5:14-41:</i> In some embodiments, the SNI standard extension is used as disclosed herein as a mechanism to facilitate destination domain

Claim	U.S. Patent No. 9,917,856	Disclosure from Buruganahalli
		extraction for secure protocols without requiring decryption. For example, a firewall can intercept and monitor data communications from a client and a remote server in order to extract the hostname data from the SNI field (e.g., server_name field of the SNI extension), which is sent as an unencrypted data communication from the client (e.g., a web browser executing on the client) to the remote server (e.g., at the start of the handshaking process for setting up a secure TLS communication channel/session between a client and a remote server). This extracted hostname data (e.g., destination domain) can then be used by the firewall to perform an action(s) based on a firewall policy (e.g., the domain name can be extracted by parsing the client hello message to extract the host name from the server_name field of the SNI extension, which can then be used to apply firewall policies or take responsive actions based on this information without having to wait for data to be decrypted), as further described herein with respect to various embodiments. Also, such SNI information is available in the first packet of this handshaking process, such that a policy (e.g., a security or firewall policy using the hostname information extracted from the SNI information) can be applied sooner based on this early extraction of the hostname from the SNI information such that the rest of the stream can be forwarded without any decryption resulting in less state to track thereby enhancing performance (e.g., of the security device, such as a firewall). <i>See, e.g., Buruganahalli at 6:66-7:15:</i> In some embodiments, destination domain extraction for secure protocols includes monitoring network communications between a client and a remote server; determining if the client sends a request to create a secure connection with the remote server (e.g., in which the network communications are initiating a setup for a secure

Claim	U.S. Patent No. 9,917,856	Disclosure from Buruganahalli
		protocol-based connection); and extracting a destination domain from the request to create the secure connection with the remote server. In some embodiments, the secure protocol is a secure sockets layer (SSL) protocol or transport layer security (TLS) protocol, and the destination domain is extracted from the server name indication (SNI) of a client hello message sent from the client to the remote server (e.g., the client hello message is an unencrypted communication). In some embodiments, destination domain extraction for secure protocols further includes applying a policy (e.g., a security policy) based on the destination domain to filter traffic at a security device. <i>See, e.g., Buruganahalli at 7:62-8:8:</i> In some embodiments, destination domain extraction for secure protocols further includes decrypting the monitored encrypted network communications of a session associated with the network communications between the client and the remote server to verify that the destination domain name extracted from a server name indication matches the domain associated with a URL for the session associated with the network communications between the client and the remote server. For example, this technique can be used in order to detect malware that attempts to fool a hosting site by including false destination domain information (e.g., in the SNI field in a client hello message of an initial set-up of a TLS session). <i>See, e.g., Buruganahalli at 8:9-28:</i> In some embodiments, destination domain extraction for secure protocols further includes intercepting a request to establish an encrypted session from the client to the remote server; sending a request to establish the encrypted session on behalf of the client to

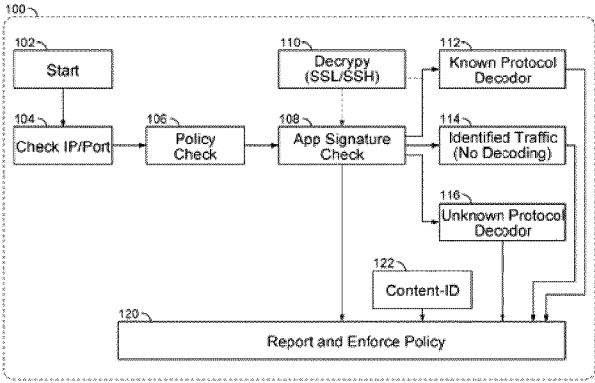
Claim	U.S. Patent No. 9,917,856	Disclosure from Buruganahalli
		the remote server; sending an encrypted session response to the client on behalf of the remote server using a session key associated with the firewall device; decrypting encrypted traffic between the client and the remote server to monitor for a request from the client to create a tunnel using the first protocol with the remote server; allowing the request to create the tunnel; monitoring decrypted session traffic between the client and the remote server over the tunnel based on one or more firewall policies; and blocking the session traffic if a violation of a first firewall policy is determined, in which the first firewall policy includes a policy for verifying that the destination domain name extracted from a server name indication matches the domain associated with a URL for a session associated with the network communications between the client and the remote server. <i>See, e.g., Buruganahalli at 10:36-55:</i> FIG. 3A is a functional block diagram illustrating an SSL/ TLS session passing through a firewall that can implement destination domain extraction for secure protocols in accordance with some embodiments. In that session, a typical firewall can initially just inspect the handshake traffic (e.g., the initial negotiation to setup a secure communication channel/tunnel between a client and a remote server), as all the application data is encrypted. As shown, a client 302 establishes a secure tunnel session (e.g., creates an SSL tunnel) with a remote server 306. The client 302 can use the secure tunnel with the server 306 to access a network service 308, which can be a network service activity that is in violation of one or more firewall policies/rules implemented by the fire- wall device 304 (e.g., a policy that includes requirements or rules related to destination domains that may be used for secure protocol communications). As the secure tunnel ses- sion traffic is encrypted

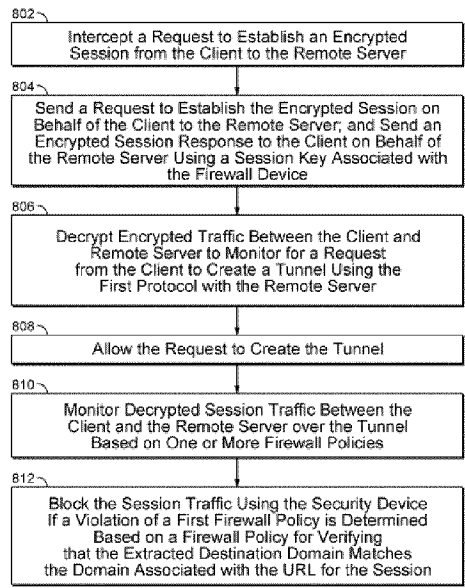
Claim	U.S. Patent No. 9,917,856	Disclosure from Buruganahalli
		and passes through a firewall 304, the firewall 304 typically cannot decrypt the encrypted secure tunnel session traffic and, thus, cannot detect such firewall policy/rule violation(s). <i>See, e.g.,</i> Buruganahalli at 11:50-12:11: FIG. 3B is another functional block diagram illustrating the use of a firewall for destination domain extraction for secure protocols in accordance with some embodiments. As shown, a client 312 attempts to establish an SSL session with a remote server 316. The client 312 can attempt to use, for example, a secure tunnel with the server 316 to access a network service 318, which can be a network service activity that is in violation of one or more firewall policies/rules implemented by a firewall device 314. However, in this case, the SSL session request is intercepted and detected by the firewall 314. In response, the firewall 314 performs a trusted man-in-the-middle technique by effectively splitting the SSL session between the client 312 and the remote server 316 into two half sessions shown as Session A and Session B in FIG. 38. In Session A, the firewall 314 acts as the remote server 316 such that it is transparent to the client 312 that it is not communicating directly with the remote server 316. Session A traffic is encrypted using the session key 51 associated with the firewall device. In Session B, the firewall 314 acts as the client 312 such that it is transparent to the remote server 316 that it is not communicating directly with the client 312. Session B traffic is encrypted using the session key S2 associated with the firewall device (e.g., the firewall device can store the fingerprint from the remote server in association with that remote server IP address). After the session set-up handshaking is completed for each of Session A and Session B, any data that is communicated from the client 312 to the firewall 314 is decrypted

Claim	U.S. Patent No. 9,917,856	Disclosure from Buruganahalli
		using a session key 51 and is then inspected by the firewall 314. <i>See, e.g.,</i> Buruganahalli at 14:45-62: FIG. 7 is a flow diagram for providing destination domain extraction for secure protocols in accordance with some embodiments. At 702, monitoring network communications between a client and a remote server is performed. At 704, determining if the client sends a request to create a secure connection with the remote server is performed (e.g., in which the network communications are initiating a setup for a secure protocol-based connection). At 706, extracting a destination domain from the request to create the secure connection with the remote server is performed. In some embodiments, the secure protocol is a secure sockets layer (SSL) protocol or transport layer security (TLS) protocol, and the destination domain is extracted from the server name indication (SNI) of a client hello message sent from the client to the remote server. In some embodiments, destination domain extraction for secure protocols further includes applying a policy (e.g., a security policy) based on the destination domain to filter traffic using a security device. <i>See, e.g.,</i> Buruganahalli at claim 1: A system for destination domain extraction for secure protocols, comprising: a security device; a processor configured to: monitor network communications between a client and a remote

Claim	U.S. Patent No. 9,917,856	Disclosure from Buruganahalli
		server; determine if the client sends a request to create a secure connection with the remote server, wherein the secure connection utilizes a secure protocol, the secure protocol being a secure sockets layer (SSL) protocol or transport layer security (TLS) protocol; and extract a destination domain from the request to create the secure connection with the remote server, comprising: before the secure connection is created, extract the destination domain from a server name indication (SNI) of a client hello message sent from the client to the remote server, the secure connection including an encrypted message, wherein the extracting of the destination domain from the SNI of the client hello message sent from the client to the remote server comprises to: identify the SNI from the client hello message during a handshaking process for setting the secure connection between the client and the remote server; extract a domain identified in a public certificate sent from the remote server to the client; compare the destination domain and the domain identified in the public certificate; and in the event that the destination domain matches the domain identified in the public certificate, apply a security policy based on the destination domain to filter traffic at the security device, wherein the security policy includes a whitelist/blacklist policy;

Claim	U.S. Patent No. 9,917,856	Disclosure from Buruganahalli
		and a memory coupled to the processor and configured to provide the processor with instructions. <i>See, e.g.,</i> Buruganahalli at claim 7: The system recited in claim 1, wherein a session associated with the network communications between the client and the remote server are decrypted to verify that the destination domain extracted from a server name indication (SNI) matches a domain associated with a uniform resource locator (URL) for the session associated with the network communications between the client and the remote server. <i>See, e.g.,</i> Buruganahalli at claim 8: The system recited in claim 1, wherein the processor is further configured to: decrypt a monitored encrypted network communications of a session associated with the network communications between the client and the remote server to verify that the destination domain extracted from a server name indication (SNI) matches a domain associated with a uniform resource locator (URL) for the session associated with the network communications between the client and the remote server. <i>See, e.g.,</i> Buruganahalli at Fig. 1:

Claim	U.S. Patent No. 9,917,856	Disclosure from Buruganahalli
		 FIG. 1 <i>See, e.g., Buruganahalli at Fig. 8:</i>

Claim	U.S. Patent No. 9,917,856	Disclosure from Buruganahalli
		 <pre> graph TD 802[802 Intercept a Request to Establish an Encrypted Session from the Client to the Remote Server] --> 804[804 Send a Request to Establish the Encrypted Session on Behalf of the Client to the Remote Server; and Send an Encrypted Session Response to the Client on Behalf of the Remote Server Using a Session Key Associated with the Firewall Device] 804 --> 806[806 Decrypt Encrypted Traffic Between the Client and Remote Server to Monitor for a Request from the Client to Create a Tunnel Using the First Protocol with the Remote Server] 806 --> 808[808 Allow the Request to Create the Tunnel] 808 --> 810[810 Monitor Decrypted Session Traffic Between the Client and the Remote Server over the Tunnel Based on One or More Firewall Policies] 810 --> 812[812 Block the Session Traffic Using the Security Device If a Violation of a First Firewall Policy is Determined Based on a Firewall Policy for Verifying that the Extracted Destination Domain Matches the Domain Associated with the URL for the Session] </pre> FIG. 8
1.3	identifying packets comprising encrypted data;	<i>See, e.g.,</i> Burugunahalli at 4:24-35: While performing various decryption techniques as described herein can facilitate applying filtering based on a policy (e.g., a

Claim	U.S. Patent No. 9,917,856	Disclosure from Buruganahalli
		security policy, such as a firewall policy), it would be desirable and efficient to further facilitate policy- based filtering on such data communication flows (e.g., sessions) prior to the set-up of the encrypted data communication for such flows (e.g., to monitor the initial unencrypted/ clear text data communications exchanged as part of an initial handshake to setup a secure connection for a new session between a client and a remote server, which is prior to the establishment of the secure communication channel/tunnel after which any data communications intercepted between the client and the remote server would be encrypted). <i>See, e.g., Buruganahalli at 5:14-41:</i> In some embodiments, the SNI standard extension is used as disclosed herein as a mechanism to facilitate destination domain extraction for secure protocols without requiring decryption. For example, a firewall can intercept and monitor data communications from a client and a remote server in order to extract the hostname data from the SNI field (e.g., server_name field of the SNI extension), which is sent as an unencrypted data communication from the client (e.g., a web browser executing on the client) to the remote server (e.g., at the start of the handshaking process for setting up a secure TLS communication channel/session between a client and a remote server). This extracted hostname data (e.g., destination domain) can then be used by the firewall to perform an action(s) based on a firewall policy (e.g., the domain name can be extracted by parsing the client hello message to extract the host name from the server_name field of the SNI extension, which can then be used to apply firewall policies or take responsive actions based on this information without having to wait for data to be decrypted), as further described herein with respect to various embodiments. Also, such SNI information is available in the first

Claim	U.S. Patent No. 9,917,856	Disclosure from Buruganahalli
		packet of this handshaking process, such that a policy (e.g., a security or firewall policy using the hostname information extracted from the SNI information) can be applied sooner based on this early extraction of the hostname from the SNI information such that the rest of the stream can be forwarded without any decryption resulting in less state to track thereby enhancing performance (e.g., of the security device, such as a firewall). <i>See, e.g., Buruganahalli at 6:66-7:15:</i> In some embodiments, destination domain extraction for secure protocols includes monitoring network communications between a client and a remote server; determining if the client sends a request to create a secure connection with the remote server (e.g., in which the network communications are initiating a setup for a secure protocol-based connection); and extracting a destination domain from the request to create the secure connection with the remote server. In some embodiments, the secure protocol is a secure sockets layer (SSL) protocol or transport layer security (TLS) protocol, and the destination domain is extracted from the server name indication (SNI) of a client hello message sent from the client to the remote server (e.g., the client hello message is an unencrypted communication). In some embodiments, destination domain extraction for secure protocols further includes applying a policy (e.g., a security policy) based on the destination domain to filter traffic at a security device. <i>See, e.g., Buruganahalli at 7:25-36:</i> In some embodiments, destination domain extraction for secure protocols further includes applying a security policy based on the destination domain to filter traffic at the security device. For

Claim	U.S. Patent No. 9,917,856	Disclosure from Buruganahalli
		example, the security policy can include a malware detection policy, a whitelist/blacklist policy (e.g., in which a session associated with the network communications between the client and the remote server are not decrypted if the destination domain is included in the whitelist, and a session associated with the network communications between the client and the remote server are decrypted if the destination domain is included in the blacklist), and/or a uniform resource locator (URL)/category filtering policy. <i>See, e.g., Buruganahalli at 7:45-52:</i> In some embodiments, destination domain extraction for secure protocols further includes applying a security policy based on the destination domain to filter traffic at the security device, in which the security policy includes a whitelist/blacklist policy, and in which a session associated with the network communications between the client and the remote server are decrypted if the destination domain is included in a blacklist of the whitelist/blacklist policy. <i>See, e.g., Buruganahalli at 7:62-8:8:</i> In some embodiments, destination domain extraction for secure protocols further includes decrypting the monitored encrypted network communications of a session associated with the network communications between the client and the remote server to verify that the destination domain name extracted from a server name indication matches the domain associated with a URL for the session associated with the network communications between the client and the remote server. For example, this technique can be used in order to detect malware that attempts to fool a hosting site by including false destination domain information (e.g., in the SNI

Claim	U.S. Patent No. 9,917,856	Disclosure from Buruganahalli
		field in a client hello message of an initial set-up of a TLS session). <i>See, e.g.,</i> Buruganahalli at 8:9-28: In some embodiments, destination domain extraction for secure protocols further includes intercepting a request to establish an encrypted session from the client to the remote server; sending a request to establish the encrypted session on behalf of the client to the remote server; sending an encrypted session response to the client on behalf of the remote server using a session key associated with the firewall device; decrypting encrypted traffic between the client and the remote server to monitor for a request from the client to create a tunnel using the first protocol with the remote server; allowing the request to create the tunnel; monitoring decrypted session traffic between the client and the remote server over the tunnel based on one or more firewall policies; and blocking the session traffic if a violation of a first firewall policy is determined, in which the first firewall policy includes a policy for verifying that the destination domain name extracted from a server name indication matches the domain associated with a URL for a session associated with the network communications between the client and the remote server. <i>See, e.g.,</i> Buruganahalli at 8:29-38: In some embodiments, destination domain extraction for secure protocols further includes monitoring encrypted network communications between a client and a remote server (e.g., using a security device, such as a firewall to provide man-in-the-middle decryption techniques, such as described herein), in which the encrypted network communications are encrypted using a first protocol (e.g., SSL/TLS or another protocol for encrypted network

Claim	U.S. Patent No. 9,917,856	Disclosure from Buruganahalli
		communications); and per- forming an action based on the decrypted network communications between the client and the remote server. <i>See, e.g.,</i> Buruganahalli at 9:7-25: As shown in FIG. 1, network traffic monitoring begins at 102. An IP address and port engine 104 determines an IP address and port number for a monitored traffic flow (e.g., a session) based on packet analysis. In some embodiments, user identification is then determined (e.g., user ID can be deduced based on the source IP address). A policy check engine 106 determines whether any policies can be applied based on the IP address and port number. As also shown in FIG. 1, an application signature check engine 108 identifies an application (e.g., using an APP-ID engine using various application signatures for identifying applications based on packet flow analysis). For example, APP-ID engine 108 can be configured to determine what type of traffic the session involves, such as HTTP traffic, HTTPS traffic, SSL/TLS traffic, SSH traffic, DNS requests, FTP traffic, unknown traffic, and various other types of traffic, and such classified traffic can be directed to an appropriate decoder, such as decoders 112, 114, and 116, to decode the classified traffic for each monitored session's traffic flow. <i>See, e.g.,</i> Buruganahalli at 9:42-57: As further illustrated in FIG. 1, if the monitored traffic is encrypted (e.g., encrypted using HTTPS, SSL/TLS, SSH, or another known encryption protocol), then the security device can be configured (e.g., based on a firewall policy) to decrypt the monitored traffic using a decrypt engine 110 (e.g., applying trusted man-in-the-middle techniques using a self-signed certificate). For example, in

Claim	U.S. Patent No. 9,917,856	Disclosure from Buruganahalli
		some cases, if the extracted destination domain is not trusted or unknown, then the firewall can be configured to decrypt the monitored traffic for further traffic monitoring and inspection based on a policy (e.g., a firewall policy). As another example, the firewall can be configured to decrypt the monitored traffic to verify that the extracted hostname (e.g., extracted from a client hello message) matches the domain that is being communicated with using the encrypted communications between the client and the remote server/external site. <i>See, e.g., Buruganahalli at 9:59-64:</i> In some embodiments, the results of the various traffic monitoring techniques using known protocol decoder engine 112, identified traffic engine 114, and unknown protocol decoder engine 116 described above are provided to report and enforce policies engine 120 (e.g., network/routing policies, security policies, and/or firewall policies). <i>See, e.g., Buruganahalli at 10:36-55:</i> FIG. 3A is a functional block diagram illustrating an SSL/ TLS session passing through a firewall that can implement destination domain extraction for secure protocols in accordance with some embodiments. In that session, a typical firewall can initially just inspect the handshake traffic (e.g., the initial negotiation to setup a secure communication channel/tunnel between a client and a remote server), as all the application data is encrypted. As shown, a client 302 establishes a secure tunnel session (e.g., creates an SSL tunnel) with a remote server 306. The client 302 can use the secure tunnel with the server 306 to access a network service 308, which can be a network service activity that is in violation of one or more

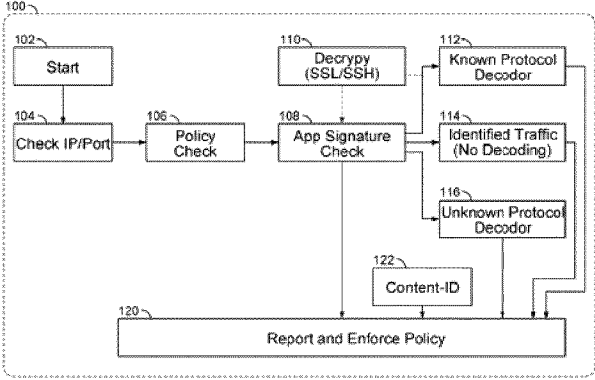
Claim	U.S. Patent No. 9,917,856	Disclosure from Buruganahalli
		firewall policies/rules implemented by the fire- wall device 304 (e.g., a policy that includes requirements or rules related to destination domains that may be used for secure protocol communications). As the secure tunnel session traffic is encrypted and passes through a firewall 304, the firewall 304 typically cannot decrypt the encrypted secure tunnel session traffic and, thus, cannot detect such firewall policy/rule violation(s). <i>See, e.g., Buruganahalli at 11:38-49:</i> In addition, various techniques for intercepting and decoding encrypted tunneling communications from a client to a remote server are disclosed herein. For example, a trusted man-in-the-middle technique that can be used for intercepting and decoding encrypted tunneling communications to further monitor such traffic in clear text (e.g., unencrypted) is described below with respect to FIG. 38. These decryption techniques can be used in addition to and/or in combination with the various techniques described herein for destination domain extraction for secure protocols to further enhance the various security and firewall related techniques disclosed herein. <i>See, e.g., Buruganahalli at 11:50-12:11:</i> FIG. 3B is another functional block diagram illustrating the use of a firewall for destination domain extraction for secure protocols in accordance with some embodiments. As shown, a client 312 attempts to establish an SSL session with a remote server 316. The client 312 can attempt to use, for example, a secure tunnel with the server 316 to access a network service 318, which can be a network service activity that is in violation of one or more firewall policies/rules implemented by a firewall device 314. However, in

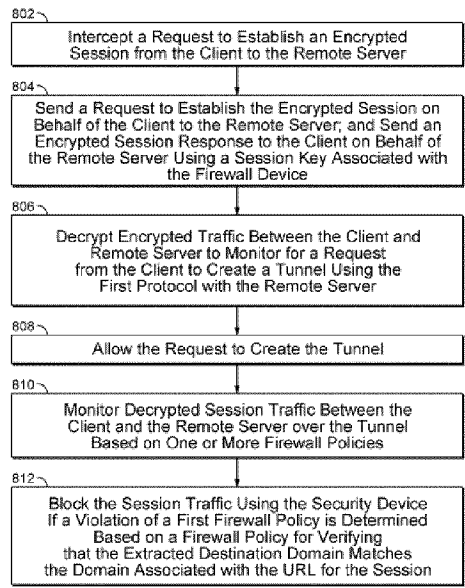
Claim	U.S. Patent No. 9,917,856	Disclosure from Buruganahalli
		this case, the SSL session request is intercepted and detected by the firewall 314. In response, the firewall 314 performs a trusted man-in-the-middle technique by effectively splitting the SSL session between the client 312 and the remote server 316 into two half sessions shown as Session A and Session B in FIG. 38. In Session A, the firewall 314 acts as the remote server 316 such that it is transparent to the client 312 that it is not communicating directly with the remote server 316. Session A traffic is encrypted using the session key 51 associated with the firewall device. In Session B, the firewall 314 acts as the client 312 such that it is transparent to the remote server 316 that it is not communicating directly with the client 312. Session B traffic is encrypted using the session key S2 associated with the firewall device (e.g., the firewall device can store the fingerprint from the remote server in association with that remote server IP address). After the session set-up handshaking is completed for each of Session A and Session B, any data that is communicated from the client 312 to the firewall 314 is decrypted using a session key 51 and is then inspected by the firewall 314. <i>See, e.g., Buruganahalli at 12:34-42:</i> Thus, using these techniques the firewall 314 is between the client 312 and the server 316 and can present itself as server of the session to the client and can present itself as the client of the session to the server. In this manner, the firewall 314 can inspect decrypted traffic and re-encrypt it before forwarding it to the other party. During inspection, if encrypted tunneling traffic is detected, the entire session can either be tom down or various other responsive actions can be performed as described herein. <i>See, e.g., Buruganahalli at 14:45-62:</i>

Claim	U.S. Patent No. 9,917,856	Disclosure from Buruganahalli
		FIG. 7 is a flow diagram for providing destination domain extraction for secure protocols in accordance with some embodiments. At 702, monitoring network communications between a client and a remote server is performed. At 704, determining if the client sends a request to create a secure connection with the remote server is performed (e.g., in which the network communications are initiating a setup for a secure protocol-based connection). At 706, extracting a destination domain from the request to create the secure connection with the remote server is performed. In some embodiments, the secure protocol is a secure sockets layer (SSL) protocol or transport layer security (TLS) protocol, and the destination domain is extracted from the server name indication (SNI) of a client hello message sent from the client to the remote server. In some embodiments, destination domain extraction for secure protocols further includes applying a policy (e.g., a security policy) based on the destination domain to filter traffic using a security device. <i>See, e.g., Buruganahalli at claim 1:</i> A system for destination domain extraction for secure protocols, comprising: a security device; a processor configured to: monitor network communications between a client and a remote server; determine if the client sends a request to create a secure connection with the remote server, wherein the secure connection utilizes a

Claim	U.S. Patent No. 9,917,856	Disclosure from Buruganahalli
		secure protocol, the secure protocol being a secure sockets layer (SSL) protocol or transport layer security (TLS) protocol; and extract a destination domain from the request to create the secure connection with the remote server, comprising: before the secure connection is created, extract the destination domain from a server name indication (SNI) of a client hello message sent from the client to the remote server, the secure connection including an encrypted message, wherein the extracting of the destination domain from the SNI of the client hello message sent from the client to the remote server comprises to: identify the SNI from the client hello message during a handshaking process for setting the secure connection between the client and the remote server; extract a domain identified in a public certificate sent from the remote server to the client; compare the destination domain and the domain identified in the public certificate; and in the event that the destination domain matches the domain identified in the public certificate, apply a security policy based on the destination domain to filter traffic at the security device, wherein the security policy includes a whitelist/blacklist policy; and a memory coupled to the processor and configured to provide the processor with instructions.

Claim	U.S. Patent No. 9,917,856	Disclosure from Buruganahalli
		<i>See, e.g.,</i> Buruganahalli at claim 7: The system recited in claim 1, wherein a session associated with the network communications between the client and the remote server are decrypted to verify that the destination domain extracted from a server name indication (SNI) matches a domain associated with a uniform resource locator (URL) for the session associated with the network communications between the client and the remote server. <i>See, e.g.,</i> Buruganahalli at claim 8: The system recited in claim 1, wherein the processor is further configured to: decrypt a monitored encrypted network communications of a session associated with the network communications between the client and the remote server to verify that the destination domain extracted from a server name indication (SNI) matches a domain associated with a uniform resource locator (URL) for the session associated with the network communications between the client and the remote server. <i>See, e.g.,</i> Buruganahalli at Fig. 1:

Claim	U.S. Patent No. 9,917,856	Disclosure from Buruganahalli
		 FIG. 1 <i>See, e.g., Buruganahalli at Fig. 8:</i>

Claim	U.S. Patent No. 9,917,856	Disclosure from Buruganahalli
		 <pre> graph TD 802[802 Intercept a Request to Establish an Encrypted Session from the Client to the Remote Server] --> 804[804 Send a Request to Establish the Encrypted Session on Behalf of the Client to the Remote Server; and Send an Encrypted Session Response to the Client on Behalf of the Remote Server Using a Session Key Associated with the Firewall Device] 804 --> 806[806 Decrypt Encrypted Traffic Between the Client and Remote Server to Monitor for a Request from the Client to Create a Tunnel Using the First Protocol with the Remote Server] 806 --> 808[808 Allow the Request to Create the Tunnel] 808 --> 810[810 Monitor Decrypted Session Traffic Between the Client and the Remote Server over the Tunnel Based on One or More Firewall Policies] 810 --> 812[812 Block the Session Traffic Using the Security Device If a Violation of a First Firewall Policy is Determined Based on a Firewall Policy for Verifying that the Extracted Destination Domain Matches the Domain Associated with the URL for the Session] </pre> FIG. 8
1.4	determining, by the packet-filtering system and based on a portion of the unencrypted data corresponding to one or more network-threat indicators of the plurality of network-threat indicators, packets comprising encrypted data that corresponds to the one or more network-threat indicators;	<i>See, e.g., Burugunahalli at 4:24-35:</i> While performing various decryption techniques as described herein can facilitate applying filtering based on a policy (e.g., a security policy, such as a firewall policy), it would be desirable and efficient to further facilitate policy- based filtering on such data communication flows (e.g., sessions) prior to the set-up of the encrypted data communication for such flows (e.g., to monitor the initial unencrypted/ clear text data communications exchanged as part of an initial handshake to setup a secure connection for a new session between a client and a remote server, which is prior to the establishment of the secure communication channel/tunnel after

Claim	U.S. Patent No. 9,917,856	Disclosure from Buruganahalli
		which any data communications intercepted between the client and the remote server would be encrypted). <i>See, e.g.,</i> Buruganahalli at 5:14-41: In some embodiments, the SNI standard extension is used as disclosed herein as a mechanism to facilitate destination domain extraction for secure protocols without requiring decryption. For example, a firewall can intercept and monitor data communications from a client and a remote server in order to extract the hostname data from the SNI field (e.g., server_name field of the SNI extension), which is sent as an unencrypted data communication from the client (e.g., a web browser executing on the client) to the remote server (e.g., at the start of the handshaking process for setting up a secure TLS communication channel/session between a client and a remote server). This extracted hostname data (e.g., destination domain) can then be used by the firewall to perform an action(s) based on a firewall policy (e.g., the domain name can be extracted by parsing the client hello message to extract the host name from the server_name field of the SNI extension, which can then be used to apply firewall policies or take responsive actions based on this information without having to wait for data to be decrypted), as further described herein with respect to various embodiments. Also, such SNI information is available in the first packet of this handshaking process, such that a policy (e.g., a security or firewall policy using the hostname information extracted from the SNI information) can be applied sooner based on this early extraction of the hostname from the SNI information such that the rest of the stream can be forwarded without any decryption resulting in less state to track thereby enhancing performance (e.g., of the security device, such as a firewall).

Claim	U.S. Patent No. 9,917,856	Disclosure from Buruganahalli
		<i>See, e.g.,</i> Buruganahalli at 6:66-7:15: In some embodiments, destination domain extraction for secure protocols includes monitoring network communications between a client and a remote server; determining if the client sends a request to create a secure connection with the remote server (e.g., in which the network communications are initiating a setup for a secure protocol-based connection); and extracting a destination domain from the request to create the secure connection with the remote server. In some embodiments, the secure protocol is a secure sockets layer (SSL) protocol or transport layer security (TLS) protocol, and the destination domain is extracted from the server name indication (SNI) of a client hello message sent from the client to the remote server (e.g., the client hello message is an unencrypted communication). In some embodiments, destination domain extraction for secure protocols further includes applying a policy (e.g., a security policy) based on the destination domain to filter traffic at a security device. <i>See, e.g.,</i> Buruganahalli at 7:25-36: In some embodiments, destination domain extraction for secure protocols further includes applying a security policy based on the destination domain to filter traffic at the security device. For example, the security policy can include a malware detection policy, a whitelist/blacklist policy (e.g., in which a session associated with the network communications between the client and the remote server are not decrypted if the destination domain is included in the whitelist, and a session associated with the network communications between the client and the remote server are decrypted if the destination domain is included in the blacklist),

Claim	U.S. Patent No. 9,917,856	Disclosure from Buruganahalli
		and/or a uniform resource locator (URL)/category filtering policy. <i>See, e.g.,</i> Buruganahalli at 7:45-52: In some embodiments, destination domain extraction for secure protocols further includes applying a security policy based on the destination domain to filter traffic at the security device, in which the security policy includes a whitelist/blacklist policy, and in which a session associated with the network communications between the client and the remote server are decrypted if the destination domain is included in a blacklist of the whitelist/blacklist policy. <i>See, e.g.,</i> Buruganahalli at 7:62-8:8: In some embodiments, destination domain extraction for secure protocols further includes decrypting the monitored encrypted network communications of a session associated with the network communications between the client and the remote server to verify that the destination domain name extracted from a server name indication matches the domain associated with a URL for the session associated with the network communications between the client and the remote server. For example, this technique can be used in order to detect malware that attempts to fool a hosting site by including false destination domain information (e.g., in the SNI field in a client hello message of an initial set-up of a TLS session). <i>See, e.g.,</i> Buruganahalli at 8:9-28: In some embodiments, destination domain extraction for secure protocols further includes intercepting a request to establish an encrypted session from the client to the remote server; sending a

Claim	U.S. Patent No. 9,917,856	Disclosure from Buruganahalli
		request to establish the encrypted session on behalf of the client to the remote server; sending an encrypted session response to the client on behalf of the remote server using a session key associated with the firewall device; decrypting encrypted traffic between the client and the remote server to monitor for a request from the client to create a tunnel using the first protocol with the remote server; allowing the request to create the tunnel; monitoring decrypted session traffic between the client and the remote server over the tunnel based on one or more firewall policies; and blocking the session traffic if a violation of a first firewall policy is determined, in which the first firewall policy includes a policy for verifying that the destination domain name extracted from a server name indication matches the domain associated with a URL for a session associated with the network communications between the client and the remote server. <i>See, e.g., Buruganahalli at 8:29-38:</i> In some embodiments, destination domain extraction for secure protocols further includes monitoring encrypted network communications between a client and a remote server (e.g., using a security device, such as a firewall to provide man-in-the-middle decryption techniques, such as described herein), in which the encrypted network communications are encrypted using a first protocol (e.g., SSL/TLS or another protocol for encrypted network communications); and performing an action based on the decrypted network communications between the client and the remote server. <i>See, e.g., Buruganahalli at 9:7-25:</i> As shown in FIG. 1, network traffic monitoring begins at 102. An

Claim	U.S. Patent No. 9,917,856	Disclosure from Buruganahalli
		IP address and port engine 104 determines an IP address and port number for a monitored traffic flow (e.g., a session) based on packet analysis. In some embodiments, user identification is then determined (e.g., user ID can be deduced based on the source IP address). A policy check engine 106 determines whether any policies can be applied based on the IP address and port number. As also shown in FIG. 1, an application signature check engine 108 identifies an application (e.g., using an APP-ID engine using various application signatures for identifying applications based on packet flow analysis). For example, APP-ID engine 108 can be configured to determine what type of traffic the session involves, such as HTTP traffic, HTTPS traffic, SSL/TLS traffic, SSH traffic, DNS requests, FTP traffic, unknown traffic, and various other types of traffic, and such classified traffic can be directed to an appropriate decoder, such as decoders 112, 114, and 116, to decode the classified traffic for each monitored session's traffic flow. <i>See, e.g., Buruganahalli at 9:42-57:</i> As further illustrated in FIG. 1, if the monitored traffic is encrypted (e.g., encrypted using HTTPS, SSL/TLS, SSH, or another known encryption protocol), then the security device can be configured (e.g., based on a firewall policy) to decrypt the monitored traffic using a decrypt engine 110 (e.g., applying trusted man-in-the-middle techniques using a self-signed certificate). For example, in some cases, if the extracted destination domain is not trusted or unknown, then the firewall can be configured to decrypt the monitored traffic for further traffic monitoring and inspection based on a policy (e.g., a firewall policy). As another example, the firewall can be configured to decrypt the monitored traffic to verify that the extracted hostname (e.g., extracted from a client hello message) matches the domain that is being communicated

Claim	U.S. Patent No. 9,917,856	Disclosure from Buruganahalli
		with using the encrypted communications between the client and the remote server/external site. <i>See, e.g.,</i> Buruganahalli at 9:59-64: In some embodiments, the results of the various traffic monitoring techniques using known protocol decoder engine 112, identified traffic engine 114, and unknown protocol decoder engine 116 described above are provided to report and enforce policies engine 120 (e.g., network/routing policies, security policies, and/or firewall policies). <i>See, e.g.,</i> Buruganahalli at 10:36-55: FIG. 3A is a functional block diagram illustrating an SSL/ TLS session passing through a firewall that can implement destination domain extraction for secure protocols in accordance with some embodiments. In that session, a typical firewall can initially just inspect the handshake traffic (e.g., the initial negotiation to setup a secure communication channel/tunnel between a client and a remote server), as all the application data is encrypted. As shown, a client 302 establishes a secure tunnel session (e.g., creates an SSL tunnel) with a remote server 306. The client 302 can use the secure tunnel with the server 306 to access a network service 308, which can be a network service activity that is in violation of one or more firewall policies/rules implemented by the fire- wall device 304 (e.g., a policy that includes requirements or rules related to destination domains that may be used for secure protocol communications). As the secure tunnel session traffic is encrypted and passes through a firewall 304, the firewall 304 typically cannot decrypt the encrypted secure tunnel session traffic and, thus, cannot

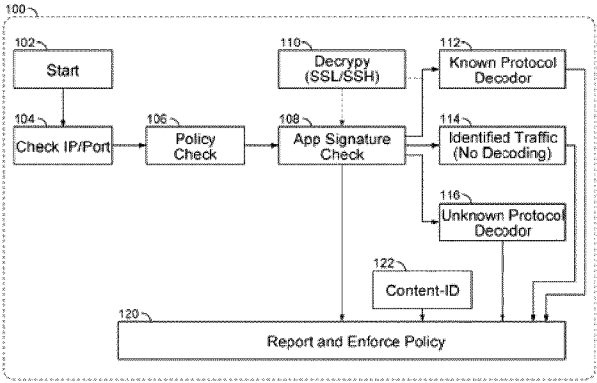
Claim	U.S. Patent No. 9,917,856	Disclosure from Buruganahalli
		detect such firewall policy/rule violation(s). <i>See, e.g.,</i> Buruganahalli at 11:38-49: In addition, various techniques for intercepting and decoding encrypted tunneling communications from a client to a remote server are disclosed herein. For example, a trusted man-in-the-middle technique that can be used for intercepting and decoding encrypted tunneling communications to further monitor such traffic in clear text (e.g., unencrypted) is described below with respect to FIG. 38. These decryption techniques can be used in addition to and/or in combination with the various techniques described herein for destination domain extraction for secure protocols to further enhance the various security and firewall related techniques disclosed herein. <i>See, e.g.,</i> Buruganahalli at 11:50-12:11: FIG. 3B is another functional block diagram illustrating the use of a firewall for destination domain extraction for secure protocols in accordance with some embodiments. As shown, a client 312 attempts to establish an SSL session with a remote server 316. The client 312 can attempt to use, for example, a secure tunnel with the server 316 to access a network service 318, which can be a network service activity that is in violation of one or more firewall policies/rules implemented by a firewall device 314. However, in this case, the SSL session request is intercepted and detected by the firewall 314. In response, the firewall 314 performs a trusted man-in-the-middle technique by effectively splitting the SSL session between the client 312 and the remote server 316 into two half sessions shown as Session A and Session B in FIG. 38. In Session A, the firewall 314 acts as the remote server 316 such that it is

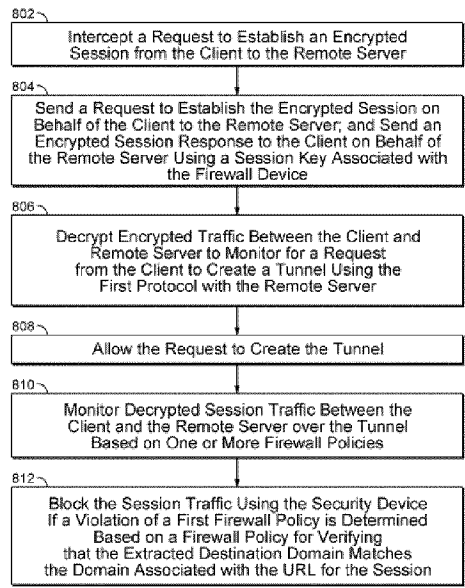
Claim	U.S. Patent No. 9,917,856	Disclosure from Buruganahalli
		transparent to the client 312 that it is not communicating directly with the remote server 316. Session A traffic is encrypted using the session key 51 associated with the firewall device. In Session B, the firewall 314 acts as the client 312 such that it is transparent to the remote server 316 that it is not communicating directly with the client 312. Session B traffic is encrypted using the session key S2 associated with the firewall device (e.g., the firewall device can store the fingerprint from the remote server in association with that remote server IP address). After the session set-up handshaking is completed for each of Session A and Session B, any data that is communicated from the client 312 to the firewall 314 is decrypted using a session key 51 and is then inspected by the firewall 314. <i>See, e.g., Buruganahalli at 12:34-42:</i> Thus, using these techniques the firewall 314 is between the client 312 and the server 316 and can present itself as server of the session to the client and can present itself as the client of the session to the server. In this manner, the firewall 314 can inspect decrypted traffic and re-encrypt it before forwarding it to the other party. During inspection, if encrypted tunneling traffic is detected, the entire session can either be tom down or various other responsive actions can be performed as described herein. <i>See, e.g., Buruganahalli at 14:3:11:</i> As described herein, an initial TLS handshake exchange of that flow can be parsed to extract a hostname to facilitate destination domain extraction for secure protocols (e.g., and to apply one or more policies if applicable based on the extracted destination domain) using various techniques described herein. If applicable (e.g., using one or more policies based on the extracted destination

Claim	U.S. Patent No. 9,917,856	Disclosure from Buruganahalli
		domain), SSL decryption is applied by SSL decryption engine 610 using various techniques as described herein. <i>See, e.g.,</i> Buruganahalli at 14:45-62: FIG. 7 is a flow diagram for providing destination domain extraction for secure protocols in accordance with some embodiments. At 702, monitoring network communications between a client and a remote server is performed. At 704, determining if the client sends a request to create a secure connection with the remote server is performed (e.g., in which the network communications are initiating a setup for a secure protocol-based connection). At 706, extracting a destination domain from the request to create the secure connection with the remote server is performed. In some embodiments, the secure protocol is a secure sockets layer (SSL) protocol or transport layer security (TLS) protocol, and the destination domain is extracted from the server name indication (SNI) of a client hello message sent from the client to the remote server. In some embodiments, destination domain extraction for secure protocols further includes applying a policy (e.g., a security policy) based on the destination domain to filter traffic using a security device. <i>See, e.g.,</i> Buruganahalli at claim 1: A system for destination domain extraction for secure protocols, comprising: a security device; a processor configured to:

Claim	U.S. Patent No. 9,917,856	Disclosure from Buruganahalli
		monitor network communications between a client and a remote server; determine if the client sends a request to create a secure connection with the remote server, wherein the secure connection utilizes a secure protocol, the secure protocol being a secure sockets layer (SSL) protocol or transport layer security (TLS) protocol; and extract a destination domain from the request to create the secure connection with the remote server, comprising: before the secure connection is created, extract the destination domain from a server name indication (SNI) of a client hello message sent from the client to the remote server, the secure connection including an encrypted message, wherein the extracting of the destination domain from the SNI of the client hello message sent from the client to the remote server comprises to: identify the SNI from the client hello message during a handshaking process for setting the secure connection between the client and the remote server; extract a domain identified in a public certificate sent from the remote server to the client; compare the destination domain and the domain identified in the public certificate; and in the event that the destination domain matches the domain identified in the public certificate, apply a security policy based on the destination domain to filter traffic at the security device, wherein the security policy includes a whitelist/blacklist policy;

Claim	U.S. Patent No. 9,917,856	Disclosure from Buruganahalli
		and a memory coupled to the processor and configured to provide the processor with instructions. <i>See, e.g.,</i> Buruganahalli at claim 7: The system recited in claim 1, wherein a session associated with the network communications between the client and the remote server are decrypted to verify that the destination domain extracted from a server name indication (SNI) matches a domain associated with a uniform resource locator (URL) for the session associated with the network communications between the client and the remote server. <i>See, e.g.,</i> Buruganahalli at claim 8: The system recited in claim 1, wherein the processor is further configured to: decrypt a monitored encrypted network communications of a session associated with the network communications between the client and the remote server to verify that the destination domain extracted from a server name indication (SNI) matches a domain associated with a uniform resource locator (URL) for the session associated with the network communications between the client and the remote server. <i>See, e.g.,</i> Buruganahalli at Fig. 1:

Claim	U.S. Patent No. 9,917,856	Disclosure from Buruganahalli
		 <pre> graph TD 102[Start] --> 104[Check IP/Port] 104 --> 106[Policy Check] 106 --> 108[App Signature Check] 108 --> 110[Decrypy (SSL/SSH)] 108 --> 114[Identified Traffic (No Decoding)] 110 --> 112[Known Protocol Decoder] 114 --> 116[Unknown Protocol Decoder] 112 --> 122[Content-ID] 116 --> 122[Content-ID] 108 --> 120[Report and Enforce Policy] 122 --> 120 116 --> 120 </pre> FIG. 1 <i>See, e.g., Buruganahalli at Fig. 8:</i>

Claim	U.S. Patent No. 9,917,856	Disclosure from Buruganahalli
		 <pre> graph TD 802[802 Intercept a Request to Establish an Encrypted Session from the Client to the Remote Server] --> 804[804 Send a Request to Establish the Encrypted Session on Behalf of the Client to the Remote Server; and Send an Encrypted Session Response to the Client on Behalf of the Remote Server Using a Session Key Associated with the Firewall Device] 804 --> 806[806 Decrypt Encrypted Traffic Between the Client and Remote Server to Monitor for a Request from the Client to Create a Tunnel Using the First Protocol with the Remote Server] 806 --> 808[808 Allow the Request to Create the Tunnel] 808 --> 810[810 Monitor Decrypted Session Traffic Between the Client and the Remote Server over the Tunnel Based on One or More Firewall Policies] 810 --> 812[812 Block the Session Traffic Using the Security Device If a Violation of a First Firewall Policy is Determined Based on a Firewall Policy for Verifying that the Extracted Destination Domain Matches the Domain Associated with the URL for the Session] </pre> FIG. 8
1.5	filtering, by the packet-filtering system and based on at least one of a uniform resource identifier (URI) specified by the plurality of packet-filtering rules, data indicating a protocol version specified by the plurality of packet-filtering rules, data indicating a method specified by the plurality of packet-filtering rules, data indicating a request specified by the plurality of packet-filtering rules, or data indicating a command specified by the plurality of packet-filtering rules:	<i>See, e.g., Burugunahalli at 4:24-35:</i> While performing various decryption techniques as described herein can facilitate applying filtering based on a policy (e.g., a security policy, such as a firewall policy), it would be desirable and efficient to further facilitate policy- based filtering on such data communication flows (e.g., sessions) prior to the set-up of the encrypted data communication for such flows (e.g., to monitor the initial unencrypted/ clear text data communications exchanged as part of an initial handshake to setup a secure connection for a new session between a client and a remote server, which is prior to the establishment of the secure communication channel/tunnel after

Claim	U.S. Patent No. 9,917,856	Disclosure from Buruganahalli
		which any data communications intercepted between the client and the remote server would be encrypted). <i>See, e.g.,</i> Buruganahalli at 5:14-41: In some embodiments, the SNI standard extension is used as disclosed herein as a mechanism to facilitate destination domain extraction for secure protocols without requiring decryption. For example, a firewall can intercept and monitor data communications from a client and a remote server in order to extract the hostname data from the SNI field (e.g., server_name field of the SNI extension), which is sent as an unencrypted data communication from the client (e.g., a web browser executing on the client) to the remote server (e.g., at the start of the handshaking process for setting up a secure TLS communication channel/session between a client and a remote server). This extracted hostname data (e.g., destination domain) can then be used by the firewall to perform an action(s) based on a firewall policy (e.g., the domain name can be extracted by parsing the client hello message to extract the host name from the server_name field of the SNI extension, which can then be used to apply firewall policies or take responsive actions based on this information without having to wait for data to be decrypted), as further described herein with respect to various embodiments. Also, such SNI information is available in the first packet of this handshaking process, such that a policy (e.g., a security or firewall policy using the hostname information extracted from the SNI information) can be applied sooner based on this early extraction of the hostname from the SNI information such that the rest of the stream can be forwarded without any decryption resulting in less state to track thereby enhancing performance (e.g., of the security device, such as a firewall).

Claim	U.S. Patent No. 9,917,856	Disclosure from Buruganahalli
		<i>See, e.g.,</i> Buruganahalli at 6:66-7:15: In some embodiments, destination domain extraction for secure protocols includes monitoring network communications between a client and a remote server; determining if the client sends a request to create a secure connection with the remote server (e.g., in which the network communications are initiating a setup for a secure protocol-based connection); and extracting a destination domain from the request to create the secure connection with the remote server. In some embodiments, the secure protocol is a secure sockets layer (SSL) protocol or transport layer security (TLS) protocol, and the destination domain is extracted from the server name indication (SNI) of a client hello message sent from the client to the remote server (e.g., the client hello message is an unencrypted communication). In some embodiments, destination domain extraction for secure protocols further includes applying a policy (e.g., a security policy) based on the destination domain to filter traffic at a security device. <i>See, e.g.,</i> Buruganahalli at 7:25-36: In some embodiments, destination domain extraction for secure protocols further includes applying a security policy based on the destination domain to filter traffic at the security device. For example, the security policy can include a malware detection policy, a whitelist/blacklist policy (e.g., in which a session associated with the network communications between the client and the remote server are not decrypted if the destination domain is included in the whitelist, and a session associated with the network communications between the client and the remote server are decrypted if the destination domain is included in the blacklist),

Claim	U.S. Patent No. 9,917,856	Disclosure from Buruganahalli
		and/or a uniform resource locator (URL)/category filtering policy. <i>See, e.g.,</i> Buruganahalli at 7:62-8:8: In some embodiments, destination domain extraction for secure protocols further includes decrypting the monitored encrypted network communications of a session associated with the network communications between the client and the remote server to verify that the destination domain name extracted from a server name indication matches the domain associated with a URL for the session associated with the network communications between the client and the remote server. For example, this technique can be used in order to detect malware that attempts to fool a hosting site by including false destination domain information (e.g., in the SNI field in a client hello message of an initial set-up of a TLS session). <i>See, e.g.,</i> Buruganahalli at 8:9-28: In some embodiments, destination domain extraction for secure protocols further includes intercepting a request to establish an encrypted session from the client to the remote server; sending a request to establish the encrypted session on behalf of the client to the remote server; sending an encrypted session response to the client on behalf of the remote server using a session key associated with the firewall device; decrypting encrypted traffic between the client and the remote server to monitor for a request from the client to create a tunnel using the first protocol with the remote server; allowing the request to create the tunnel; monitoring decrypted session traffic between the client and the remote server over the tunnel based on one or more firewall policies; and blocking the session traffic if a violation of a first firewall policy is determined, in which the first firewall policy includes a policy for verifying that

Claim	U.S. Patent No. 9,917,856	Disclosure from Buruganahalli
		the destination domain name extracted from a server name indication matches the domain associated with a URL for a session associated with the network communications between the client and the remote server. <i>See, e.g., Buruganahalli at 8:29-38:</i> In some embodiments, destination domain extraction for secure protocols further includes monitoring encrypted network communications between a client and a remote server (e.g., using a security device, such as a firewall to provide man-in-the-middle decryption techniques, such as described herein), in which the encrypted network communications are encrypted using a first protocol (e.g., SSL/TLS or another protocol for encrypted network communications); and performing an action based on the decrypted network communications between the client and the remote server. <i>See, e.g., Buruganahalli at 9:7-25:</i> As shown in FIG. 1, network traffic monitoring begins at 102. An IP address and port engine 104 determines an IP address and port number for a monitored traffic flow (e.g., a session) based on packet analysis. In some embodiments, user identification is then determined (e.g., user ID can be deduced based on the source IP address). A policy check engine 106 determines whether any policies can be applied based on the IP address and port number. As also shown in FIG. 1, an application signature check engine 108 identifies an application (e.g., using an APP-ID engine using various application signatures for identifying applications based on packet flow analysis). For example, APP-ID engine 108 can be configured to determine what type of traffic the session involves,

Claim	U.S. Patent No. 9,917,856	Disclosure from Buruganahalli
		such as HTTP traffic, HTTPS traffic, SSL/TLS traffic, SSH traffic, DNS requests, FTP traffic, unknown traffic, and various other types of traffic, and such classified traffic can be directed to an appropriate decoder, such as decoders 112, 114, and 116, to decode the classified traffic for each monitored session's traffic flow. <i>See, e.g.,</i> Buruganahalli at 9:42-57: As further illustrated in FIG. 1, if the monitored traffic is encrypted (e.g., encrypted using HTTPS, SSL/TLS, SSH, or another known encryption protocol), then the security device can be configured (e.g., based on a firewall policy) to decrypt the monitored traffic using a decrypt engine 110 (e.g., applying trusted man-in-the-middle techniques using a self-signed certificate). For example, in some cases, if the extracted destination domain is not trusted or unknown, then the firewall can be configured to decrypt the monitored traffic for further traffic monitoring and inspection based on a policy (e.g., a firewall policy). As another example, the firewall can be configured to decrypt the monitored traffic to verify that the extracted hostname (e.g., extracted from a client hello message) matches the domain that is being communicated with using the encrypted communications between the client and the remote server/external site. <i>See, e.g.,</i> Buruganahalli at 10:36-55: FIG. 3A is a functional block diagram illustrating an SSL/ TLS session passing through a firewall that can implement destination domain extraction for secure protocols in accordance with some embodiments. In that session, a typical firewall can initially just inspect the handshake traffic (e.g., the initial negotiation to setup a secure communication channel/tunnel between a client and a

Claim	U.S. Patent No. 9,917,856	Disclosure from Buruganahalli
		remote server), as all the application data is encrypted. As shown, a client 302 establishes a secure tunnel session (e.g., creates an SSL tunnel) with a remote server 306. The client 302 can use the secure tunnel with the server 306 to access a network service 308, which can be a network service activity that is in violation of one or more firewall policies/rules implemented by the fire- wall device 304 (e.g., a policy that includes requirements or rules related to destination domains that may be used for secure protocol communications). As the secure tunnel session traffic is encrypted and passes through a firewall 304, the firewall 304 typically cannot decrypt the encrypted secure tunnel session traffic and, thus, cannot detect such firewall policy/rule violation(s). <i>See, e.g., Buruganahalli at 11:38-49:</i> In addition, various techniques for intercepting and decoding encrypted tunneling communications from a client to a remote server are disclosed herein. For example, a trusted man-in-the-middle technique that can be used for intercepting and decoding encrypted tunneling communications to further monitor such traffic in clear text (e.g., unencrypted) is described below with respect to FIG. 38. These decryption techniques can be used in addition to and/or in combination with the various techniques described herein for destination domain extraction for secure protocols to further enhance the various security and firewall related techniques disclosed herein. <i>See, e.g., Buruganahalli at 11:50-12:11:</i> FIG. 3B is another functional block diagram illustrating the use of a firewall for destination domain extraction for secure protocols in accordance with some embodiments. As shown, a client 312

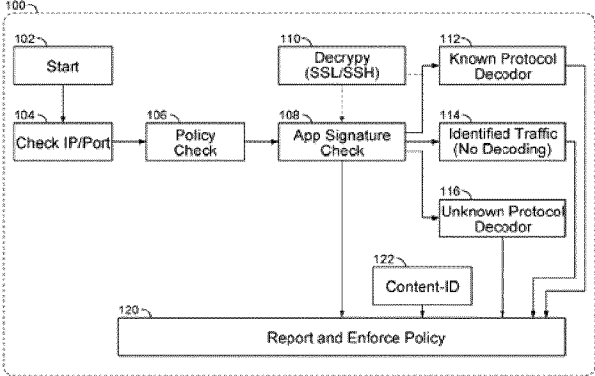
Claim	U.S. Patent No. 9,917,856	Disclosure from Buruganahalli
		attempts to establish an SSL session with a remote server 316. The client 312 can attempt to use, for example, a secure tunnel with the server 316 to access a network service 318, which can be a network service activity that is in violation of one or more firewall policies/rules implemented by a firewall device 314. However, in this case, the SSL session request is intercepted and detected by the firewall 314. In response, the firewall 314 performs a trusted man-in-the-middle technique by effectively splitting the SSL session between the client 312 and the remote server 316 into two half sessions shown as Session A and Session B in FIG. 38. In Session A, the firewall 314 acts as the remote server 316 such that it is transparent to the client 312 that it is not communicating directly with the remote server 316. Session A traffic is encrypted using the session key 51 associated with the firewall device. In Session B, the firewall 314 acts as the client 312 such that it is transparent to the remote server 316 that it is not communicating directly with the client 312. Session B traffic is encrypted using the session key S2 associated with the firewall device (e.g., the firewall device can store the fingerprint from the remote server in association with that remote server IP address). After the session set-up handshaking is completed for each of Session A and Session B, any data that is communicated from the client 312 to the firewall 314 is decrypted using a session key 51 and is then inspected by the firewall 314. <i>See, e.g., Buruganahalli at 12:34-42:</i> Thus, using these techniques the firewall 314 is between the client 312 and the server 316 and can present itself as server of the session to the client and can present itself as the client of the session to the server. In this manner, the firewall 314 can inspect decrypted traffic and re-encrypt it before forwarding it to the other party. During inspection, if encrypted tunneling traffic is detected,

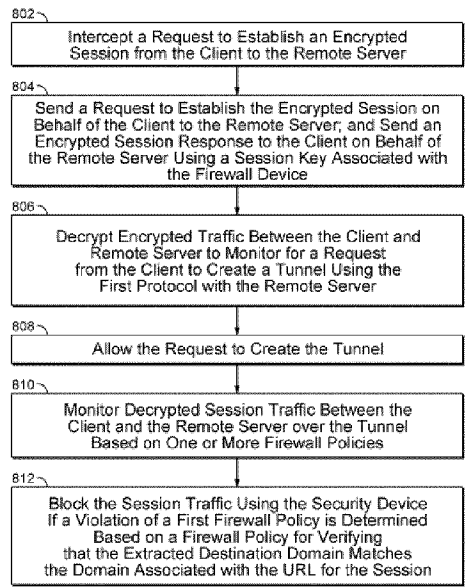
Claim	U.S. Patent No. 9,917,856	Disclosure from Buruganahalli
		the entire session can either be tom down or various other responsive actions can be performed as described herein. <i>See, e.g.,</i> Burugunahalli at 14:3:11: As described herein, an initial TLS handshake exchange of that flow can be parsed to extract a hostname to facilitate destination domain extraction for secure protocols (e.g., and to apply one or more policies if applicable based on the extracted destination domain) using various techniques described herein. If applicable (e.g., using one or more policies based on the extracted destination domain), SSL decryption is applied by SSL decryption engine 610 using various techniques as described herein. <i>See, e.g.,</i> Burugunahalli at 14:45-62: FIG. 7 is a flow diagram for providing destination domain extraction for secure protocols in accordance with some embodiments. At 702, monitoring network communications between a client and a remote server is performed. At 704, determining if the client sends a request to create a secure connection with the remote server is performed (e.g., in which the network communications are initiating a setup for a secure protocol-based connection). At 706, extracting a destination domain from the request to create the secure connection with the remote server is performed. In some embodiments, the secure protocol is a secure sockets layer (SSL) protocol or transport layer security (TLS) protocol, and the destination domain is extracted from the server name indication (SNI) of a client hello message sent from the client to the remote server. In some embodiments, destination domain extraction for secure protocols further includes applying a policy (e.g., a security policy) based on the destination

Claim	U.S. Patent No. 9,917,856	Disclosure from Buruganahalli
		domain to filter traffic using a security device. <i>See, e.g.,</i> Buruganahalli at claim 1: A system for destination domain extraction for secure protocols, comprising: a security device; a processor configured to: monitor network communications between a client and a remote server; determine if the client sends a request to create a secure connection with the remote server, wherein the secure connection utilizes a secure protocol, the secure protocol being a secure sockets layer (SSL) protocol or transport layer security (TLS) protocol; and extract a destination domain from the request to create the secure connection with the remote server, comprising: before the secure connection is created, extract the destination domain from a server name indication (SNI) of a client hello message sent from the client to the remote server, the secure connection including an encrypted message, wherein the extracting of the destination domain from the SNI of the client hello message sent from the client to the remote server comprises to: identify the SNI from the client hello message during a handshaking process for setting the secure connection between the client and the remote server;

Claim	U.S. Patent No. 9,917,856	Disclosure from Buruganahalli
		extract a domain identified in a public certificate sent from the remote server to the client; compare the destination domain and the domain identified in the public certificate; and in the event that the destination domain matches the domain identified in the public certificate, apply a security policy based on the destination domain to filter traffic at the security device, wherein the security policy includes a whitelist/blacklist policy; and a memory coupled to the processor and configured to provide the processor with instructions. <i>See, e.g., Buruganahalli at claim 7:</i> The system recited in claim 1, wherein a session associated with the network communications between the client and the remote server are decrypted to verify that the destination domain extracted from a server name indication (SNI) matches a domain associated with a uniform resource locator (URL) for the session associated with the network communications between the client and the remote server. <i>See, e.g., Buruganahalli at claim 8:</i> The system recited in claim 1, wherein the processor is further configured to: decrypt a monitored encrypted network communications of a session associated with the network communications between the client and the remote server to verify that the destination domain extracted from a server name indication

Claim	U.S. Patent No. 9,917,856	Disclosure from Buruganahalli
		(SNI) matches a domain associated with a uniform resource locator (URL) for the session associated with the network communications between the client and the remote server. <i>See, e.g.,</i> Buruganahalli at claim 9: The system recited in claim 1, wherein the processor is further configured to: intercept a request to establish an encrypted session from the client to the remote server; send a request to establish the encrypted session on behalf of the client to the remote server; send an encrypted session response to the client on behalf of the remote server using a session key associated with the security device; decrypt encrypted traffic between the client and the remote server to monitor for a request from the client to create a tunnel using a first protocol with the remote server; allow the request to create the tunnel; monitor decrypted session traffic between the client and the remote server over the tunnel based on one or more firewall policies; and block the session traffic using the security device if a violation of a first firewall policy is determined, wherein the first firewall policy includes a policy for verifying that the destination domain extracted from a server name indication (SNI) matches a domain associated with a uniform resource locator (URL) for a session associated

Claim	U.S. Patent No. 9,917,856	Disclosure from Buruganahalli
		with the network communications between the client and the remote server. See, e.g., Buruganahalli at Fig. 1:  <pre> graph TD 102[Start] --> 104[Check IP/Port] 104 --> 106[Policy Check] 106 --> 108[App Signature Check] 108 --> 110[Decrypy (SSL/SSH)] 108 --> 112[Known Protocol Decoder] 110 --> 114[Identified Traffic (No Decoding)] 112 --> 116[Unknown Protocol Decoder] 114 --> 120[Report and Enforce Policy] 116 --> 120 122[Content-ID] --> 120 </pre> FIG. 1 See, e.g., Buruganahalli at Fig. 8:

Claim	U.S. Patent No. 9,917,856	Disclosure from Buruganahalli
		 <pre> graph TD 802[802 Intercept a Request to Establish an Encrypted Session from the Client to the Remote Server] --> 804[804 Send a Request to Establish the Encrypted Session on Behalf of the Client to the Remote Server; and Send an Encrypted Session Response to the Client on Behalf of the Remote Server Using a Session Key Associated with the Firewall Device] 804 --> 806[806 Decrypt Encrypted Traffic Between the Client and Remote Server to Monitor for a Request from the Client to Create a Tunnel Using the First Protocol with the Remote Server] 806 --> 808[808 Allow the Request to Create the Tunnel] 808 --> 810[810 Monitor Decrypted Session Traffic Between the Client and the Remote Server over the Tunnel Based on One or More Firewall Policies] 810 --> 812[812 Block the Session Traffic Using the Security Device If a Violation of a First Firewall Policy is Determined Based on a Firewall Policy for Verifying that the Extracted Destination Domain Matches the Domain Associated with the URL for the Session] </pre> FIG. 8
1.6	packets comprising the portion of the unencrypted data that corresponds to one or more network-threat indicators of the plurality of network-threat indicators; and	<i>See, e.g.,</i> Burugunahalli at 4:24-35: While performing various decryption techniques as described herein can facilitate applying filtering based on a policy (e.g., a security policy, such as a firewall policy), it would be desirable and efficient to further facilitate policy- based filtering on such data communication flows (e.g., sessions) prior to the set-up of the encrypted data communication for such flows (e.g., to monitor the initial unencrypted/ clear text data communications exchanged as part of an initial handshake to setup a secure connection for a new session between a client and a remote server, which is prior to the establishment of the secure communication channel/tunnel after

Claim	U.S. Patent No. 9,917,856	Disclosure from Buruganahalli
		which any data communications intercepted between the client and the remote server would be encrypted). <i>See, e.g.,</i> Buruganahalli at 5:14-41: In some embodiments, the SNI standard extension is used as disclosed herein as a mechanism to facilitate destination domain extraction for secure protocols without requiring decryption. For example, a firewall can intercept and monitor data communications from a client and a remote server in order to extract the hostname data from the SNI field (e.g., server_name field of the SNI extension), which is sent as an unencrypted data communication from the client (e.g., a web browser executing on the client) to the remote server (e.g., at the start of the handshaking process for setting up a secure TLS communication channel/session between a client and a remote server). This extracted hostname data (e.g., destination domain) can then be used by the firewall to perform an action(s) based on a firewall policy (e.g., the domain name can be extracted by parsing the client hello message to extract the host name from the server_name field of the SNI extension, which can then be used to apply firewall policies or take responsive actions based on this information without having to wait for data to be decrypted), as further described herein with respect to various embodiments. Also, such SNI information is available in the first packet of this handshaking process, such that a policy (e.g., a security or firewall policy using the hostname information extracted from the SNI information) can be applied sooner based on this early extraction of the hostname from the SNI information such that the rest of the stream can be forwarded without any decryption resulting in less state to track thereby enhancing performance (e.g., of the security device, such as a firewall).

Claim	U.S. Patent No. 9,917,856	Disclosure from Buruganahalli
		<i>See, e.g.,</i> Buruganahalli at 5:54-58: For example, if a new flow is determined to violate a policy prior to the set-up of the encrypted data communication for that flow between a client and a remote server, then the flow can be blocked and decryption is not required. As an example, Bob who is a user (e.g., an employee) of ACME Company may attempt to logon using a web browser executing on his desktop office computer to a remote server that is associated with the online banking site (e.g., web site) of Banking Corporation. If the firewall policy of ACME Company has white listed the domain associated with the Banking Corporation as a trusted domain, then Bob's connection (e.g., an SSL/TLS session) with the web site for the Banking Corporation can be allowed using various techniques described herein. <i>See, e.g.,</i> Buruganahalli at 6:66-7:15: In some embodiments, destination domain extraction for secure protocols includes monitoring network communications between a client and a remote server; determining if the client sends a request to create a secure connection with the remote server (e.g., in which the network communications are initiating a setup for a secure protocol-based connection); and extracting a destination domain from the request to create the secure connection with the remote server. In some embodiments, the secure protocol is a secure sockets layer (SSL) protocol or transport layer security (TLS) protocol, and the destination domain is extracted from the server name indication (SNI) of a client hello message sent from the client to the remote server (e.g., the client hello message is an unencrypted communication). In some embodiments, destination domain extraction for secure protocols further includes applying a policy (e.g., a security policy) based on the destination domain to

Claim	U.S. Patent No. 9,917,856	Disclosure from Buruganahalli
		filter traffic at a security device. <i>See, e.g.,</i> Buruganahalli at 7:25-36: In some embodiments, destination domain extraction for secure protocols further includes applying a security policy based on the destination domain to filter traffic at the security device. For example, the security policy can include a malware detection policy, a whitelist/blacklist policy (e.g., in which a session associated with the network communications between the client and the remote server are not decrypted if the destination domain is included in the whitelist, and a session associated with the network communications between the client and the remote server are decrypted if the destination domain is included in the blacklist), and/or a uniform resource locator (URL)/category filtering policy. <i>See, e.g.,</i> Buruganahalli at 7:45-52: In some embodiments, destination domain extraction for secure protocols further includes applying a security policy based on the destination domain to filter traffic at the security device, in which the security policy includes a whitelist/blacklist policy, and in which a session associated with the network communications between the client and the remote server are decrypted if the destination domain is included in a blacklist of the whitelist/blacklist policy. <i>See, e.g.,</i> Buruganahalli at 9:59-64: In some embodiments, the results of the various traffic monitoring techniques using known protocol decoder engine 112, identified traffic engine 114, and unknown protocol decoder engine 116

Claim	U.S. Patent No. 9,917,856	Disclosure from Buruganahalli
		described above are provided to report and enforce policies engine 120 (e.g., network/routing policies, security policies, and/or firewall policies). <i>See, e.g.,</i> Buruganahalli at 9:64-10:2: For example, firewall policies can be applied to the monitored network traffic using application identification, user identification, and/or other information to match signatures (e.g., file-based, protocol-based, and/or other types/forms of signatures for detecting malware or suspicious behavior). <i>See, e.g.,</i> Buruganahalli at 10:3-9: In some embodiments, firewall 100 also includes a content-ID engine 122, and, in some embodiments, the content-ID engine's identified content is also used by report and enforce policy engine 120 to provide, for example, URL/category filtering, possibly in various combinations with other information, such as application, user, and/or other information, to enforce various security/firewall policies/rules. <i>See, e.g.,</i> Buruganahalli at claim 1: A system for destination domain extraction for secure protocols, comprising: a security device; a processor configured to: monitor network communications between a client and a remote

Claim	U.S. Patent No. 9,917,856	Disclosure from Buruganahalli
		server; determine if the client sends a request to create a secure connection with the remote server, wherein the secure connection utilizes a secure protocol, the secure protocol being a secure sockets layer (SSL) protocol or transport layer security (TLS) protocol; and extract a destination domain from the request to create the secure connection with the remote server, comprising: before the secure connection is created, extract the destination domain from a server name indication (SNI) of a client hello message sent from the client to the remote server, the secure connection including an encrypted message, wherein the extracting of the destination domain from the SNI of the client hello message sent from the client to the remote server comprises to: identify the SNI from the client hello message during a handshaking process for setting the secure connection between the client and the remote server; extract a domain identified in a public certificate sent from the remote server to the client; compare the destination domain and the domain identified in the public certificate; and in the event that the destination domain matches the domain identified in the public certificate, apply a security policy based on the destination domain to filter traffic at the security device, wherein the security policy includes a whitelist/blacklist policy;

Claim	U.S. Patent No. 9,917,856	Disclosure from Buruganahalli
		and a memory coupled to the processor and configured to provide the processor with instructions. <i>See, e.g.,</i> Buruganahalli at claim 2: The system recited in claim 1, wherein the client hello message is an unencrypted communication. <i>See, e.g.,</i> Buruganahalli at claim 4: The system recited in claim 1, wherein the security policy includes a uniform resource locator (URL)/category filtering related policy. <i>See, e.g.,</i> Buruganahalli at claim 7: The system recited in claim 1, wherein a session associated with the network communications between the client and the remote server are decrypted to verify that the destination domain extracted from a server name indication (SNI) matches a domain associated with a uniform resource locator (URL) for the session associated with the network communications between the client and the remote server. <i>See, e.g.,</i> Buruganahalli at claim 8: The system recited in claim 1, wherein the processor is further configured to: decrypt a monitored encrypted network communications of a session associated with the network communications between the client and the remote server to verify that the destination domain extracted from a server name indication (SNI) matches a domain associated with a uniform resource locator

Claim	U.S. Patent No. 9,917,856	Disclosure from Buruganahalli
		(URL) for the session associated with the network communications between the client and the remote server. <i>See, e.g.,</i> Buruganahalli at claim 9: The system recited in claim 1, wherein the processor is further configured to: - intercept a request to establish an encrypted session from the client to the remote server; - send a request to establish the encrypted session on behalf of the client to the remote server; - send an encrypted session response to the client on behalf of the remote server using a session key associated with the security device; - decrypt encrypted traffic between the client and the remote server to monitor for a request from the client to create a tunnel using a first protocol with the remote server; - allow the request to create the tunnel; - monitor decrypted session traffic between the client and the remote server over the tunnel based on one or more firewall policies; and - block the session traffic using the security device if a violation of a first firewall policy is determined, wherein the first firewall policy includes a policy for verifying that the destination domain extracted from a server name indication (SNI) matches a domain associated with a uniform resource locator (URL) for a session associated with the network communications between the client and the remote server.
1.7	the determined packets comprising the encrypted data that corresponds to the one or more network-threat indicators; and	<i>See, e.g.,</i> Buruganahalli at 4:24-35: While performing various decryption techniques as described herein can facilitate applying filtering based on a policy (e.g., a security policy, such as a firewall policy), it would be desirable and

Claim	U.S. Patent No. 9,917,856	Disclosure from Buruganahalli
		efficient to further facilitate policy- based filtering on such data communication flows (e.g., sessions) prior to the set-up of the encrypted data communication for such flows (e.g., to monitor the initial unencrypted/ clear text data communications exchanged as part of an initial handshake to setup a secure connection for a new session between a client and a remote server, which is prior to the establishment of the secure communication channel/tunnel after which any data communications intercepted between the client and the remote server would be encrypted). <i>See, e.g., Buruganahalli at 5:14-41:</i> In some embodiments, the SNI standard extension is used as disclosed herein as a mechanism to facilitate destination domain extraction for secure protocols without requiring decryption. For example, a firewall can intercept and monitor data communications from a client and a remote server in order to extract the hostname data from the SNI field (e.g., server_name field of the SNI extension), which is sent as an unencrypted data communication from the client (e.g., a web browser executing on the client) to the remote server (e.g., at the start of the handshaking process for setting up a secure TLS communication channel/session between a client and a remote server). This extracted hostname data (e.g., destination domain) can then be used by the firewall to perform an action(s) based on a firewall policy (e.g., the domain name can be extracted by parsing the client hello message to extract the host name from the server_name field of the SNI extension, which can then be used to apply firewall policies or take responsive actions based on this information without having to wait for data to be decrypted), as further described herein with respect to various embodiments. Also, such SNI information is available in the first packet of this handshaking process, such that a policy (e.g., a

Claim	U.S. Patent No. 9,917,856	Disclosure from Buruganahalli
		security or firewall policy using the hostname information extracted from the SNI information) can be applied sooner based on this early extraction of the hostname from the SNI information such that the rest of the stream can be forwarded without any decryption resulting in less state to track thereby enhancing performance (e.g., of the security device, such as a firewall). <i>See, e.g., Buruganahalli at 5:54-58:</i> For example, if a new flow is determined to violate a policy prior to the set-up of the encrypted data communication for that flow between a client and a remote server, then the flow can be blocked and decryption is not required. As an example, Bob who is a user (e.g., an employee) of ACME Company may attempt to logon using a web browser executing on his desktop office computer to a remote server that is associated with the online banking site (e.g., web site) of Banking Corporation. If the firewall policy of ACME Company has white listed the domain associated with the Banking Corporation as a trusted domain, then Bob's connection (e.g., an SSL/TLS session) with the web site for the Banking Corporation can be allowed using various techniques described herein. <i>See, e.g., Buruganahalli at 6:6-11:</i> For example, if a flow is initially determined to be permissible based on a policy prior to the set-up of the encrypted data communication for that flow between a client and a remote server, then the flow can be allowed, and a policy may require that the session be decrypted for further analysis and further potential filtering based on the policy.

Claim	U.S. Patent No. 9,917,856	Disclosure from Buruganahalli
		<i>See, e.g.,</i> Buruganahalli at 6:66-7:15: In some embodiments, destination domain extraction for secure protocols includes monitoring network communications between a client and a remote server; determining if the client sends a request to create a secure connection with the remote server (e.g., in which the network communications are initiating a setup for a secure protocol-based connection); and extracting a destination domain from the request to create the secure connection with the remote server. In some embodiments, the secure protocol is a secure sockets layer (SSL) protocol or transport layer security (TLS) protocol, and the destination domain is extracted from the server name indication (SNI) of a client hello message sent from the client to the remote server (e.g., the client hello message is an unencrypted communication). In some embodiments, destination domain extraction for secure protocols further includes applying a policy (e.g., a security policy) based on the destination domain to filter traffic at a security device. <i>See, e.g.,</i> Buruganahalli at 7:25-36: In some embodiments, destination domain extraction for secure protocols further includes applying a security policy based on the destination domain to filter traffic at the security device. For example, the security policy can include a malware detection policy, a whitelist/blacklist policy (e.g., in which a session associated with the network communications between the client and the remote server are not decrypted if the destination domain is included in the whitelist, and a session associated with the network communications between the client and the remote server are decrypted if the destination domain is included in the blacklist),

Claim	U.S. Patent No. 9,917,856	Disclosure from Buruganahalli
		and/or a uniform resource locator (URL)/category filtering policy. <i>See, e.g.,</i> Buruganahalli at 7:45-52: In some embodiments, destination domain extraction for secure protocols further includes applying a security policy based on the destination domain to filter traffic at the security device, in which the security policy includes a whitelist/blacklist policy, and in which a session associated with the network communications between the client and the remote server are decrypted if the destination domain is included in a blacklist of the whitelist/blacklist policy. <i>See, e.g.,</i> Buruganahalli at 9:59-64: In some embodiments, the results of the various traffic monitoring techniques using known protocol decoder engine 112, identified traffic engine 114, and unknown protocol decoder engine 116 described above are provided to report and enforce policies engine 120 (e.g., network/routing policies, security policies, and/or firewall policies). <i>See, e.g.,</i> Buruganahalli at 9:64-10:2: For example, firewall policies can be applied to the monitored network traffic using application identification, user identification, and/or other information to match signatures (e.g., file-based, protocol-based, and/or other types/forms of signatures for detecting malware or suspicious behavior). <i>See, e.g.,</i> Buruganahalli at 10:3-9:

Claim	U.S. Patent No. 9,917,856	Disclosure from Buruganahalli
		In some embodiments, firewall 100 also includes a content-ID engine 122, and, in some embodiments, the content-ID engine's identified content is also used by report and enforce policy engine 120 to provide, for example, URL/category filtering, possibly in various combinations with other information, such as application, user, and/or other information, to enforce various security/firewall policies/rules. <i>See, e.g., Buruganahalli at 11:50-12:11:</i> FIG. 3B is another functional block diagram illustrating the use of a firewall for destination domain extraction for secure protocols in accordance with some embodiments. As shown, a client 312 attempts to establish an SSL session with a remote server 316. The client 312 can attempt to use, for example, a secure tunnel with the server 316 to access a network service 318, which can be a network service activity that is in violation of one or more firewall policies/rules implemented by a firewall device 314. However, in this case, the SSL session request is intercepted and detected by the firewall 314. In response, the firewall 314 performs a trusted man-in-the-middle technique by effectively splitting the SSL session between the client 312 and the remote server 316 into two half sessions shown as Session A and Session B in FIG. 38. In Session A, the firewall 314 acts as the remote server 316 such that it is transparent to the client 312 that it is not communicating directly with the remote server 316. Session A traffic is encrypted using the session key S1 associated with the firewall device. In Session B, the firewall 314 acts as the client 312 such that it is transparent to the remote server 316 that it is not communicating directly with the client 312. Session B traffic is encrypted using the session key S2 associated with the firewall device (e.g., the firewall device can store the fingerprint from the remote server in association with that

Claim	U.S. Patent No. 9,917,856	Disclosure from Buruganahalli
		remote server IP address). After the session set-up handshaking is completed for each of Session A and Session B, any data that is communicated from the client 312 to the firewall 314 is decrypted using a session key 51 and is then inspected by the firewall 314. <i>See, e.g., Buruganahalli at 12:34-42:</i> Thus, using these techniques the firewall 314 is between the client 312 and the server 316 and can present itself as server of the session to the client and can present itself as the client of the session to the server. In this manner, the firewall 314 can inspect decrypted traffic and re-encrypt it before forwarding it to the other party. During inspection, if encrypted tunneling traffic is detected, the entire session can either be tom down or various other responsive actions can be performed as described herein. <i>See, e.g., Buruganahalli at 14:15-44:</i> For example, application identification module 612 can recognize a GET request in the received data and conclude that the session requires an HTTP decoder. For each type of protocol, there exists a corresponding decoder 614. In some embodiments, the application identification is performed by an application identification module (e.g., APP-ID engine), and a user identification is performed by another function/engine. Based on the determination made by application identification module 612, the packets are sent to an appropriate decoder 614. Decoder 614 is configured to assemble packets (e.g., which may be received out of order) into the correct order, perform tokenization, and extract out information (e.g., to decode and an initial TLS handshake exchange of a flow in order to extract a hostname to facilitate destination domain extraction for secure protocols). Decoder 614 also performs signature matching to

Claim	U.S. Patent No. 9,917,856	Disclosure from Buruganahalli
		determine what should happen to the packet. SSL encryption engine 616 performs SSL encryption using various techniques as described herein. As also shown, policies 620 are received and stored in the management plane 602. In some embodiments, policy enforcement (e.g., policies can include one or more rules, which can be specified using domain and/or host/ server names, and rules can apply one or more signatures or other matching criteria or heuristics, such as for matching that the decrypted traffic associated with the flow is communicating with a domain that matches the extracted destination domain) using signatures are applied as described herein with respect to various embodiments based on the monitored, decrypted, identified, and decoded session traffic flows. <i>See, e.g., Buruganahalli at 14:45-62:</i> FIG. 7 is a flow diagram for providing destination domain extraction for secure protocols in accordance with some embodiments. At 702, monitoring network communications between a client and a remote server is performed. At 704, determining if the client sends a request to create a secure connection with the remote server is performed (e.g., in which the network communications are initiating a setup for a secure protocol-based connection). At 706, extracting a destination domain from the request to create the secure connection with the remote server is performed. In some embodiments, the secure protocol is a secure sockets layer (SSL) protocol or transport layer security (TLS) protocol, and the destination domain is extracted from the server name indication (SNI) of a client hello message sent from the client to the remote server. In some embodiments, destination domain extraction for secure protocols further includes applying a policy (e.g., a security policy) based on the destination

Claim	U.S. Patent No. 9,917,856	Disclosure from Buruganahalli
		domain to filter traffic using a security device. <i>See, e.g.,</i> Buruganahalli at 14:64-15: FIG. 8 is another flow diagram for providing destination domain extraction for secure protocols in accordance with some embodiments. At 802, intercepting a request to establish an encrypted session from the client to the remote server is performed. At 804, sending a request to establish the encrypted session on behalf of the client to the remote server is performed, and sending an encrypted session response to the client on behalf of the remote server using a session key associated with the firewall device is performed. At 806, decrypting encrypted traffic between the client and remote server to monitor for a request from the client to create a tunnel using the first protocol with the remote server is performed. At 808, allowing the request to create the tunnel is performed. At 810, monitoring decrypted session traffic between the client and the remote server over the tunnel based on one or more firewall policies is performed. At 812, blocking the session traffic using the security device is performed if a violation of a first firewall policy is determined, in which the first firewall policy includes a policy for verifying that the destination domain name extracted from a server name indication (SNI) matches the domain associated with a uniform resource locator (URL) for a session associated with the network communications between the client and the remote server. <i>See, e.g.,</i> Buruganahalli at claim 1: A system for destination domain extraction for secure protocols, comprising:

Claim	U.S. Patent No. 9,917,856	Disclosure from Buruganahalli
		a security device; a processor configured to: monitor network communications between a client and a remote server; determine if the client sends a request to create a secure connection with the remote server, wherein the secure connection utilizes a secure protocol, the secure protocol being a secure sockets layer (SSL) protocol or transport layer security (TLS) protocol; and extract a destination domain from the request to create the secure connection with the remote server, comprising: before the secure connection is created, extract the destination domain from a server name indication (SNI) of a client hello message sent from the client to the remote server, the secure connection including an encrypted message, wherein the extracting of the destination domain from the SNI of the client hello message sent from the client to the remote server comprises to: identify the SNI from the client hello message during a handshaking process for setting the secure connection between the client and the remote server; extract a domain identified in a public certificate sent from the remote server to the client; compare the destination domain and the domain identified in the public certificate; and in the event that the destination domain matches the domain

Claim	U.S. Patent No. 9,917,856	Disclosure from Buruganahalli
		identified in the public certificate, apply a security policy based on the destination domain to filter traffic at the security device, wherein the security policy includes a whitelist/blacklist policy; and a memory coupled to the processor and configured to provide the processor with instructions. <i>See, e.g., Buruganahalli at claim 2:</i> The system recited in claim 1, wherein the client hello message is an unencrypted communication. <i>See, e.g., Buruganahalli at claim 4:</i> The system recited in claim 1, wherein the security policy includes a uniform resource locator (URL)/category filtering related policy. <i>See, e.g., Buruganahalli at claim 7:</i> The system recited in claim 1, wherein a session associated with the network communications between the client and the remote server are decrypted to verify that the destination domain extracted from a server name indication (SNI) matches a domain associated with a uniform resource locator (URL) for the session associated with the network communications between the client and the remote server. <i>See, e.g., Buruganahalli at claim 8:</i> The system recited in claim 1, wherein the processor is further configured to: decrypt a monitored encrypted network communications of a session associated with the network

Claim	U.S. Patent No. 9,917,856	Disclosure from Buruganahalli
		communications between the client and the remote server to verify that the destination domain extracted from a server name indication (SNI) matches a domain associated with a uniform resource locator (URL) for the session associated with the network communications between the client and the remote server. <i>See, e.g., Buruganahalli at claim 9:</i> The system recited in claim 1, wherein the processor is further configured to: - intercept a request to establish an encrypted session from the client to the remote server; - send a request to establish the encrypted session on behalf of the client to the remote server; - send an encrypted session response to the client on behalf of the remote server using a session key associated with the security device; - decrypt encrypted traffic between the client and the remote server to monitor for a request from the client to create a tunnel using a first protocol with the remote server; - allow the request to create the tunnel; - monitor decrypted session traffic between the client and the remote server over the tunnel based on one or more firewall policies; and - block the session traffic using the security device if a violation of a first firewall policy is determined, wherein the first firewall policy includes a policy for verifying that the destination domain extracted from a server name indication (SNI) matches a domain associated with a uniform resource locator (URL) for a session associated with the network communications between the client and the remote server.

Claim	U.S. Patent No. 9,917,856	Disclosure from Buruganahalli
1.8	routing, by the packet-filtering system, filtered packets to a proxy system based on a determination that the filtered packets comprise data that corresponds to the one or more network-threat indicators.	<i>See, e.g.,</i> Burugunahalli at 4:24-35: While performing various decryption techniques as described herein can facilitate applying filtering based on a policy (e.g., a security policy, such as a firewall policy), it would be desirable and efficient to further facilitate policy- based filtering on such data communication flows (e.g., sessions) prior to the set-up of the encrypted data communication for such flows (e.g., to monitor the initial unencrypted/ clear text data communications exchanged as part of an initial handshake to setup a secure connection for a new session between a client and a remote server, which is prior to the establishment of the secure communication channel/tunnel after which any data communications intercepted between the client and the remote server would be encrypted). <i>See, e.g.,</i> Burugunahalli at 5:14-41: In some embodiments, the SNI standard extension is used as disclosed herein as a mechanism to facilitate destination domain extraction for secure protocols without requiring decryption. For example, a firewall can intercept and monitor data communications from a client and a remote server in order to extract the hostname data from the SNI field (e.g., server_name field of the SNI extension), which is sent as an unencrypted data communication from the client (e.g., a web browser executing on the client) to the remote server (e.g., at the start of the handshaking process for setting up a secure TLS communication channel/session between a client and a remote server). This extracted hostname data (e.g., destination domain) can then be used by the firewall to perform an action(s) based on a firewall policy (e.g., the domain name can be extracted by parsing the client hello message to extract the host name from the server_name field of the SNI extension, which can

Claim	U.S. Patent No. 9,917,856	Disclosure from Buruganahalli
		then be used to apply firewall policies or take responsive actions based on this information without having to wait for data to be decrypted), as further described herein with respect to various embodiments. Also, such SNI information is available in the first packet of this handshaking process, such that a policy (e.g., a security or firewall policy using the hostname information extracted from the SNI information) can be applied sooner based on this early extraction of the hostname from the SNI information such that the rest of the stream can be forwarded without any decryption resulting in less state to track thereby enhancing performance (e.g., of the security device, such as a firewall). <i>See, e.g., Burugunahalli at 5:54-58:</i> For example, if a new flow is determined to violate a policy prior to the set-up of the encrypted data communication for that flow between a client and a remote server, then the flow can be blocked and decryption is not required. As an example, Bob who is a user (e.g., an employee) of ACME Company may attempt to logon using a web browser executing on his desktop office computer to a remote server that is associated with the online banking site (e.g., web site) of Banking Corporation. If the firewall policy of ACME Company has white listed the domain associated with the Banking Corporation as a trusted domain, then Bob's connection (e.g., an SSL/TLS session) with the web site for the Banking Corporation can be allowed using various techniques described herein. <i>See, e.g., Burugunahalli at 6:6-11:</i> For example, if a flow is initially determined to be permissible based on a policy prior to the set-up of the encrypted data communication for that flow between a client and a remote server,

Claim	U.S. Patent No. 9,917,856	Disclosure from Buruganahalli
		then the flow can be allowed, and a policy may require that the session be decrypted for further analysis and further potential filtering based on the policy. <i>See, e.g.,</i> Buruganahalli at 6:66-7:15: In some embodiments, destination domain extraction for secure protocols includes monitoring network communications between a client and a remote server; determining if the client sends a request to create a secure connection with the remote server (e.g., in which the network communications are initiating a setup for a secure protocol-based connection); and extracting a destination domain from the request to create the secure connection with the remote server. In some embodiments, the secure protocol is a secure sockets layer (SSL) protocol or transport layer security (TLS) protocol, and the destination domain is extracted from the server name indication (SNI) of a client hello message sent from the client to the remote server (e.g., the client hello message is an unencrypted communication). In some embodiments, destination domain extraction for secure protocols further includes applying a policy (e.g., a security policy) based on the destination domain to filter traffic at a security device. <i>See, e.g.,</i> Buruganahalli at 7:25-36: In some embodiments, destination domain extraction for secure protocols further includes applying a security policy based on the destination domain to filter traffic at the security device. For example, the security policy can include a malware detection policy, a whitelist/blacklist policy (e.g., in which a session associated with the network communications between the client and the remote server are not decrypted if the destination domain is

Claim	U.S. Patent No. 9,917,856	Disclosure from Buruganahalli
		included in the whitelist, and a session associated with the network communications between the client and the remote server are decrypted if the destination domain is included in the blacklist), and/or a uniform resource locator (URL)/category filtering policy. <i>See, e.g.,</i> Buruganahalli at 7:45-52: In some embodiments, destination domain extraction for secure protocols further includes applying a security policy based on the destination domain to filter traffic at the security device, in which the security policy includes a whitelist/blacklist policy, and in which a session associated with the network communications between the client and the remote server are decrypted if the destination domain is included in a blacklist of the whitelist/blacklist policy. <i>See, e.g.,</i> Buruganahalli at 9:59-64: In some embodiments, the results of the various traffic monitoring techniques using known protocol decoder engine 112, identified traffic engine 114, and unknown protocol decoder engine 116 described above are provided to report and enforce policies engine 120 (e.g., network/routing policies, security policies, and/or firewall policies). <i>See, e.g.,</i> Buruganahalli at 9:64-10:2: For example, firewall policies can be applied to the monitored network traffic using application identification, user identification, and/or other information to match signatures (e.g., file-based, protocol-based, and/or other types/forms of signatures for detecting

Claim	U.S. Patent No. 9,917,856	Disclosure from Buruganahalli
		malware or suspicious behavior). <i>See, e.g.,</i> Buruganahalli at 10:3-9: In some embodiments, firewall 100 also includes a content-ID engine 122, and, in some embodiments, the content-ID engine's identified content is also used by report and enforce policy engine 120 to provide, for example, URL/category filtering, possibly in various combinations with other information, such as application, user, and/or other information, to enforce various security/firewall policies/rules. <i>See, e.g.,</i> Buruganahalli at 11:50-12:11: FIG. 3B is another functional block diagram illustrating the use of a firewall for destination domain extraction for secure protocols in accordance with some embodiments. As shown, a client 312 attempts to establish an SSL session with a remote server 316. The client 312 can attempt to use, for example, a secure tunnel with the server 316 to access a network service 318, which can be a network service activity that is in violation of one or more firewall policies/rules implemented by a firewall device 314. However, in this case, the SSL session request is intercepted and detected by the firewall 314. In response, the firewall 314 performs a trusted man-in-the-middle technique by effectively splitting the SSL session between the client 312 and the remote server 316 into two half sessions shown as Session A and Session B in FIG. 38. In Session A, the firewall 314 acts as the remote server 316 such that it is transparent to the client 312 that it is not communicating directly with the remote server 316. Session A traffic is encrypted using the session key 51 associated with the firewall device. In Session B, the firewall 314 acts as the client 312 such that it is transparent to

Claim	U.S. Patent No. 9,917,856	Disclosure from Buruganahalli
		the remote server 316 that it is not communicating directly with the client 312. Session B traffic is encrypted using the session key S2 associated with the firewall device (e.g., the firewall device can store the fingerprint from the remote server in association with that remote server IP address). After the session set-up handshaking is completed for each of Session A and Session B, any data that is communicated from the client 312 to the firewall 314 is decrypted using a session key 51 and is then inspected by the firewall 314. <i>See, e.g., Buruganahalli at 12:34-42:</i> Thus, using these techniques the firewall 314 is between the client 312 and the server 316 and can present itself as server of the session to the client and can present itself as the client of the session to the server. In this manner, the firewall 314 can inspect decrypted traffic and re-encrypt it before forwarding it to the other party. During inspection, if encrypted tunneling traffic is detected, the entire session can either be tom down or various other responsive actions can be performed as described herein. <i>See, e.g., Buruganahalli at 14:15-44:</i> For example, application identification module 612 can recognize a GET request in the received data and conclude that the session requires an HTTP decoder. For each type of protocol, there exists a corresponding decoder 614. In some embodiments, the application identification is performed by an application identification module (e.g., APP-ID engine), and a user identification is performed by another function/engine. Based on the determination made by application identification module 612, the packets are sent to an appropriate decoder 614. Decoder 614 is configured to assemble packets (e.g., which may be received out of order) into the correct

Claim	U.S. Patent No. 9,917,856	Disclosure from Buruganahalli
		order, perform tokenization, and extract out information (e.g., to decode and an initial TLS handshake exchange of a flow in order to extract a hostname to facilitate destination domain extraction for secure protocols). Decoder 614 also performs signature matching to determine what should happen to the packet. SSL encryption engine 616 performs SSL encryption using various techniques as described herein. As also shown, policies 620 are received and stored in the management plane 602. In some embodiments, policy enforcement (e.g., policies can include one or more rules, which can be specified using domain and/or host/ server names, and rules can apply one or more signatures or other matching criteria or heuristics, such as for matching that the decrypted traffic associated with the flow is communicating with a domain that matches the extracted destination domain) using signatures are applied as described herein with respect to various embodiments based on the monitored, decrypted, identified, and decoded session traffic flows. <i>See, e.g., Buruganahalli at 14:45-62:</i> FIG. 7 is a flow diagram for providing destination domain extraction for secure protocols in accordance with some embodiments. At 702, monitoring network communications between a client and a remote server is performed. At 704, determining if the client sends a request to create a secure connection with the remote server is performed (e.g., in which the network communications are initiating a setup for a secure protocol-based connection). At 706, extracting a destination domain from the request to create the secure connection with the remote server is performed. In some embodiments, the secure protocol is a secure sockets layer (SSL) protocol or transport layer security (TLS) protocol, and the destination domain is extracted from the server name indication (SNI) of a client hello message

Claim	U.S. Patent No. 9,917,856	Disclosure from Buruganahalli
		sent from the client to the remote server. In some embodiments, destination domain extraction for secure protocols further includes applying a policy (e.g., a security policy) based on the destination domain to filter traffic using a security device. <i>See, e.g., Buruganahalli at 14:64-15:</i> FIG. 8 is another flow diagram for providing destination domain extraction for secure protocols in accordance with some embodiments. At 802, intercepting a request to establish an encrypted session from the client to the remote server is performed. At 804, sending a request to establish the encrypted session on behalf of the client to the remote server is performed, and sending an encrypted session response to the client on behalf of the remote server using a session key associated with the firewall device is performed. At 806, decrypting encrypted traffic between the client and remote server to monitor for a request from the client to create a tunnel using the first protocol with the remote server is performed. At 808, allowing the request to create the tunnel is performed. At 810, monitoring decrypted session traffic between the client and the remote server over the tunnel based on one or more firewall policies is performed. At 812, block- ing the session traffic using the security device is performed if a violation of a first firewall policy is determined, in which the first firewall policy includes a policy for verifying that the destination domain name extracted from a server name indication (SNI) matches the domain associated with a uniform resource locator (URL) for a session associated with the network communications between the client and the remote server. <i>See, e.g., Buruganahalli at claim 1:</i>

Claim	U.S. Patent No. 9,917,856	Disclosure from Buruganahalli
		A system for destination domain extraction for secure protocols, comprising: a security device; a processor configured to: monitor network communications between a client and a remote server; determine if the client sends a request to create a secure connection with the remote server, wherein the secure connection utilizes a secure protocol, the secure protocol being a secure sockets layer (SSL) protocol or transport layer security (TLS) protocol; and extract a destination domain from the request to create the secure connection with the remote server, comprising: before the secure connection is created, extract the destination domain from a server name indication (SNI) of a client hello message sent from the client to the remote server, the secure connection including an encrypted message, wherein the extracting of the destination domain from the SNI of the client hello message sent from the client to the remote server comprises to: identify the SNI from the client hello message during a handshaking process for setting the secure connection between the client and the remote server; extract a domain identified in a public certificate sent from the remote server to the client; compare the destination domain and the domain identified in the

Claim	U.S. Patent No. 9,917,856	Disclosure from Buruganahalli
		public certificate; and in the event that the destination domain matches the domain identified in the public certificate, apply a security policy based on the destination domain to filter traffic at the security device, wherein the security policy includes a whitelist/blacklist policy; and a memory coupled to the processor and configured to provide the processor with instructions. <i>See, e.g., Buruganahalli at claim 2:</i> The system recited in claim 1, wherein the client hello message is an unencrypted communication. <i>See, e.g., Buruganahalli at claim 4:</i> The system recited in claim 1, wherein the security policy includes a uniform resource locator (URL)/category filtering related policy. <i>See, e.g., Buruganahalli at claim 7:</i> The system recited in claim 1, wherein a session associated with the network communications between the client and the remote server are decrypted to verify that the destination domain extracted from a server name indication (SNI) matches a domain associated with a uniform resource locator (URL) for the session associated with the network communications between the client and the remote server. <i>See, e.g., Buruganahalli at claim 8:</i>

Claim	U.S. Patent No. 9,917,856	Disclosure from Buruganahalli
		The system recited in claim 1, wherein the processor is further configured to: decrypt a monitored encrypted network communications of a session associated with the network communications between the client and the remote server to verify that the destination domain extracted from a server name indication (SNI) matches a domain associated with a uniform resource locator (URL) for the session associated with the network communications between the client and the remote server. <i>See, e.g., Buruganahalli at claim 9:</i> The system recited in claim 1, wherein the processor is further configured to: - intercept a request to establish an encrypted session from the client to the remote server; - send a request to establish the encrypted session on behalf of the client to the remote server; - send an encrypted session response to the client on behalf of the remote server using a session key associated with the security device; - decrypt encrypted traffic between the client and the remote server to monitor for a request from the client to create a tunnel using a first protocol with the remote server; - allow the request to create the tunnel; - monitor decrypted session traffic between the client and the remote server over the tunnel based on one or more firewall policies; and - block the session traffic using the security device if a violation of a first firewall policy is determined, wherein the first firewall policy includes a policy for verifying that the destination domain extracted from a server name indication (SNI) matches a domain associated with a uniform resource locator (URL) for a session associated with the network communications between the client and the

Claim	U.S. Patent No. 9,917,856	Disclosure from Buruganahalli
		remote server.
18.0	The method of claim 1, wherein:	<i>See, e.g.</i> , disclosures at Claim 1.0.
18.1	the packets comprising unencrypted data comprise one or more packets comprising one or more handshake messages configured to establish an encrypted communication session between a client and a server, the method further comprising:	<i>See, e.g.</i>, Buruganahalli at 4:24-35: While performing various decryption techniques as described herein can facilitate applying filtering based on a policy (e.g., a security policy, such as a firewall policy), it would be desirable and efficient to further facilitate policy- based filtering on such data communication flows (e.g., sessions) prior to the set-up of the encrypted data communication for such flows (e.g., to monitor the initial unencrypted/ clear text data communications exchanged as part of an initial handshake to setup a secure connection for a new session between a client and a remote server, which is prior to the establishment of the secure communication channel/tunnel after which any data communications intercepted between the client and the remote server would be encrypted). <i>See, e.g.</i>, Buruganahalli at 5:14-41: In some embodiments, the SNI standard extension is used as disclosed herein as a mechanism to facilitate destination domain extraction for secure protocols without requiring decryption. For example, a firewall can intercept and monitor data communications from a client and a remote server in order to extract the hostname data from the SNI field (e.g., server_name field of the SNI extension), which is sent as an unencrypted data communication from the client (e.g., a web browser executing on the client) to the remote server (e.g., at the start of the handshaking process for setting up a secure TLS communication channel/session between a client and a remote server). This extracted hostname data (e.g., destination domain) can then be used by the firewall to perform an

Claim	U.S. Patent No. 9,917,856	Disclosure from Buruganahalli
		action(s) based on a firewall policy (e.g., the domain name can be extracted by parsing the client hello message to extract the host name from the server_name field of the SNI extension, which can then be used to apply firewall policies or take responsive actions based on this information without having to wait for data to be decrypted), as further described herein with respect to various embodiments. Also, such SNI information is available in the first packet of this handshaking process, such that a policy (e.g., a security or firewall policy using the hostname information extracted from the SNI information) can be applied sooner based on this early extraction of the hostname from the SNI information such that the rest of the stream can be forwarded without any decryption resulting in less state to track thereby enhancing performance (e.g., of the security device, such as a firewall). <i>See, e.g., Buruganahalli at 10:36-55:</i> FIG. 3A is a functional block diagram illustrating an SSL/ TLS session passing through a firewall that can implement destination domain extraction for secure protocols in accordance with some embodiments. In that session, a typical firewall can initially just inspect the handshake traffic (e.g., the initial negotiation to setup a secure communication channel/tunnel between a client and a remote server), as all the application data is encrypted. As shown, a client 302 establishes a secure tunnel session (e.g., creates an SSL tunnel) with a remote server 306. The client 302 can use the secure tunnel with the server 306 to access a network service 308, which can be a network service activity that is in violation of one or more firewall policies/rules implemented by the fire- wall device 304 (e.g., a policy that includes requirements or rules related to destination domains that may be used for secure protocol communications). As the secure tunnel session traffic is encrypted

Claim	U.S. Patent No. 9,917,856	Disclosure from Buruganahalli
		and passes through a firewall 304, the firewall 304 typically cannot decrypt the encrypted secure tunnel session traffic and, thus, cannot detect such firewall policy/rule violation(s). <i>See, e.g., Buruganahalli at 11:11-34:</i> Below is a simplified example of such a TLS handshake exchange (e.g., handshake/negotiation) between a client (C) and a remote server (S) using the SNI approach: Step 1. C: (TLS Handshake) Hello, I support XYZ Encryption, and I am trying to connect to 'website.example.com'. Step 2. S: (TLS Handshake) Hi there, here is my Public Certificate, and let's use this encryption algorithm. Step 3. C: (TLS Handshake) Agreed. Step 4. C: (Encrypted) HTTP Request. Step 5. S: (Encrypted) HTTP Reply. As shown in the above example TLS handshake exchange, the client provides a hostname identification in Step 1. In particular, the client indicates in this hello message which hostname it wants to setup this secure connection with so that the server then knows which public certificate to send back to the client. As discussed above, the firewall can intercept and decode (e.g., parse) this TLS handshake exchange to extract the hostname from the client hello message in order to apply various policies based on the destination domain. As also discussed above, the firewall can also verify whether the extracted hostname matches the hostname associated with the provided Public Certificate that was sent in Step 2.

Claim	U.S. Patent No. 9,917,856	Disclosure from Buruganahalli
		<i>See, e.g.,</i> Buruganahalli at 14:3:11: As described herein, an initial TLS handshake exchange of that flow can be parsed to extract a hostname to facilitate destination domain extraction for secure protocols (e.g., and to apply one or more policies if applicable based on the extracted destination domain) using various techniques described herein. If applicable (e.g., using one or more policies based on the extracted destination domain), SSL decryption is applied by SSL decryption engine 610 using various techniques as described herein. <i>See, e.g.,</i> Buruganahalli at 14:15-44: For example, application identification module 612 can recognize a GET request in the received data and conclude that the session requires an HTTP decoder. For each type of protocol, there exists a corresponding decoder 614. In some embodiments, the application identification is performed by an application identification module (e.g., APP-ID engine), and a user identification is performed by another function/engine. Based on the determination made by application identification module 612, the packets are sent to an appropriate decoder 614. Decoder 614 is configured to assemble packets (e.g., which may be received out of order) into the correct order, perform tokenization, and extract out information (e.g., to decode and an initial TLS handshake exchange of a flow in order to extract a hostname to facilitate destination domain extraction for secure protocols). Decoder 614 also performs signature matching to determine what should happen to the packet. SSL encryption engine 616 performs SSL encryption using various techniques as described herein. As also shown, policies 620 are received and stored in the management plane 602. In some embodiments, policy enforcement (e.g., policies can include one or more rules, which

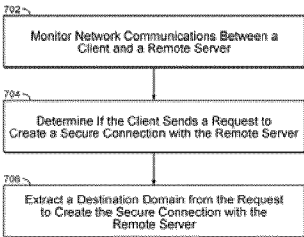
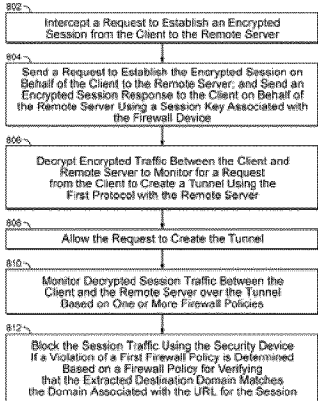
Claim	U.S. Patent No. 9,917,856	Disclosure from Buruganahalli
		can be specified using domain and/or host/ server names, and rules can apply one or more signatures or other matching criteria or heuristics, such as for matching that the decrypted traffic associated with the flow is communicating with a domain that matches the extracted destination domain) using signatures are applied as described herein with respect to various embodiments based on the monitored, decrypted, identified, and decoded session traffic flows. <i>See, e.g., Buruganahalli at 14:64-15:</i> FIG. 8 is another flow diagram for providing destination domain extraction for secure protocols in accordance with some embodiments. At 802, intercepting a request to establish an encrypted session from the client to the remote server is performed. At 804, sending a request to establish the encrypted session on behalf of the client to the remote server is performed, and sending an encrypted session response to the client on behalf of the remote server using a session key associated with the firewall device is performed. At 806, decrypting encrypted traffic between the client and remote server to monitor for a request from the client to create a tunnel using the first protocol with the remote server is performed. At 808, allowing the request to create the tunnel is performed. At 810, monitoring decrypted session traffic between the client and the remote server over the tunnel based on one or more firewall policies is performed. At 812, block- ing the session traffic using the security device is performed if a violation of a first firewall policy is determined, in which the first firewall policy includes a policy for verifying that the destination domain name extracted from a server name indication (SNI) matches the domain associated with a uniform resource locator (URL) for a session associated with the network communications between the client

Claim	U.S. Patent No. 9,917,856	Disclosure from Buruganahalli
		and the remote server. <i>See, e.g.,</i> Buruganahalli at claim 1: A system for destination domain extraction for secure protocols, comprising: a security device; a processor configured to: monitor network communications between a client and a remote server; determine if the client sends a request to create a secure connection with the remote server, wherein the secure connection utilizes a secure protocol, the secure protocol being a secure sockets layer (SSL) protocol or transport layer security (TLS) protocol; and extract a destination domain from the request to create the secure connection with the remote server, comprising: before the secure connection is created, extract the destination domain from a server name indication (SNI) of a client hello message sent from the client to the remote server, the secure connection including an encrypted message, wherein the extracting of the destination domain from the SNI of the client hello message sent from the client to the remote server comprises to: identify the SNI from the client hello message during a handshaking process for setting the secure connection between the client and the remote server;

Claim	U.S. Patent No. 9,917,856	Disclosure from Buruganahalli
		extract a domain identified in a public certificate sent from the remote server to the client; compare the destination domain and the domain identified in the public certificate; and in the event that the destination domain matches the domain identified in the public certificate, apply a security policy based on the destination domain to filter traffic at the security device, wherein the security policy includes a whitelist/blacklist policy; and a memory coupled to the processor and configured to provide the processor with instructions. <i>See, e.g.,</i> Buruganahalli at claim 9: The system recited in claim 1, wherein the processor is further configured to: intercept a request to establish an encrypted session from the client to the remote server; send a request to establish the encrypted session on behalf of the client to the remote server; send an encrypted session response to the client on behalf of the remote server using a session key associated with the security device; decrypt encrypted traffic between the client and the remote server to monitor for a request from the client to create a tunnel using a

Claim	U.S. Patent No. 9,917,856	Disclosure from Buruganahalli
		first protocol with the remote server; allow the request to create the tunnel; monitor decrypted session traffic between the client and the remote server over the tunnel based on one or more firewall policies; and block the session traffic using the security device if a violation of a first firewall policy is determined, wherein the first firewall policy includes a policy for verifying that the destination domain extracted from a server name indication (SNI) matches a domain associated with a uniform resource locator (URL) for a session associated with the network communications between the client and the remote server. <i>See, e.g., Buruganahalli at claim 11:</i> A method for destination domain extraction for secure protocols, comprising: monitoring network communications between a client and a remote server using a security device; determining if the client sends a request to create a secure connection with the remote server, wherein the secure connection utilizes a secure protocol, the secure protocol being a secure sockets layer (SSL) protocol or transport layer security (TLS) protocol; and extracting a destination domain from the request to create the secure connection with the remote server, comprising: before the secure connection is created, extracting the destination

Claim	U.S. Patent No. 9,917,856	Disclosure from Buruganahalli
		domain from a server name indication (SNI) of a client hello message sent from the client to the remote server, the secure connection including an encrypted message, wherein the extracting of the destination domain from the SNI of the client hello message sent from the client to the remote server comprises: identifying the SNI from the client hello message during a handshaking process for setting the secure connection between the client and the remote server; extracting a domain identified in a public certificate sent from the remote server to the client; comparing the destination domain and the domain identified in the public certificate; and in the event that the destination domain matches the domain identified in the public certificate, applying a security policy based on the destination domain to filter traffic at the security device, wherein the security policy includes a whitelist/blacklist policy. <i>See, e.g., Buruganahalli at Fig. 7:</i>

Claim	U.S. Patent No. 9,917,856	Disclosure from Buruganahalli
		 FIG. 7 <i>See, e.g., Buruganahalli at Fig. 8:</i>  FIG. 8
18.2	determining that the one or more handshake messages comprise the domain name identified as the network threat.	<i>See, e.g., Buruganahalli at 4:24-35:</i> While performing various decryption techniques as described herein can facilitate applying filtering based on a policy (e.g., a security policy, such as a firewall policy), it would be desirable and efficient to further facilitate policy- based filtering on such data

Claim	U.S. Patent No. 9,917,856	Disclosure from Buruganahalli
		communication flows (e.g., sessions) prior to the set-up of the encrypted data communication for such flows (e.g., to monitor the initial unencrypted/ clear text data communications exchanged as part of an initial handshake to setup a secure connection for a new session between a client and a remote server, which is prior to the establishment of the secure communication channel/tunnel after which any data communications intercepted between the client and the remote server would be encrypted). <i>See, e.g., Buruganahalli at 5:14-41:</i> In some embodiments, the SNI standard extension is used as disclosed herein as a mechanism to facilitate destination domain extraction for secure protocols without requiring decryption. For example, a firewall can intercept and monitor data communications from a client and a remote server in order to extract the hostname data from the SNI field (e.g., server_name field of the SNI extension), which is sent as an unencrypted data communication from the client (e.g., a web browser executing on the client) to the remote server (e.g., at the start of the handshaking process for setting up a secure TLS communication channel/session between a client and a remote server). This extracted hostname data (e.g., destination domain) can then be used by the firewall to perform an action(s) based on a firewall policy (e.g., the domain name can be extracted by parsing the client hello message to extract the host name from the server_name field of the SNI extension, which can then be used to apply firewall policies or take responsive actions based on this information without having to wait for data to be decrypted), as further described herein with respect to various embodiments. Also, such SNI information is available in the first packet of this handshaking process, such that a policy (e.g., a security or firewall policy using the hostname information

Claim	U.S. Patent No. 9,917,856	Disclosure from Burugunahalli
		extracted from the SNI information) can be applied sooner based on this early extraction of the hostname from the SNI information such that the rest of the stream can be forwarded without any decryption resulting in less state to track thereby enhancing performance (e.g., of the security device, such as a firewall). <i>See, e.g., Burugunahalli at 10:36-55:</i> FIG. 3A is a functional block diagram illustrating an SSL/ TLS session passing through a firewall that can implement destination domain extraction for secure protocols in accordance with some embodiments. In that session, a typical firewall can initially just inspect the handshake traffic (e.g., the initial negotiation to setup a secure communication channel/tunnel between a client and a remote server), as all the application data is encrypted. As shown, a client 302 establishes a secure tunnel session (e.g., creates an SSL tunnel) with a remote server 306. The client 302 can use the secure tunnel with the server 306 to access a network service 308, which can be a network service activity that is in violation of one or more firewall policies/rules implemented by the fire- wall device 304 (e.g., a policy that includes requirements or rules related to destination domains that may be used for secure protocol communications). As the secure tunnel session traffic is encrypted and passes through a firewall 304, the firewall 304 typically cannot decrypt the encrypted secure tunnel session traffic and, thus, cannot detect such firewall policy/rule violation(s). <i>See, e.g., Burugunahalli at 11:11-34:</i> Below is a simplified example of such a TLS handshake exchange (e.g., handshake/negotiation) between a client (C) and a remote

Claim	U.S. Patent No. 9,917,856	Disclosure from Buruganahalli
		server (S) using the SNI approach: Step 1. C: (TLS Handshake) Hello, I support XYZ Encryption, and I am trying to connect to 'website.example.com'. Step 2. S: (TLS Handshake) Hi there, here is my Public Certificate, and let's use this encryption algorithm. Step 3. C: (TLS Handshake) Agreed. Step 4. C: (Encrypted) HTTP Request. Step 5. S: (Encrypted) HTTP Reply. As shown in the above example TLS handshake exchange, the client provides a hostname identification in Step 1. In particular, the client indicates in this hello message which hostname it wants to setup this secure connection with so that the server then knows which public certificate to send back to the client. As discussed above, the firewall can intercept and decode (e.g., parse) this TLS handshake exchange to extract the hostname from the client hello message in order to apply various policies based on the destination domain. As also discussed above, the firewall can also verify whether the extracted hostname matches the hostname associated with the provided Public Certificate that was sent in Step 2. <i>See, e.g., Buruganahalli at 14:3:11:</i> As described herein, an initial TLS handshake exchange of that flow can be parsed to extract a hostname to facilitate destination domain extraction for secure protocols (e.g., and to apply one or more policies if applicable based on the extracted destination domain) using various techniques described herein. If applicable (e.g., using one or more policies based on the extracted destination domain), SSL decryption is applied by SSL decryption engine 610

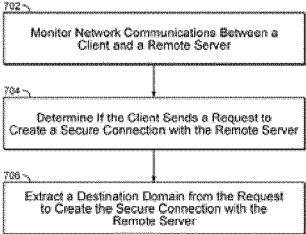
Claim	U.S. Patent No. 9,917,856	Disclosure from Buruganahalli
		using various techniques as described herein. <i>See, e.g.,</i> Buruganahalli at 14:15-44: For example, application identification module 612 can recognize a GET request in the received data and conclude that the session requires an HTTP decoder. For each type of protocol, there exists a corresponding decoder 614. In some embodiments, the application identification is performed by an application identification module (e.g., APP-ID engine), and a user identification is performed by another function/engine. Based on the determination made by application identification module 612, the packets are sent to an appropriate decoder 614. Decoder 614 is configured to assemble packets (e.g., which may be received out of order) into the correct order, perform tokenization, and extract out information (e.g., to decode and an initial TLS handshake exchange of a flow in order to extract a hostname to facilitate destination domain extraction for secure protocols). Decoder 614 also performs signature matching to determine what should happen to the packet. SSL encryption engine 616 performs SSL encryption using various techniques as described herein. As also shown, policies 620 are received and stored in the management plane 602. In some embodiments, policy enforcement (e.g., policies can include one or more rules, which can be specified using domain and/or host/ server names, and rules can apply one or more signatures or other matching criteria or heuristics, such as for matching that the decrypted traffic associated with the flow is communicating with a domain that matches the extracted destination domain) using signatures are applied as described herein with respect to various embodiments based on the monitored, decrypted, identified, and decoded session traffic flows.

Claim	U.S. Patent No. 9,917,856	Disclosure from Buruganahalli
		<i>See, e.g.,</i> Buruganahalli at 14:64-15: FIG. 8 is another flow diagram for providing destination domain extraction for secure protocols in accordance with some embodiments. At 802, intercepting a request to establish an encrypted session from the client to the remote server is performed. At 804, sending a request to establish the encrypted session on behalf of the client to the remote server is performed, and sending an encrypted session response to the client on behalf of the remote server using a session key associated with the firewall device is performed. At 806, decrypting encrypted traffic between the client and remote server to monitor for a request from the client to create a tunnel using the first protocol with the remote server is performed. At 808, allowing the request to create the tunnel is performed. At 810, monitoring decrypted session traffic between the client and the remote server over the tunnel based on one or more firewall policies is performed. At 812, blocking the session traffic using the security device is performed if a violation of a first firewall policy is determined, in which the first firewall policy includes a policy for verifying that the destination domain name extracted from a server name indication (SNI) matches the domain associated with a uniform resource locator (URL) for a session associated with the network communications between the client and the remote server. <i>See, e.g.,</i> Buruganahalli at claim 1: A system for destination domain extraction for secure protocols, comprising: a security device;

Claim	U.S. Patent No. 9,917,856	Disclosure from Buruganahalli
		a processor configured to: monitor network communications between a client and a remote server; determine if the client sends a request to create a secure connection with the remote server, wherein the secure connection utilizes a secure protocol, the secure protocol being a secure sockets layer (SSL) protocol or transport layer security (TLS) protocol; and extract a destination domain from the request to create the secure connection with the remote server, comprising: before the secure connection is created, extract the destination domain from a server name indication (SNI) of a client hello message sent from the client to the remote server, the secure connection including an encrypted message, wherein the extracting of the destination domain from the SNI of the client hello message sent from the client to the remote server comprises to: identify the SNI from the client hello message during a handshaking process for setting the secure connection between the client and the remote server; extract a domain identified in a public certificate sent from the remote server to the client; compare the destination domain and the domain identified in the public certificate; and in the event that the destination domain matches the domain identified in the public certificate, apply a security policy based on

Claim	U.S. Patent No. 9,917,856	Disclosure from Buruganahalli
		the destination domain to filter traffic at the security device, wherein the security policy includes a whitelist/blacklist policy; and a memory coupled to the processor and configured to provide the processor with instructions. <i>See, e.g., Buruganahalli at claim 9:</i> The system recited in claim 1, wherein the processor is further configured to: intercept a request to establish an encrypted session from the client to the remote server; send a request to establish the encrypted session on behalf of the client to the remote server; send an encrypted session response to the client on behalf of the remote server using a session key associated with the security device; decrypt encrypted traffic between the client and the remote server to monitor for a request from the client to create a tunnel using a first protocol with the remote server; allow the request to create the tunnel; monitor decrypted session traffic between the client and the remote server over the tunnel based on one or more firewall policies; and block the session traffic using the security device if a violation of a first firewall policy is determined, wherein the first firewall policy

Claim	U.S. Patent No. 9,917,856	Disclosure from Buruganahalli
		includes a policy for verifying that the destination domain extracted from a server name indication (SNI) matches a domain associated with a uniform resource locator (URL) for a session associated with the network communications between the client and the remote server. <i>See, e.g., Buruganahalli at claim 11:</i> A method for destination domain extraction for secure protocols, comprising: monitoring network communications between a client and a remote server using a security device; determining if the client sends a request to create a secure connection with the remote server, wherein the secure connection utilizes a secure protocol, the secure protocol being a secure sockets layer (SSL) protocol or transport layer security (TLS) protocol; and extracting a destination domain from the request to create the secure connection with the remote server, comprising: before the secure connection is created, extracting the destination domain from a server name indication (SNI) of a client hello message sent from the client to the remote server, the secure connection including an encrypted message, wherein the extracting of the destination domain from the SNI of the client hello message sent from the client to the remote server comprises: identifying the SNI from the client hello message during a handshaking process for setting the secure connection between the

Claim	U.S. Patent No. 9,917,856	Disclosure from Buruganahalli
		client and the remote server; extracting a domain identified in a public certificate sent from the remote server to the client; comparing the destination domain and the domain identified in the public certificate; and in the event that the destination domain matches the domain identified in the public certificate, applying a security policy based on the destination domain to filter traffic at the security device, wherein the security policy includes a whitelist/blacklist policy. <i>See, e.g., Buruganahalli at Fig. 7:</i>  <pre> graph TD 702[702 Monitor Network Communications Between a Client and a Remote Server] --> 704[704 Determine If the Client Sends a Request to Create a Secure Connection with the Remote Server] 704 --> 706[706 Extract a Destination Domain from the Request to Create the Secure Connection with the Remote Server] </pre> FIG. 7 <i>See, e.g., Buruganahalli at Fig. 8:</i>

Claim	U.S. Patent No. 9,917,856	Disclosure from Buruganahalli
		<pre> graph TD 802[802 Intercept a Request to Establish an Encrypted Session from the Client to the Remote Server] --> 804[804 Send a Request to Establish the Encrypted Session on Behalf of the Client to the Remote Server, and Send an Encrypted Session Response to the Client on Behalf of the Remote Server Using a Session Key Associated with the Firewall Device] 804 --> 806[806 Decrypt Encrypted Traffic Between the Client and Remote Server to Monitor for a Request from the Client to Create a Tunnel Using the First Protocol with the Remote Server] 806 --> 808[808 Allow the Request to Create the Tunnel] 808 --> 810[810 Monitor Decrypted Session Traffic Between the Client and the Remote Server over the Tunnel Based on One or More Firewall Policies] 810 --> 812[812 Block the Session Traffic Using the Security Device If a Violation of a First Firewall Policy is Determined Based on a Firewall Policy for Verifying that the Extracted Destination Domain Matches the Domain Associated with the URL for the Session] </pre> FIG. 8
24.0	24. A packet-filtering system comprising:	<i>See, e.g., disclosures at Claim 1.1.</i>
24.1	at least one hardware processor; and	<i>See, e.g., disclosures at Claim 1.1.</i>
24.2	memory storing instructions that when executed by the at least one hardware processor cause the packet-filtering system to:	<i>See, e.g., disclosures at Claim 1.1.</i>
24.3	receive data indicating a plurality of network-threat indicators, wherein at least one of the plurality of network-threat indicators comprise a domain name identified as a network threat;	<i>See, e.g., disclosures at Claim 1.1.</i> <i>See, e.g., Buruganahalli at 6:66-7:15:</i> In some embodiments, destination domain extraction for secure

Claim	U.S. Patent No. 9,917,856	Disclosure from Buruganahalli
		protocols includes monitoring network communications between a client and a remote server; determining if the client sends a request to create a secure connection with the remote server (e.g., in which the network communications are initiating a setup for a secure protocol-based connection); and extracting a destination domain from the request to create the secure connection with the remote server. In some embodiments, the secure protocol is a secure sockets layer (SSL) protocol or transport layer security (TLS) protocol, and the destination domain is extracted from the server name indication (SNI) of a client hello message sent from the client to the remote server (e.g., the client hello message is an unencrypted communication). In some embodiments, destination domain extraction for secure protocols further includes applying a policy (e.g., a security policy) based on the destination domain to filter traffic at a security device. <i>See, e.g., Buruganahalli at 7:25-36:</i> In some embodiments, destination domain extraction for secure protocols further includes applying a security policy based on the destination domain to filter traffic at the security device. For example, the security policy can include a malware detection policy, a whitelist/blacklist policy (e.g., in which a session associated with the network communications between the client and the remote server are not decrypted if the destination domain is included in the whitelist, and a session associated with the network communications between the client and the remote server are decrypted if the destination domain is included in the blacklist), and/or a uniform resource locator (URL)/category filtering policy. <i>See, e.g., Buruganahalli at 7:45-52:</i>

Claim	U.S. Patent No. 9,917,856	Disclosure from Buruganahalli
		In some embodiments, destination domain extraction for secure protocols further includes applying a security policy based on the destination domain to filter traffic at the security device, in which the security policy includes a whitelist/blacklist policy, and in which a session associated with the network communications between the client and the remote server are decrypted if the destination domain is included in a blacklist of the whitelist/blacklist policy. <i>See, e.g., Buruganahalli at 7:62-8:8:</i> In some embodiments, destination domain extraction for secure protocols further includes decrypting the monitored encrypted network communications of a session associated with the network communications between the client and the remote server to verify that the destination domain name extracted from a server name indication matches the domain associated with a URL for the session associated with the network communications between the client and the remote server. For example, this technique can be used in order to detect malware that attempts to fool a hosting site by including false destination domain information (e.g., in the SNI field in a client hello message of an initial set-up of a TLS session). <i>See, e.g., Buruganahalli at claim 9:</i> The system recited in claim 1, wherein the processor is further configured to: intercept a request to establish an encrypted session from the client to the remote server; send a request to establish the encrypted session on behalf of the

Claim	U.S. Patent No. 9,917,856	Disclosure from Buruganahalli
		client to the remote server; send an encrypted session response to the client on behalf of the remote server using a session key associated with the security device; decrypt encrypted traffic between the client and the remote server to monitor for a request from the client to create a tunnel using a first protocol with the remote server; allow the request to create the tunnel; monitor decrypted session traffic between the client and the remote server over the tunnel based on one or more firewall policies; and block the session traffic using the security device if a violation of a first firewall policy is determined, wherein the first firewall policy includes a policy for verifying that the destination domain extracted from a server name indication (SNI) matches a domain associated with a uniform resource locator (URL) for a session associated with the network communications between the client and the remote server.
24.4	identify packets comprising unencrypted data;	<i>See, e.g.,</i> disclosures at Claim 1.2.
24.5	identify packets comprising encrypted data;	<i>See, e.g.,</i> disclosures at Claim 1.3.

Claim	U.S. Patent No. 9,917,856	Disclosure from Buruganahalli
24.6	determine, based on a portion of the unencrypted data corresponding to one or more network-threat indicators of the plurality of network-threat indicators, packets comprising encrypted data that corresponds to the one or more network-threat indicators;	<i>See, e.g.</i>, disclosures at Claim 1.4. <i>See, e.g.</i>, Buruganahalli at claim 9: The system recited in claim 1, wherein the processor is further configured to: intercept a request to establish an encrypted session from the client to the remote server; send a request to establish the encrypted session on behalf of the client to the remote server; send an encrypted session response to the client on behalf of the remote server using a session key associated with the security device; decrypt encrypted traffic between the client and the remote server to monitor for a request from the client to create a tunnel using a first protocol with the remote server; allow the request to create the tunnel; monitor decrypted session traffic between the client and the remote server over the tunnel based on one or more firewall policies; and block the session traffic using the security device if a violation of a first firewall policy is determined, wherein the first firewall policy includes a policy for verifying that the destination domain extracted from a server name indication (SNI) matches a domain associated with a uniform resource locator (URL) for a session associated with the network communications between the client and the

Claim	U.S. Patent No. 9,917,856	Disclosure from Buruganahalli
		remote server.
24.7	filter, based on at least one of a uniform resource identifier (URI) specified by a plurality of packet-filtering rules, data indicating a protocol version specified by the plurality of packet-filtering rules, data indicating a method specified by the plurality of packet-filtering rules, data indicating a request specified by the plurality of packet-filtering rules, or data indicating a command specified by the plurality of packet-filtering rules:	<i>See, e.g.</i>, disclosures at Claim 1.5. <i>See, e.g.</i>, Buruganahalli at claim 9: The system recited in claim 1, wherein the processor is further configured to: intercept a request to establish an encrypted session from the client to the remote server; send a request to establish the encrypted session on behalf of the client to the remote server; send an encrypted session response to the client on behalf of the remote server using a session key associated with the security device; decrypt encrypted traffic between the client and the remote server to monitor for a request from the client to create a tunnel using a first protocol with the remote server; allow the request to create the tunnel; monitor decrypted session traffic between the client and the remote server over the tunnel based on one or more firewall policies; and block the session traffic using the security device if a violation of a first firewall policy is determined, wherein the first firewall policy includes a policy for verifying that the destination domain extracted

Claim	U.S. Patent No. 9,917,856	Disclosure from Buruganahalli
		from a server name indication (SNI) matches a domain associated with a uniform resource locator (URL) for a session associated with the network communications between the client and the remote server.
24.8	packets comprising the portion of the unencrypted data corresponding to one or more network-threat indicators of the plurality of network-threat indicators; and	<i>See, e.g.</i> , disclosures at Claim 1.6.
24.9	the determined packets comprising the encrypted data that corresponds to the one or more network-threat indicators; and	<i>See, e.g.</i> , disclosures at Claim 1.7.
24.10	route, by the packet-filtering system, filtered packets to a proxy system based on a determination that the filtered packets comprise data that corresponds to the one or more network-threat indicators.	<i>See, e.g.</i> , disclosures at Claim 1.8.
25.0	One or more non-transitory computer-readable media comprising instructions that when executed by at least one hardware processor of a packet-filtering system cause the packet-filtering system to:	<i>See, e.g.</i>, disclosures at Claim 1.0. <i>See, e.g.</i>, Buruganahalli at claim 12: A tangible non-transitory computer readable storage medium comprising a computer program for destination domain extraction for secure protocols, the computer program comprising computer instructions for: monitoring network communications between a client and a remote

Claim	U.S. Patent No. 9,917,856	Disclosure from Buruganahalli
		server; determining if the client sends a request to create a secure connection with the remote server, wherein the secure connection utilizes a secure protocol, the secure protocol being a secure sockets layer (SSL) protocol or transport layer security (TLS) protocol; and extracting a destination domain from the request to create the secure connection with the remote server, comprising: before the secure connection is created, extracting the destination domain from a server name indication (SNI) of a client hello message sent from the client to the remote server, the secure connection including an encrypted message, wherein the extracting of the destination domain from the SNI of the client hello message sent from the client to the remote server comprises: identifying the SNI from the client hello message during a handshaking process for setting the secure connection between the client and the remote server; extracting a domain identified in a public certificate sent from the remote server to the client; comparing the destination domain and the domain identified in the public certificate; and in the event that the destination domain matches the domain identified in the public certificate, applying a security policy based on the destination domain to filter traffic at a security device,

Claim	U.S. Patent No. 9,917,856	Disclosure from Buruganahalli
		wherein the security policy includes a whitelist/blacklist policy.
25.1	receive data indicating a plurality of network-threat indicators, wherein at least one of the plurality of network-threat indicators comprise a domain name identified as a network threat;	<i>See, e.g.</i>, disclosures at Claim 1.1. <i>See, e.g.</i>, Buruganahalli at claim 12: A tangible non-transitory computer readable storage medium comprising a computer program for destination domain extraction for secure protocols, the computer program comprising computer instructions for: monitoring network communications between a client and a remote server; determining if the client sends a request to create a secure connection with the remote server, wherein the secure connection utilizes a secure protocol, the secure protocol being a secure sockets layer (SSL) protocol or transport layer security (TLS) protocol; and extracting a destination domain from the request to create the secure connection with the remote server, comprising: before the secure connection is created, extracting the destination domain from a server name indication (SNI) of a client hello message sent from the client to the remote server, the secure connection including an encrypted message, wherein the extracting of the destination domain from the SNI of the client hello

Claim	U.S. Patent No. 9,917,856	Disclosure from Buruganahalli
		message sent from the client to the remote server comprises: identifying the SNI from the client hello message during a handshaking process for setting the secure connection between the client and the remote server; extracting a domain identified in a public certificate sent from the remote server to the client; comparing the destination domain and the domain identified in the public certificate; and in the event that the destination domain matches the domain identified in the public certificate, applying a security policy based on the destination domain to filter traffic at a security device, wherein the security policy includes a whitelist/blacklist policy.
25.2	identify packets comprising unencrypted data;	<i>See, e.g.</i>, disclosures at Claim 1.2. <i>See, e.g.</i>, Buruganahalli at claim 12: A tangible non-transitory computer readable storage medium comprising a computer program for destination domain extraction for secure protocols, the computer program comprising computer instructions for: monitoring network communications between a client and a remote server; determining if the client sends a request to create a secure connection with the remote server, wherein the secure connection

Claim	U.S. Patent No. 9,917,856	Disclosure from Buruganahalli
		utilizes a secure protocol, the secure protocol being a secure sockets layer (SSL) protocol or transport layer security (TLS) protocol; and extracting a destination domain from the request to create the secure connection with the remote server, comprising: before the secure connection is created, extracting the destination domain from a server name indication (SNI) of a client hello message sent from the client to the remote server, the secure connection including an encrypted message, wherein the extracting of the destination domain from the SNI of the client hello message sent from the client to the remote server comprises: identifying the SNI from the client hello message during a handshaking process for setting the secure connection between the client and the remote server; extracting a domain identified in a public certificate sent from the remote server to the client; comparing the destination domain and the domain identified in the public certificate; and in the event that the destination domain matches the domain identified in the public certificate, applying a security policy based on the destination domain to filter traffic at a security device, wherein the security policy includes a whitelist/blacklist policy.
25.3	identify packets comprising encrypted data;	<i>See, e.g., disclosures at Claim 1.3.</i>

Claim	U.S. Patent No. 9,917,856	Disclosure from Buruganahalli
		<i>See, e.g.,</i> Buruganahalli at claim 12: A tangible non-transitory computer readable storage medium comprising a computer program for destination domain extraction for secure protocols, the computer program comprising computer instructions for: monitoring network communications between a client and a remote server; determining if the client sends a request to create a secure connection with the remote server, wherein the secure connection utilizes a secure protocol, the secure protocol being a secure sockets layer (SSL) protocol or transport layer security (TLS) protocol; and extracting a destination domain from the request to create the secure connection with the remote server, comprising: before the secure connection is created, extracting the destination domain from a server name indication (SNI) of a client hello message sent from the client to the remote server, the secure connection including an encrypted message, wherein the extracting of the destination domain from the SNI of the client hello message sent from the client to the remote server comprises: identifying the SNI from the client hello message during a handshaking process for setting the secure connection between the client and the remote server; extracting a domain identified in a public certificate sent from the

Claim	U.S. Patent No. 9,917,856	Disclosure from Buruganahalli
		remote server to the client; comparing the destination domain and the domain identified in the public certificate; and in the event that the destination domain matches the domain identified in the public certificate, applying a security policy based on the destination domain to filter traffic at a security device, wherein the security policy includes a whitelist/blacklist policy.
25.4	determine, based on a portion of the unencrypted data corresponding to one or more network-threat indicators of the plurality of network-threat indicators, packets comprising encrypted data that corresponds to the one or more network-threat indicators;	<i>See, e.g.</i>, disclosures at Claim 1.4. <i>See, e.g.</i>, Buruganahalli at claim 12: A tangible non-transitory computer readable storage medium comprising a computer program for destination domain extraction for secure protocols, the computer program comprising computer instructions for: monitoring network communications between a client and a remote server; determining if the client sends a request to create a secure connection with the remote server, wherein the secure connection utilizes a secure protocol, the secure protocol being a secure sockets layer (SSL) protocol or transport layer security (TLS) protocol; and extracting a destination domain from the request to create the secure connection with the remote server, comprising:

Claim	U.S. Patent No. 9,917,856	Disclosure from Buruganahalli
		before the secure connection is created, extracting the destination domain from a server name indication (SNI) of a client hello message sent from the client to the remote server, the secure connection including an encrypted message, wherein the extracting of the destination domain from the SNI of the client hello message sent from the client to the remote server comprises: identifying the SNI from the client hello message during a handshaking process for setting the secure connection between the client and the remote server; extracting a domain identified in a public certificate sent from the remote server to the client; comparing the destination domain and the domain identified in the public certificate; and in the event that the destination domain matches the domain identified in the public certificate, applying a security policy based on the destination domain to filter traffic at a security device, wherein the security policy includes a whitelist/blacklist policy.
25.5	filter, by the packet-filtering system and based on at least one of a uniform resource identifier (URI) specified by a plurality of packet-filtering rules indicating one or more of the plurality of network-threat indicators, data indicating a protocol version specified by the plurality of packet-filtering rules, data indicating a method specified by the plurality of packet-filtering rules, data indicating a request specified by the	<i>See, e.g.</i>, disclosures at Claim 1.5. <i>See, e.g.</i>, Buruganahalli at claim 12: A tangible non-transitory computer readable storage medium comprising a computer program for destination domain extraction for secure protocols, the computer program comprising computer

Claim	U.S. Patent No. 9,917,856	Disclosure from Buruganahalli
	plurality of packet-filtering rules, or data 25f. indicating a command specified by the plurality of packet-filtering rules:	instructions for: monitoring network communications between a client and a remote server; determining if the client sends a request to create a secure connection with the remote server, wherein the secure connection utilizes a secure protocol, the secure protocol being a secure sockets layer (SSL) protocol or transport layer security (TLS) protocol; and extracting a destination domain from the request to create the secure connection with the remote server, comprising: before the secure connection is created, extracting the destination domain from a server name indication (SNI) of a client hello message sent from the client to the remote server, the secure connection including an encrypted message, wherein the extracting of the destination domain from the SNI of the client hello message sent from the client to the remote server comprises: identifying the SNI from the client hello message during a handshaking process for setting the secure connection between the client and the remote server; extracting a domain identified in a public certificate sent from the remote server to the client; comparing the destination domain and the domain identified in the public certificate; and in the event that the destination domain matches the domain

Claim	U.S. Patent No. 9,917,856	Disclosure from Buruganahalli
		identified in the public certificate, applying a security policy based on the destination domain to filter traffic at a security device, wherein the security policy includes a whitelist/blacklist policy.
25.6	packets comprising the portion of the unencrypted data corresponding to one or more network-threat indicators of the plurality of network-threat indicators; and	<i>See, e.g.</i>, disclosures at Claim 1.6. <i>See, e.g.</i>, Buruganahalli at claim 12: A tangible non-transitory computer readable storage medium comprising a computer program for destination domain extraction for secure protocols, the computer program comprising computer instructions for: monitoring network communications between a client and a remote server; determining if the client sends a request to create a secure connection with the remote server, wherein the secure connection utilizes a secure protocol, the secure protocol being a secure sockets layer (SSL) protocol or transport layer security (TLS) protocol; and extracting a destination domain from the request to create the secure connection with the remote server, comprising: before the secure connection is created, extracting the destination domain from a server name indication (SNI) of a client hello message sent from the client to the remote server, the secure connection including an encrypted message, wherein the extracting of the destination domain from the SNI of the client hello

Claim	U.S. Patent No. 9,917,856	Disclosure from Buruganahalli
		message sent from the client to the remote server comprises: identifying the SNI from the client hello message during a handshaking process for setting the secure connection between the client and the remote server; extracting a domain identified in a public certificate sent from the remote server to the client; comparing the destination domain and the domain identified in the public certificate; and in the event that the destination domain matches the domain identified in the public certificate, applying a security policy based on the destination domain to filter traffic at a security device, wherein the security policy includes a whitelist/blacklist policy.
25.7	the determined packets comprising the encrypted data that corresponds to the one or more network-threat indicators; and	<i>See, e.g.</i>, disclosures at Claim 1.7. <i>See, e.g.</i>, Buruganahalli at claim 12: A tangible non-transitory computer readable storage medium comprising a computer program for destination domain extraction for secure protocols, the computer program comprising computer instructions for: monitoring network communications between a client and a remote server; determining if the client sends a request to create a secure connection with the remote server, wherein the secure connection utilizes a secure protocol, the secure protocol being a secure

Claim	U.S. Patent No. 9,917,856	Disclosure from Buruganahalli
		sockets layer (SSL) protocol or transport layer security (TLS) protocol; and extracting a destination domain from the request to create the secure connection with the remote server, comprising: before the secure connection is created, extracting the destination domain from a server name indication (SNI) of a client hello message sent from the client to the remote server, the secure connection including an encrypted message, wherein the extracting of the destination domain from the SNI of the client hello message sent from the client to the remote server comprises: identifying the SNI from the client hello message during a handshaking process for setting the secure connection between the client and the remote server; extracting a domain identified in a public certificate sent from the remote server to the client; comparing the destination domain and the domain identified in the public certificate; and in the event that the destination domain matches the domain identified in the public certificate, applying a security policy based on the destination domain to filter traffic at a security device, wherein the security policy includes a whitelist/blacklist policy.
25.8	route, by the packet-filtering system, filtered packets to a proxy system based on a determination that the	<i>See, e.g., disclosures at Claim 1.8.</i>

Claim	U.S. Patent No. 9,917,856	Disclosure from Buruganahalli
	filtered packets comprise data that corresponds to the one or more network-threat indicators.	<i>See, e.g.,</i> Burugunahalli at claim 12: A tangible non-transitory computer readable storage medium comprising a computer program for destination domain extraction for secure protocols, the computer program comprising computer instructions for: monitoring network communications between a client and a remote server; determining if the client sends a request to create a secure connection with the remote server, wherein the secure connection utilizes a secure protocol, the secure protocol being a secure sockets layer (SSL) protocol or transport layer security (TLS) protocol; and extracting a destination domain from the request to create the secure connection with the remote server, comprising: before the secure connection is created, extracting the destination domain from a server name indication (SNI) of a client hello message sent from the client to the remote server, the secure connection including an encrypted message, wherein the extracting of the destination domain from the SNI of the client hello message sent from the client to the remote server comprises: identifying the SNI from the client hello message during a handshaking process for setting the secure connection between the client and the remote server; extracting a domain identified in a public certificate sent from the

Claim	U.S. Patent No. 9,917,856	Disclosure from Buruganahalli
		remote server to the client; comparing the destination domain and the domain identified in the public certificate; and in the event that the destination domain matches the domain identified in the public certificate, applying a security policy based on the destination domain to filter traffic at a security device, wherein the security policy includes a whitelist/blacklist policy.