Design Guide

Cisco IOS Firewall

Cisco IOS® Firewall is a stateful security software component of Cisco IOS Software. Firewall integration in Cisco IOS routers augments a router's inherent capabilities: multitopology interfaces, industry-standard routing protocols, and a broad range of services, as well as an expanding group of other security features such as virtual private network (VPN) and intrusion prevention system (IPS) features. Cisco IOS Firewall interoperates with other Cisco IOS Software technologies, including Network Address Translation (NAT), quality of service (QoS), and IP Security (IPsec) and Secure Sockets Layer (SSL) VPN, to become a vital component of an end-to-end network security infrastructure.

Cisco IOS Firewall includes multiple security features:

- Cisco IOS Firewall stateful packet inspection provides true firewall capabilities to protect networks against unauthorized traffic and control legitimate business-critical data.
- Authentication proxy controls access to hosts or networks based on user credentials stored in an authentication, authorization, and accounting (AAA) server.
- Multi-VRF firewall offers firewall services on virtual routers with virtual routing and forwarding (VRF), accommodating overlapping address space to provide multiple isolated private route spaces with a full range of security services.
- Transparent firewall adds stateful inspection without time-consuming, disruptive IP addressing modifications.
- Application inspection controls application activity to provide granular policy enforcement of application usage, protecting legitimate application protocols from rogue applications and malicious activity.

Cisco IOS Firewall is primarily supported in Cisco IOS Software mainline and technology release trains. The service provider release train incorporates limited firewall feature set capabilities, and is not as current as the mainline and technology releases for integration of new features.

This document offers technical discussion of most Cisco IOS Firewall features and provides deployment scenarios in typical network infrastructures, using features supported up to Cisco IOS Software Release 12.4(2)T. The scenarios can act as a guide for the deployment of features that are useful when securing a range of home, small business, branch office, extranet, and enterprise networks.

CISCO IOS FIREWALL BENEFITS

Cisco IOS Firewall has been tested for compliance with several industry certifications, offering third-party validation that Cisco IOS Firewall provides ample firewall protection to meet business security requirements. ICSA Laboratories certified Cisco IOS Firewall under the Modular Firewall Certification Criteria Version 4.1.

Cisco IOS Firewall offers stateful inspection capability on par with competing firewall products, and several benefits when compared to dedicated firewall appliances. A Cisco IOS router with the Firewall Feature set offers additional functions and benefits integrated with the firewall's capabilities:

- **Integrated Routing Capabilities**—Cisco IOS Firewall provides integrated, inline security services. These enhance current Cisco IOS Software capabilities: secure IP routing, multitopology interfaces, industry-standard routing protocols, NAT, and voice and video services.
- **Industry-Leading VPN**—Cisco IOS Software offers secure VPN capabilities to address almost any secure network requirement. EasyVPN, DMVPN, traditional IPsec site-to-site, and Web-based SSL VPN support capabilities to securely connect remote-access users and remote sites over the public Internet, or to offer added security to existing private-network connections.

- **Full-Featured Firewall**—Stateful Packet Inspection provides stateful security and control for both common and user-defined network services, and configurable protection from denial of service (DoS) attacks.
- **Authentication Proxy**—The Authentication Proxy (Auth Proxy) offers per-user authenticated access control to network resources. Users' authorization policy may be provided to the Auth Proxy device by an AAA server.
- **Scalability**—Available on a wide variety of Cisco IOS platforms, Cisco IOS Firewall scales to meet any network's bandwidth and performance requirements.
- **Security in the Infrastructure**—Enables sophisticated security and policy enforcement for connections within an organization (intranet), from central sites to remote offices and telecommuters in the home, between an organization and its partner networks, and between the organization and the Internet.
- **Integrates with Existing Investment in Cisco**—IT managers can enhance security without additional cost and complexity of adding standalone security appliances.

TECHNICAL HIGHLIGHTS

Cisco IOS Firewall consists of several major subsystems:

- Stateful Packet Inspection provides a granular firewall engine
- Authentication Proxy offers a per-host access control mechanism
- Application Inspection features add protocol conformance checking and network use policy control

Enhancements to these features extend these capabilities to VRF instances to support multiple virtual routers per device, and to Cisco Integrated Route-Bridging features to allow greater deployment flexibility, reduce implementation timelines, and ease requirements to add security to existing networks.

This portion of the document introduces the individual subsystems and discusses their benefits.

Cisco IOS Software Stateful Packet Inspection

Stateful Packet Inspection (SPI) is at the heart of Cisco IOS Firewall, providing a per-application control mechanism across network perimeters, as well as within networks through the Transparent Firewall capability. Stateful Packet Inspection was known as Context-Based Access Control (CBAC) in early versions of Cisco IOS Firewall, but the name was changed as the feature set was enhanced and augmented far beyond the original CBAC capability. SPI enhances security for TCP and UDP applications by scrutinizing several attributes of data connection. The inspection engine tracks the state and context of network connections to secure traffic flow. SPI provides support for several complex, advanced services such as streaming protocols, IP voice, and other complex services that require detailed scrutiny to support additional data and media channels.

Protection Against Attack

Cisco IOS Firewall SPI provides DoS detection and prevention against some popular attack modes, such as SYN (synchronize/start) flooding, port scans, and packet injection. When the router detects unusually high rates of new connections, it issues an alert message, and resets excessive half-open TCP connections to prevent system resource depletion. Cisco IOS Firewall tracks connections by destination address and port pairs to control undesired activity and reduce impact on hosts on the protected network that are under attack from malicious activity originating outside the firewall.

Cisco IOS Firewall SPI protects against packet-injection attacks by checking several components of TCP and UDP sessions. Source and destination IP address and port numbers must match, as well as TCP sequence number. Other attributes are checked as well, such as TCP window size, reducing the likelihood of buffer overrun attacks.

Alerts and Audit Trails

Cisco routers generate real-time alerts and audit trails based on inputs from the SPI engine. Enhanced audit trail features use syslog to track all network transactions for advanced, session-based reporting—recording time stamps, source host, destination host, ports used, and the total number of transmitted bytes.

Real-time alerts send syslog error messages to central management consoles upon detecting suspicious activity. Using the firewall engine inspection rules, alerts and audit trail information can be configured on a per-application protocol basis. These configurable real-time alerts, audit trail, and logging events allow administrators to track potential security breaches and other nonstandard activities in real time.

Authentication Proxy

Network administrators can create specific security policies for each user with the Cisco IOS Firewall per-user authentication and authorization. Previously, user identity and related authorized access was determined by a user's fixed IP address, or a single security policy had to be applied to an entire user group or subnet. Now, per-user policy can be downloaded dynamically to the router from a TACACS+ or RADIUS authentication server.

Users log into the network resources or onto the Internet via HTTP, HTTPS, FTP, and Telnet authentication interfaces, and their specific access profiles are downloaded to Cisco IOS Firewall routers with Authentication Proxy upon successful authentication. Authentication and authorization can be applied for inbound and/or outbound traffic, which means that auth proxy can support Internet, intranet, and extranet configurations.

Synergy with NAT and Port-to-Application Mapping (PAM)

The combination of Cisco IOS Firewall and NAT enable the firewall to perform stateful inspection, while hiding the internal IP addresses from the outside world and minimizing public Internet address space requirements. Flexible port-application mapping (PAM) supports applications running on nonstandard ports, customizing access control for specific applications and services to meet the requirements of the network.

Application Inspection

Cisco IOS Firewall offers deeper, more detailed inspection of certain application protocols to prevent abuse and malicious activity that may be transmitted over service ports generally used for more desirable traffic, such as HTTP, Simple Mail Transfer Protocol (SMTP)/Extended SMTP (ESMTP), Post Office Protocol (POP), and Internet Mail Access Protocol (IMAP). Application Inspection capabilities vary by service, from checking authentication to ensure login credentials are encrypted, to performing granular control over types and quantities of content that may be carried over a controlled service.

CISCO IOS FIREWALL TECHNICAL DISCUSSION AND APPLICATION EXAMPLES

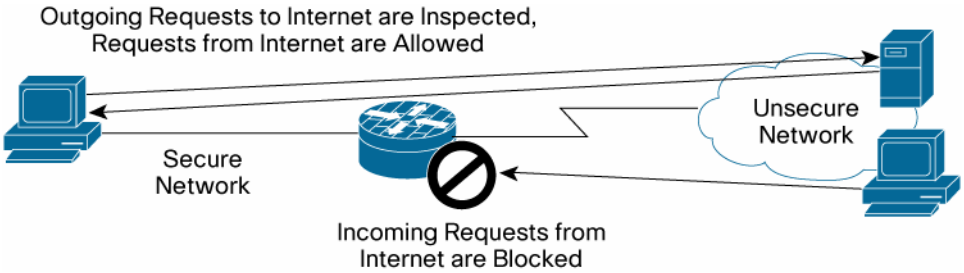
Stateful Packet Inspection

Stateful Packet Inspection (SPI), the foundation of Cisco IOS Firewall, was introduced in Cisco IOS Software Release 11.2P, and found its way into an early release of 12.0. SPI has been enhanced several times to improve performance, capability, and flexibility. This document focuses on the most recent implementation of Cisco IOS SPI, but offers some evolutionary background where appropriate, specifically when relevant to improvements in function, monitoring, and configuration.

SPI was introduced as a feature called Context-Based Access Control (CBAC). Prior to CBAC, Cisco IOS Software's only packet-filtering mechanism was the access control list (ACL). CBAC greatly enhanced the packet filtering capability of ACLs by introducing stateful filtering capability. The early Cisco IOS Firewall capability was occasionally perceived as a "glorified" ACL. This misconception is partly due to the fact that ACL monitoring commands were used to monitor CBAC activity, as well as the fact that inspection used (and still uses) ACLs to filter traffic, permitting desired traffic, while blocking unwanted, potentially harmful traffic. However, CBAC substantially augments an ACL's capability for restricting traffic. CBAC monitors several attributes in TCP connections, UDP sessions, and Internet Control Message Protocol (ICMP) dialogue to ensure that the only traffic allowed through a firewall ACL is the return traffic for dialogue that was originated on the private side of the firewall.

Cisco IOS SPI can be explained most simply as being a mechanism to discover “good” connections that originate on the secure side of the firewall, and watch for and allow the return traffic that correlates with these connections. Connections originating on the unsecure side of the firewall are not allowed to reach the secure network, as controlled by an ACL facing the unsecure network (Figure 1).

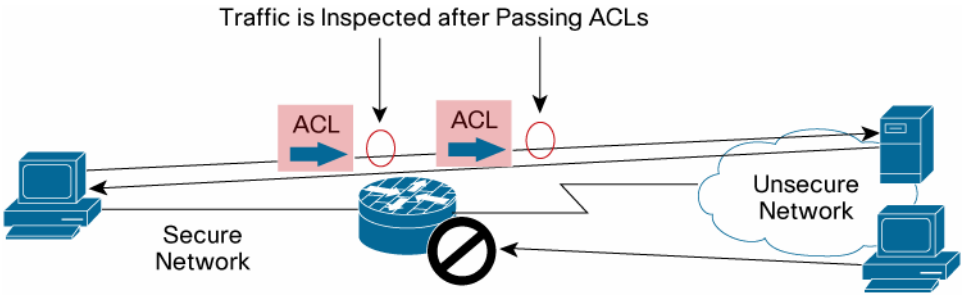
Figure 1. Cisco IOS Firewall Stateful Inspection



Many changes have been made to CBAC to enhance its capability and increase performance. Inspection of some protocols has been enhanced to ensure protocol compliance or offer application-level service filtering. Cisco IOS Software Release 12.3(4)T’s ACL Bypass feature introduced substantial improvements in performance and significant changes to the stateful inspection architecture. CBAC had outgrown its original, basic function and was renamed Cisco IOS Stateful Packet Inspection to more accurately reflect the feature’s capability. CBAC is frequently still used synonymously with Stateful Packet Inspection, but the CBAC name does not reflect the full feature set offered by Cisco IOS SPI.

SPI inspects the packet after it passes the inbound ACL of an input interface if **ip inspect in** is applied, or after the outbound ACL of output interface if **ip inspect out** is used. Thus, outbound traffic must be permitted by input ACLs facing the source, and outbound ACLs facing the destination.

Figure 2. Access List Action on Traffic



SPI monitors connections from a secured network to an unsecured network, and anticipates the traffic returning to the secure host from the unsecure network. The mechanism for anticipating and allowing the return traffic changed slightly as Cisco IOS Firewall changed from CBAC to SPI. Prior to Cisco IOS Software Release 12.3(4)T, CBAC placed dynamic access control entries (ACEs) in ACLs in the return path for internally originated connections, as indicated in “show access-list” for a simple “deny all” ACL:

```
sdp-ezvpn#show access-lists
Extended IP access list 111
  permit tcp host 172.16.105.1 eq telnet host 172.16.105.10 eq
  1176 (30 matches)
  10 deny ip any any (708 matches)
```

With the introduction of Cisco IOS Software Release 12.3(4)T, the ACL Bypass feature modified SPI's infrastructure so dynamic ACEs are no longer used. Instead, SPI maintains a session table listing all of the firewall's active sessions. The contents of the session table can be viewed with the "show ip inspect sessions" command:

```
yourname#sh ip insp sessions
Established Sessions
Session 63D9A9E0 (192.168.110.10:1038)=>(172.16.110.1:23) telnet
SIS_OPEN
```

ACL Bypass improves firewall performance for two reasons. SPI is able to maintain a more efficient list to track active sessions, reducing the time required for session setup and verification. Also, return traffic is not subjected to ACLs on the return path, so when return traffic finds a matching entry in the session table, it is shunted past the ACLs in the packet path, reducing the CPU overhead the packet incurs as it moves through the router's processing.

Stateful Packet Inspection Memory Consumption

The router allocates a small amount of memory for every session it must track. Basic SPI functions require roughly 700 bytes of the router's memory to record source address and port number, destination address and port number, and protocol. If the router is configured for multi-VRF capability, the source and destination VRF for each connection are maintained in the session information as well. Firewall inspection using additional, deeper application-layer inspection requires more memory per session. The router allocates a 500-session block of memory when SPI is configured. As more sessions are tracked by the firewall, the router allocates additional 500-session blocks from the router's unused memory. When the router has no remaining free memory, new session allocation requests will fail.

Stateful Inspection Design and Configuration Tasks

Deployment scenarios toward the end of this document offer several examples for SPI configuration, but two basic discussions of firewall configurations are offered here. One configuration is the least complex, offering the easiest configuration, requiring little knowledge of network usage patterns, but offering little network use control. The other configuration is the preferred application of Cisco IOS SPI, offering better network policy control and tighter security, but requiring a better grasp of network protocol usage.

Least Complex Cisco IOS Firewall

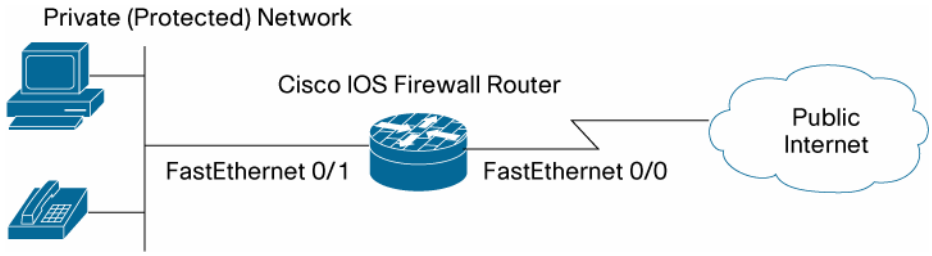
The least complex SPI configuration uses a "deny any" ACL facing the unsecure network, and offers limited capability to restrict network usage to a limited application list. This prevents hosts on the unsecure network from sending traffic to the secure network, but blocks return traffic on legitimate, internally originated connections. To facilitate the return of legitimate traffic, you will need to configure a simple inspection set and apply it to inbound traffic on the internal interface, or to the outbound traffic on the external interface.

Least complex Cisco IOS SPI configuration tasks:

- Configure ACLs to *block* traffic from the unsecure network.
- Be sure ACLs *permit* legitimate traffic connections from the secure network to the unsecure network.
- Create inspection rules. Apply the rules inbound to the secure-side interface or outbound to the unsecure-side interface.
- Verify firewall function.

Consider this example:

Figure 3. Simple Network Diagram



Configure and apply the “deny any” ACL on the public-facing interface, fastethernet 0/0, to block requests from the unsecure network.

```
access-list 101 deny ip any any
interface fastethernet 0/0
ip access-group 101 in
```

Configure a basic inspection policy. Most Internet traffic can be inspected by “inspect tcp”, “inspect udp”, and “inspect icmp”. This permits the most common Internet traffic, including Web browsing, e-mail applications, file transfer, remote-console and remote-desktop applications, instant messaging, and peer-to-peer file transfer applications. Certain applications that use a secondary data channel, such as voice applications or streaming media applications, may require that you configure the protocol-specific inspection for that particular service, such as “inspect ftp”, “inspect skinny”, or “inspect h.323”. If you set up Cisco IOS Stateful Inspection and find that one of your network applications that must traverse the firewall stops working, you should consult the product’s documentation or knowledgebase and determine if the software vendor offers documentation specific to setting up a Cisco IOS Firewall.

Define the inspection set:

```
ip inspect name myfw tcp
ip inspect name myfw udp
ip inspect name myfw icmp
```

Apply the inspection inbound to the inside interface:

```
interface fastethernet 0/1
ip inspect myfw in
```

Or, apply inspection outbound to the outside interface:

```
interface fastethernet 0/0
ip inspect myfw out
```

This completes the configuration of the least complex Cisco IOS SPI.

Inspecting Complex Services

Several Internet services use multiple channels to handle the service control and data communications. For instance, FTP uses one channel to open initial communications from the client to the server, and the server opens a separate channel back to the client to send the actual file transfer traffic. Similarly, H.323 uses one channel for initial call setup, and other channels are negotiated from the initial connection to carry the actual streaming media, such as audio traffic in an IP telephony connection.

The Stateful Inspection engine only needs to see the initiating connection for these complex services. Subsequent connections for the session are dynamically opened for the session, based on SPI's scrutiny of the connection setup. This is usually known as "fixup". If an outbound ACL is configured on an interface to restrict network access policy, it must only account for the initiating port. The task of accommodating the media channels will be handled by Stateful iInspection's fixup. The following command lists the complex services and their initiating ports that Cisco IOS Stateful Inspection can handle:

```
FWRouter# sh ip port-map

Default mapping: vdolive     port 7000   system defined
Default mapping: sunrpc      port 111    system defined
Default mapping: netshow     port 1755   system defined
Default mapping: cuseeme     port 7648   system defined
Default mapping: rtsp        port 8554   system defined
Default mapping: realmedia   port 7070   system defined
Default mapping: streamworks port 1558   system defined
Default mapping: ftp         port 21     system defined
Default mapping: rtsp        port 554    system defined
Default mapping: h323        port 1720   system defined
Default mapping: sip         port 5060   system defined
Default mapping: mgcp        port 2427   system defined
```

Granular Inspection

Granular Protocol Inspection (GPI), introduced in Cisco IOS Software Release 12.3(14)T, offered complete integration with PAM. Prior to GPI, a firewall policy was defined by configuring inspection for outbound TCP, UDP, and ICMP traffic. Inspection was explicitly configured for specific protocols, such as FTP, H.323, Skinny, Session Initiation Protocol (SIP) and others that required fixup to watch for and allow protocol-specific media channels. Common single-connection services such as POP, Telnet, Microsoft RPC, and other simple protocols were inspected by the generic capability of TCP, UDP, and ICMP inspection. Using these generic inspection capabilities is simple to configure, but it limits Stateful Packet Inspection's granularity—any traffic that was allowed to leave through a firewall was allowed to return because inspection created an ACL Bypass entry for that traffic.

GPI allows creation of specific ACL Bypass for only the desired traffic, as defined by an inspection list consisting of only the protocols that are explicitly permitted by an organization's Internet/security access policy.

A complete list of the default services that GPI can inspect is contained in the appendix at the enof the Stateful Inspection section.

More Secure Cisco IOS Firewall

A more secure firewall bears a slight resemblance to the least complex firewall, at least from the standpoint of the unsecured public network's access to the secured network. However, the stateful inspection policy is much more focused on the specific services that will be allowed through the firewall. This means that you must have a better understanding of the network's requirement for service use. Your organization should have a documented list of services that are appropriate for use on the network. If you maintain a list of acceptable network services, you can follow a clear course of action should it become necessary to address employee violation of policy or abuse of network resources.

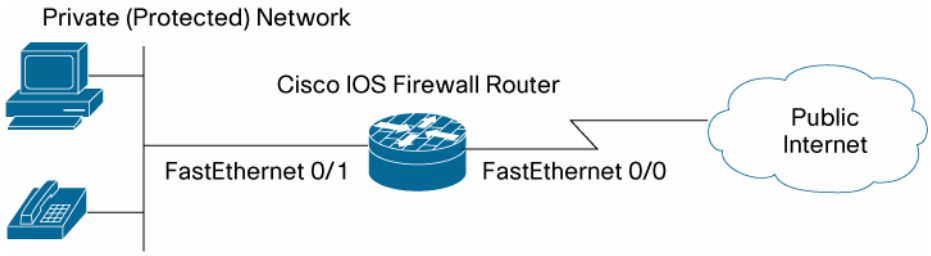
GPI allows the user to specify PAM-defined services for firewall permission. Prior to GPI, if you wished to restrict your firewall policy, you simply used "inspect tcp/udp/icmp" as in the least complex firewall example, but you placed an ACL in the outbound packet path to block access to specific services, or to restrict the list to a specific few. If you use GPI, you must explicitly state the list of protocols, but you will be able to use the user-configurable PAM names for the allowed services, instead of using specific port numbers or ACL service names, which cannot be modified to reflect network requirements.

More secure Cisco IOS Stateful Packet Inspection configuration tasks:

- Identify traffic that will be allowed out through the firewall.
- Configure ACLs to *block* traffic from the unsecure network.
- Be sure ACLs *permit* legitimate traffic from the secure network to the unsecure network.
- Create inspection rules. Apply the rules inbound to the secure-side interface or outbound to the unsecure-side interface.
- Verify firewall function.

Consider this example:

Figure 4. Simple Network Diagram



Network policy in this example allows users to access these Internet services:

- Web and secure Web (HTTP/HTTPS)
- Mail (POP3, IMAP, SMTP)
- Secure terminal (SSH)
- Internet name resolution (DNS)
- File transfer (FTP)

Apply the "deny any" ACL on the public-facing interface to block requests from the unsecure network:

```
access-list 101 deny ip any any
interface fastethernet 0/0
ip access-group 101 in
```

Configure the inspection policy for the protocols listed in the network use policy. Every protocol that will be allowed through the firewall must be specifically named. If you wish to allow additional services, add the services to the inspection policy. If you set up Cisco IOS Stateful Inspection and find that one of your network applications that must traverse the firewall stops working, you should consult the product's documentation or knowledgebase and determine if the software vendor offers documentation specific to setting up a Cisco IOS Firewall.

Define the inspection set:

```
ip inspect name myfw http
ip inspect name myfw https
ip inspect name myfw pop3
ip inspect name myfw esmtp
ip inspect name myfw imap
ip inspect name myfw ssh
ip inspect name myfw dns
ip inspect name myfw ftp
ip inspect name myfw icmp
```

"Inspect http" adds capability to inspect returned content for java applets, offering the option to block potentially malicious java content. However, java filtering incurs a substantial performance penalty. To configure an http inspection policy that does not inspect for embedded java content, define an ACL exempting network address ranges from java inspection and associate the ACL with "inspect http":

```
access-list 102 permit ip any any
ip inspect name myfw http java-list 102
```

Apply the inspection inbound to the inside interface:

```
interface fastethernet 0/1
ip inspect myfw in
```

Or, apply inspection outbound to the outside interface:

```
interface fastethernet 0/0
ip inspect myfw out
```

This completes the configuration of the most secure Cisco IOS Stateful Packet Inspection. Cisco.com's reference for Granular Protocol Inspection is available at: http://www.cisco.com/en/US/products/sw/iosswrel/ps5207/products_feature_guide09186a008040afd7.html

Note: In general, network administrators should account for their specific local topology (which interfaces are considered protected and unprotected). Additional time should be allowed for testing and access considerations for ongoing management of the routers before configuring the firewall. To prevent traffic via the firewall, it is critical to understand how extended access lists function.

Denial of Service

Cisco IOS Stateful Packet Inspection maintains counters of the number of "half-open" TCP connections, as well as the total connection rate through the firewall and IPS software. These half-open connections are TCP connections that have not completed the SYN—SYN/ACK—ACK handshake that is always used by TCP peers to negotiate the parameters of their mutual connection. Cisco IOS Firewall also regards UDP sessions with traffic in only one direction as "half-open", as nearly all applications that use UDP for transport will acknowledge reception of data. UDP sessions without

acknowledgement are likely indicative of DoS activity, or attempts to connect between two hosts where one of the hosts has become unresponsive. Some malicious individuals write worms or viruses that infect multiple hosts on the Internet, then attempt to overwhelm specific Internet servers with a SYN attack, in which large numbers of SYN connections are sent to a server by multiple hosts on the public Internet or within an organization's private network. SYN attacks represent a hazard to Internet servers, as servers' connection table can be loaded with "bogus" SYN connection attempts that arrive faster than the server can deal with the new connections. This is called a "Denial-of-Service" attack, as the large number of connections in the victim server's TCP connection list prevents legitimate users from gaining access to the victim Internet servers.

Cisco IOS Stateful Packet Inspection provides protection from DoS attack as a *default* when an inspection rule is applied. The DoS protection is enabled on the interface, in the direction in which the firewall is applied, for the protocols that the firewall policy is configured to inspect. DoS protection is only enabled on network traffic if the traffic enters or leaves an interface with inspection applied in the same direction of the traffic's initial movement. Cisco IOS Firewall inspection provides several adjustable values to protect against DoS attacks. These settings have default values that may interfere with proper network operation if they are not configured for the appropriate level of network activity in networks where connection rates will exceed the defaults:

- `ip inspect max-incomplete high` *`value`* `(default 500)`
- `ip inspect max-incomplete low` *`value`* `(default 400)`
- `ip inspect one-minute high` *`value`* `(default 500)`
- `ip inspect one-minute low` *`value`* `(default 400)`
- `ip inspect tcp max-incomplete host` *`value`* `(default 50) [block-time` *`minutes`* `(default 0)]`

These parameters allow you to configure the points at which your firewall router's DoS protection begins to take effect. When your router's DoS counters exceed the default or configured values, the router will reset one old half-open connection for every new connection that exceeds the configured *max-incomplete* or *one-minute high* values, until the number of half-open sessions drops below the *max-incomplete low* values. The router will send a syslog message if logging is enabled, and if Intrusion Protection System (IPS) is configured on the router, the firewall router will send a DoS signature message via SDEE. If the DoS parameters are not adjusted to your network's normal behavior, normal network activity may trigger the DoS protection mechanism, causing application failures, poor network performance, and high CPU utilization on the Cisco IOS Firewall router.

While you cannot "disable" your firewall's DoS protection, you can adjust the DoS protection so that it will not take effect unless a very large number of half-open connections are present in your firewall router's Stateful Inspection session table.

Follow this procedure to tune your firewall's DoS Protection to your network's activity:

Step 1. Be sure your network is not infected with viruses or worms that could lead to erroneously large half-open connection values and attempted connection rates. If your network is not a "clean slate", there is no way to properly adjust your firewall's DoS protection.

Step 2. Set the max-incomplete high values to very high values:

```
ip inspect max-incomplete high 20000000
ip inspect one-minute high 100000000
ip inspect tcp max-incomplete host 100000 block-time 0
```

This will prevent the router from providing DoS protection for the time being while you observe your network's connection patterns. If you wish to leave DoS protection disabled, stop following this procedure now.

Step 3. Clear the IOS Firewall statistics, using the following command:

```
show ip inspect statistics reset
```

Step 4. Leave the router configured in this state for some time, perhaps as long as 24-48 hours, so you can observe the network's pattern over a full day's activity cycle. **While the values are adjusted to very high levels, your network will not benefit from Cisco IOS Firewall or IPS DoS protection.**

Step 5. After waiting for some observation period, check the DoS counters with the following command. The parameters you must observe to tune your DoS protection are highlighted in bold:

```
router#show ip inspect statistics
Packet inspection statistics [process switch:fast switch]
  tcp packets: [528:22519]
  udp packets: [318:0]
Interfaces configured for inspection 1
Session creations since subsystem startup or last reset 766
Current session counts (estab/half-open/terminating) [1:0:0]
Maxever session counts (estab/half-open/terminating) [48:12:5]
Last session created 00:12:21
Last statistic reset never
Last session creation rate 0
Last half-open session total 0
```

Step 6. Configure "ip inspect max-incomplete high" to a value twenty-five percent higher than your router's indicated *maxever session count half-open* value.

For example:

```
Maxever session counts (estab/half-open/terminating) [48:12:5]
```

65 * 1.25 = 81.25, thus, configure:

```
router(config)#ip inspect max-incomplete high 818
```

Step 7. Configure "ip inspect max-incomplete low" to the value your router displayed for its *maxever session count half-open* value.

For example:

```
Maxever session counts (estab/half-open/terminating) [48:12:5]
```

Thus, configure:

```
router(config)#ip inspect max-incomplete low 65
```

Step 8. The counter for "ip inspect one-minute high" and "one-minute low" maintains a sum of all TCP, UDP, and ICMP connection attempts during the preceding minute of the router's operation, whether the connections have been successful or not. A rising connection rate could be indicative of a worm infection on a private network, or an attempted DoS against a server. IOS does not maintain a value of the maxever one-minute connection rate, so you must calculate the value you will apply based on observed maxever values. While the maximum indicated values for established, half-open, and terminating sessions are unlikely to occur in the same instant, the calculated values used for the one-minute settings have been observed to be reasonably accurate. To calculate the *ip inspect one-minute low* value, add the indicated established, half-open, and terminating values, then multiply the sum by three.

For example:

```
Maxever session counts (estab/half-open/terminating) [48:12:5]
```

(48+12+5) * 3 = 195, thus, configure:

```
ip inspect one-minute low 195
```

Step 9. Calculate and configure "ip inspect max-incomplete high." The *ip inspect one-minute high* value should be twenty-five percent greater than the calculated *one-minute low* value.

For example:

ip inspect one-minute low (195) * 1.25 = 244, thus, configure:

```
ip inspect one-minute high 244
```

Step 10. You will need to define a value for "ip inspect tcp max-incomplete host" according to your understanding of your servers' capability.

Step 11. Monitor your network's DoS protection activity. Ideally, you should use a syslog server and record occurrences of DoS attack detection. If detection happens very frequently, you may need to monitor and adjust your DoS protection parameters.

For more information about TCP SYN DoS attacks, please visit:
http://www.cisco.com/en/US/tech/tk828/technologies_tech_note09186a00800f67d5.shtml

Logging and Audit-Trail

Real-time alerts send syslog error messages to central management consoles upon the detection of suspicious activity. Enhanced audit trail features use syslog to track all transactions and to record time stamps, source host , destination host, ports used, session duration, and the total number of transmitted bytes for advanced, session-based reporting.

To enable logging and send messages to a syslog server:

```
FWRouter(config)# logging on
FWRouter(config)# logging 192.168.1.11
```

To enable audit-trail of firewall messages:

```
FWRouter(config)# ip inspect audit-trail
```

Audit-trail can be enabled or disabled per protocol in the firewall rules to control the amount of audit-trail messages.

Packet Path for Cisco IOS Firewall Inspection

Understanding the inspection process can be important when configuring Cisco IOS Firewall. When an outbound packet arrives at an interface, it will be processed sequentially:

- The inbound ACL of the input interface is applied
- The NAT inbound is applied
- The NAT outbound is applied
- The outbound ACL of the output interface is applied
- Advanced firewall inspection processing occurs

- The IP packet goes through the output interface

Cisco IOS Firewall inspects packets after input and output ACL checks. When inspecting, the advanced firewall engine may insert or remove the ACL items associated with a session, depending upon its state and context. The following explains the process that the packet undertakes for many of the components within the router.

Troubleshooting CBAC: http://www.cisco.com/en/US/products/sw/secursw/ps1018/products_tech_note09186a0080094112.shtml

Stateful Inspection Enhancements

Some protocol inspection, such as inspection for HTTP, SMTP, ESMTP, and Sun RPC, offers additional, deeper scrutiny into application activity to ensure that malicious or unauthorized activity is not occurring.

Appendix 1: Complete List of Granular Protocol Inspection-Supported Services

802-11-iapp	IEEE 802.11 WLANs WG IAPP
ace-svr	ACE server/propagation
aol	America Online
appfw	Application firewall
appleqtc	Apple QuickTime
bgp	Border Gateway Protocol (BGP)
bliff	Bliff mail notification
bootpc	Bootstrap Protocol Client
bootps	Bootstrap Protocol Server
cddbp	CD Database Protocol
cifs	Common Internet file system (CIFS)
cisco-fna	Cisco FNATIVE
cisco-net-mgmt	cisco-net-mgmt
cisco-svcs	Cisco license/perf/GDP/X.25/ident svcs
cisco-sys	Cisco SYSMAINT
cisco-tdp	Cisco Tag Distribution Protocol (TDP)
cisco-tna	Cisco TNATIVE
citrix	Citrix IMA/ADMIN/RTMP
citriximaclient	Citrix IMA client
clp	Cisco Line Protocol
creativepartnr	Creative Partner
creativeserver	Creative Server
cuseeme	CUSeeMe Protocol
daytime	Daytime (RFC 867)
dbase	dBASE UNIX
dbcontrol_agent	Oracle dbControl Agent po
ddns-v3	Dynamic DNS Version 3

dhcp-failover	Dynamic Host Control Protocol (DHCP) failover
discard	Discard port
dns	Domain Name System (DNS)
dnsix	DNSIX Securit Attribute Token Map
echo	Echo port
entrust-svc-handler	Entrust KM/Administration Service Handler
entrust-svcs	Entrust sps/aaas/aams
esmtp	Extended SMTP
exec	Remote process execution
fcip-port	FCIP
finger	Finger
fragment	IP fragment inspection
ftp	File Transfer Protocol (FTP)
ftps	FTP over Transport Layer Security/Secure Sockets Layer (TLS/SSL)
gdoi	Group Domain of Interpretation (GDOI) Protocol
giop	Oracle GIOP/SSL
gopher	Gopher
gtpv0	General Packet Radio Service (GPRS) Tunneling Protocol Version 0
gtpv1	GPRS Tunneling Protocol Version 1
h323	H.323 Protocol (Microsoft NetMeeting, Intel Video Phone)
h323callsigalt	H.323 Call Signal Alternate
h323gatestat	H.323 Gatestat
hp-alarm-mgr	HP Performance data alarm manager
hp-collector	HP Performance data collector
hp-managed-node	HP Performance data managed node
hsrp	Hot Standby Router Protocol (HSRP)
http	HTTP
https	Secure HTTP
ica	ica (Citrix)
icabrowser	icabrowser (Citrix)
icmp	Internet Control Message Protocol (ICMP)
ident	Authentication Service
igmpv3lite	Internet Group Management Protocol (IGMP) over UDP for SSM
imap	IMAP
imap3	Interactive Mail Access Protocol 3
imaps	IMAP over TLS/SSL
ipass	IPASS

ipsec-msft	Microsoft IP Security (IPSec) NAT-T
ipx	IPX
irc	Internet Relay Chat Protocol
irc-serv	IRC-SERV
ircs	IRC over TLS/SSL
ircu	IRCU
isakmp	ISAKMP
iscsi	iSCSI
iscsi-target	iSCSI port
kazaa	KAZAA
kerberos	Kerberos
kermit	kermit
l2tp	Layer 2 Tunneling Protocol (L2TP)/Layer 2 Forwarding (L2F)
ldap	Lightweight Directory Access Protocol (LDAP)
ldap-admin	LDAP admin server port
ldaps	LDAP over TLS/SSL
login	Remote login
lotusmtap	Lotus Mail Tracking Agent Protocol
lotusnote	Lotus Notes
microsoft-ds	Microsoft-DS
ms-cluster-net	Microsoft Cluster Net
ms-dotnetster	Microsoft .NETster Port
ms-sna	Microsoft SNA Server/Base
ms-sql	Microsoft SQL
ms-sql-m	Microsoft SQL Monitor
msexch-routing	Microsoft Exchange Routing
mysql	MySQL
n2h2server	N2H2 Filter Service Port
ncp-tcp	NCP (Novell)
net8-cman	Oracle Net8 Cman/Admin
netbios-dgm	NETBIOS Datagram Service
netbios-ns	NETBIOS Name Service
netbios-ssn	NETBIOS Session Service
netshow	Microsoft NetShow Protocol
netstat	Variant of systat
nfs	Network File System (NFS)
nntp	Network News Transport Protocol (NNTP)

ntp	Network Time Protocol (NTP)
oem-agent	OEM Agent (Oracle)
oracle	Oracle
oracle-em-vp	Oracle EM/VP
oraclenames	Oracle Names
orasrv	Oracle SQL*Net v1/v2
parameter	Specify inspection parameters
pcanywheredata	pcANYWHEREdata
pcanywherestat	pcANYWHEREstat
pop3	POP3
pop3s	POP3 over TLS/SSL
pptp	Point-to-Point Tunneling Protocol (PPTP)
pwdgen	Password Generator Protocol
qmtp-tcp	Quick Mail Transfer Protocol
r-winsock	remote-winsock
radius	RADIUS and accounting
rcmd	R commands (r-exec, r-login, r-sh)
rdb-dbs-disp	Oracle RDB
realaudio	Real Audio Protocol
realmedia	RealNetwork's Realmedia Protocol
realsecure	ISS Real Secure Console Service Port
router	Local Routing Process
rpc	Remote Procedure Call (RPC) Protocol
rsvd-tcp	RSVD
rsvp-encap	RSVP ENCAPSULATION-1/2
rsvp_tunnel	RSVP Tunnel
rtc-pm-port	Oracle RTC-PM port
rtelnet	Remote Telnet service
rtsp	Real-Time Streaming Protocol (RTSP)
send-tcp	SEND
shell	Remote command
sip	Session Initiation Protocol (SIP)
sip-tls	SIP-TLS
skinny	Skinny Client Control Protocol (SCCP)
sms	SMS RCINFO/XFER/CHAT
smtp	Simple Mail Transfer Protocol (SMTP)
snmp	Simple Network Management Protocol (SNMP)

snmptrap	SNMP Trap
socks	Socks
sql-net	SQL-NET
sqlnet	SQL Net Protocol
sqlserv	SQL Services
sqlsrv	SQL Service
ssh	SSH Remote Login Protocol
sshell	SSLshell
ssp	State Sync Protocol
streamworks	StreamWorks Protocol
stun	cisco STUN
sunrpc	SUN Remote Procedure Call
syslog	Syslog service
syslog-conn	Reliable Syslog service
tacacs	Login Host Protocol (TACACS)
tacacs-ds	TACACS -Database Service
tarantella	Tarantella
tcp	Transmission Control Protocol (TCP)
telnet	Telnet
telnets	Telnet over TLS/SSL
tftp	Trivial File Transfer Protocol (TFTP)
time	Time
timed	Time server
tr-rsrb	Cisco RSRB
ttc	Oracle TTC/SSL
udp	User Datagram Protocol (UDP)
uucp	UUCPD/UUCP-RLOGIN
vdolive	VDOLive Protocol
vqp	VQP
webster	Network dictionary
who	Whois service
wins	Microsoft WINS
x11	X Window System
xdmcp	XDM Control Protocol

Configuring Cisco IOS Transparent Firewall

Many networks have IP address flexibility limitations or may require a firewall to temporarily augment security or assist in diagnosing a network security issue. In these circumstances, a simple “drop-in” firewall can be placed between two physical network segments to protect hosts in one segment from the hosts in the other segment, while maintaining existing IP addressing. The drop-in firewall requires no network addressing changes and offers a short implementation period. Cisco IOS Transparent Firewall answers this need by integrating Layer 2 bridging with Cisco IOS Firewall packet inspection.

This section describes two applications for Cisco IOS Transparent Firewall.

Background

Most firewall applications require TCP/IP Layer 3 routing, where the network that a firewall protects must be in a completely separate subnet from the potentially hostile network. The firewall must forward the packets from one subnet to the other, inspecting the traffic for firewall policy compliance during the routing operation. This is impractical for some applications, where networking or operational requirements do not allow sufficient address space or downtime flexibility to reconfigure a network to accommodate a traditional “routing” firewall.

Cisco IOS Transparent Firewall uses Layer 2 bridging to apply Cisco IOS Firewall capabilities within a single IP subnet. Cisco IOS Firewall inspection is applied as the router bridges the traffic from one segment to the other, according to the policy. Cisco IOS Transparent Firewall only inspects the traffic moving between the segments of the bridge group. Traffic to other subnets requires inspection as it traverses Layer 3 interfaces. Appendix A provides links to more Cisco IOS bridging information.

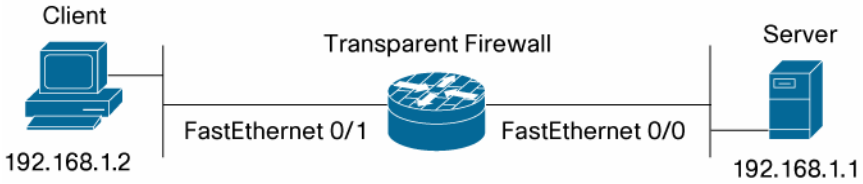
The Transparent Firewall feature was introduced in Cisco IOS Software Release 12.3(7)T. Table 1 indicates which releases added support for particular platforms.

Table 1. Cisco IOS Transparent Firewall Release and Platform Support

Release	Feature	Platform
12.3(7)T	Transparent Firewall	Legacy 800 Series, 1700, 2600XM, 2651, 3700, 7200
12.3(11)T	Added ISR Support for Transparent Firewall	85x/87x, 1800, 2800, 3800
12.4(1)	First mainline release to support Transparent Firewall	All current platforms with IOS Firewall Support

The simplest application of Cisco IOS Transparent Firewall involves bridges between two Ethernet ports on a router, while inspecting all traffic in one direction (client HTTP traffic sending requests to the server, and denying all connections from the server toward the client, for example), shown in Figure 5.

Figure 5. Cisco IOS Transparent Firewall Network Example



The bridging- and firewall-relevant configuration for this simple example is as follows:

```
! generic ip inspection policy
ip inspect name 1-fw tcp audit-trail off
ip inspect name 1-fw udp
ip inspect name 1-fw icmp
!
! set up bridging
bridge irb
!
! interfaces in a bridge group become Layer 2 interfaces, thus they
! have no IP address and must be associated with a bridge group.
interface FastEthernet0/0
 no ip address
 ip access-group 111 in
 bridge-group 1
!
! interfaces in a bridge group become Layer 2 interfaces, thus they
! have no IP address and must be associated with a bridge group.
interface FastEthernet0/1
 no ip address
 ip inspect 1-fw in
 bridge-group 1
!
! bridge interface for bridge group
interface BVI1
 ip address 192.168.1.254 255.255.255.0
!
! define ACL to block connections from “hostile” net
access-list 111 remark private net in-acl
access-list 111 deny   ip any any
!
! define bridge behavior
bridge 1 protocol ieee
bridge 1 route ip
```

Practical Applications

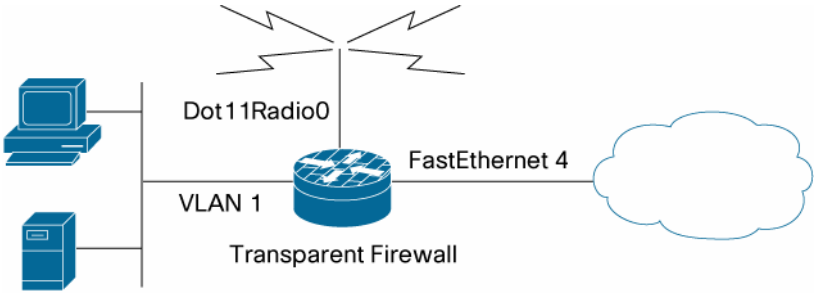
This document explores two different practical applications of Cisco IOS Transparent Firewall. The first example is a scenario where two physical network segments in the same subnet are offered similar policies to the public Internet, but one segment is firewalled from the other. The second example is a configuration for limiting public access to a DMZ, and blocking all outbound DMZ access except Extended Simple Mail Transport Protocol (ESMTP) traffic.

Firewall Separating Two Network Segments in Same Subnet

The first scenario illustrates a public LAN segment, a private LAN segment, and a public Internet connection, as might be seen on a retail network with a public Wi-Fi hotspot. This application is particularly suited for a Cisco integrated services router, such as the Cisco 871W or 1811W, which include built-in Wi-Fi and Ethernet switch interfaces. One LAN segment is a wireless segment for Wi-Fi clients; the other is a group of hosts connected to switch ports on a fixed-configuration router (or a Cisco EtherSwitch® module on a modular router). The transparent firewall will deny access from the Wi-Fi hosts to the hosts on the wired LAN, but hosts on the Ethernet LAN will be able to connect to hosts on the Wi-Fi LAN. All hosts will have the same firewall policy for access to the public Internet.

Both segments will use Port Address Translation (PAT) to access the public Internet, to conserve IP addresses.

Figure 6. Cisco IOS Transparent Firewall Applying Dissimilar Policy to Two LAN Segments in One Subnet



Interfaces assigned to a bridge group do not have an IP address. The interfaces share the address on the bridge virtual interface (BVI), as mentioned in the Transparent Firewall background.

Define the bridge:

```
bridge irb
bridge 1 protocol ieee
bridge 1 route ip
```

Configure the BVI. The BVI will act as the PAT inside interface:

```
interface BVI 1
 ip add 192.168.1.254 255.255.255.0
 ip nat inside
```

Configure the outside interface. This instance will use a static IP address for Ethernet WAN connectivity through a DSL or cable modem:

```
interface FastEthernet 4
 ip address 171.71.58.67 255.255.255.0
 ip nat outside
```

Set up PAT:

```
ip nat inside source list 101 interface FastEthernet 4 overload
access-list 101 permit ip 192.168.1.0 0.0.0.255 any
```

Configure the ACL to restrict access from the wireless LAN to the wired LAN. This ACL will only affect traffic switched to other segments within the bridge group. If you wish to permit access from the Wi-Fi network to the Ethernet network, you must edit this ACL:

```
access-list 102 deny ip any any
```

Apply the ACL to the wireless interface and assign the interface to the bridge group:

```
interface Dot11Radio0
 no ip address
 ip access-group 102 in
 bridge-group 1
```

Define the firewall policy for access from the Ethernet segment to the Wi-Fi segment:

```
ip inspect name transparent tcp
ip inspect name transparent udp
ip inspect name transparent icmp
```

Assign the VLAN 1 interface for the Ethernet LAN to the bridge group and apply the inspection policy. This inspection policy will only inspect traffic moving between interfaces in the bridge group:

```
interface VLAN 1
 no ip address
 ip inspect transparent in
 bridge-group 1
```

Define the firewall policy for access from the Ethernet and Wi-Fi segments to the public Internet:

```
ip inspect name internet tcp
ip inspect name internet udp
ip inspect name internet icmp
```

Apply the Internet access policy to the BVI. This inspection policy only inspects traffic leaving the bridge group:

```
interface BVI 1
 ip inspect internet in
```

Define the ACL to block traffic from the Internet:

```
access-list 103 deny ip any any
interface FastEthernet 4
 ip access-group 103 in
```

This completes the Cisco IOS Transparent Firewall configuration for separating two network segments. The complete configuration is available in Appendix B.

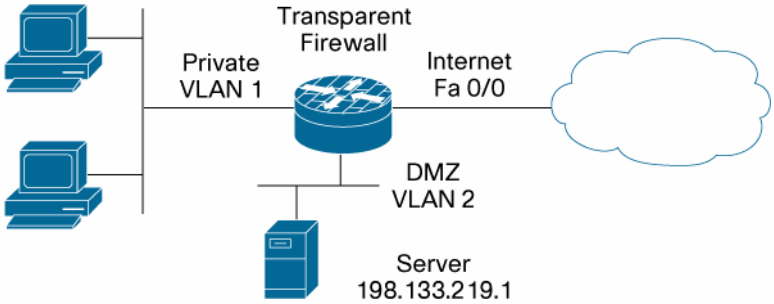
Several commands are useful for troubleshooting firewall activity and viewing the configuration. “Show” and “Debug” commands are discussed in the “Transparent Firewall Troubleshooting and Management” section of this document.

DMZ in Same Subnet as Protected Hosts

The second scenario examines an application to split an IP subnet into a “private” network and a DMZ. This example illustrates an application using a Cisco 1841 Integrated Services Router, in which an ISP has granted a routable subnet of 32 numbers (30 usable addresses) for a small network, so that all hosts on the LAN can be assigned a routable address. Some hosts will be exposed to the Internet to offer e-mail, Web, and other services to hosts on the public Internet.

Exposing a service to the Internet opens a vulnerability to compromise by worms and malicious activity. If the exposed host is compromised, a firewall between the infected host and other hosts is desired to contain as much of the infection as possible. A traditional routing-type firewall would cause segmentation of the available address space and waste IP addresses. A transparent firewall minimizes address loss, improving efficiency of address use. This example could also be viewed as an enterprise assigning a small subnet to a remote location, and needing to restrict headquarters’ access to a portion of the subnet.

Figure 7. Cisco IOS Firewall Separating DMZ and Protected Private LAN



This network’s configuration does not differ appreciably from the previous example, except for the omission of the NAT policy and addition of some ACL entries to minimize service accessibility between network segments. A common practice in use with most Internet service DMZs is to disallow any connections from the DMZ to any host, so that a compromised host will not act as an attack “zombie” or offer a stepping-stone to attack other hosts and mask the true source of an attack.

Define the bridge:

```
bridge irb
bridge 1 protocol ieee
bridge 1 route ip
```

Configure the BVI. The BVI will act as the PAT inside interface:

```
interface BVI 1
 ip add 192.168.1.254 255.255.255.0
```

Configure the outside interface. This instance will use a static IP address for Ethernet WAN connectivity through a DSL or cable modem:

```
interface FastEthernet 0/0
 ip address 171.71.58.67 255.255.255.0
```

Configure the ACL to restrict access from the DMZ LAN to the protected LAN:

```
access-list 102 deny ip any any
```

Apply the ACL to the wireless interface and assign the interface to the bridge group:

```
interface Dot11Radio0
 no ip address
 ip access-group 102 in
 bridge-group 1
```

Define the firewall policy for access from the Ethernet segment to the Wi-Fi segment:

```
ip inspect name transparent tcp
ip inspect name transparent udp
ip inspect name transparent icmp
```

Assign the VLAN 1 interface for the Ethernet LAN to the bridge group and apply the inspection policy:

```
interface VLAN 1
 no ip address
 ip inspect transparent in
 bridge-group 1
```

Define the firewall policy for access from the private and DMZ segments to the public Internet. The ACL will block all connections from the DMZ except ESMTP, which will be checked for protocol conformance by “inspect ESMTP”. The mail protocol must be allowed to pass so that the Internet server can send outbound mail:

```
ip inspect name internet tcp
ip inspect name internet udp
ip inspect name internet icmp
ip inspect name internet esmtp
```

Apply the Internet access policy to the BVI:

```
interface BVI 1
 ip inspect internet in
```

Define the ACL to block traffic from the Internet, but to allow access to services on hosts in the DMZ. For this example, we will permit requests for Web (HTTP), secure Web (HTTPS), mail (SMTP), Internet name resolution (DNS), and file transfer (FTP) to reach the Internet server. We will permit ICMP echo requests to the entire subnet:

```
access-list 103 permit tcp any host 198.133.219.1 eq 80
access-list 103 permit tcp any host 198.133.219.1 eq 443
access-list 103 permit tcp any host 198.133.219.1 eq 25
access-list 103 permit tcp any host 198.133.219.1 eq 53
access-list 103 permit udp any host 198.133.219.1 eq 53
```

```
access-list 103 permit tcp any host 198.133.219.1 eq 25
access-list 103 permit icmp any 198.133.219.0 0.0.0.31
access-list 103 deny ip any any
```

Apply the ACL to the public interface:

```
interface FastEthernet 0/0
 ip access-group 103 in
```

Transparent Firewall Troubleshooting and Management

- **show ip inspect Sessions**: Displays established firewall connections, and firewall connections that are still opening. There is no difference in this output between Transparent Firewall and Layer 3 Firewall.
- **show ip inspect Statistics**: Displays firewall policy activity with regard to type and number of packets handled, as well as switching path in which the packets were handled.
- **show ip inspect name [name]**: Displays inspect policy configuration by policy name.
- **show ip inspect Interfaces**: Displays inspection policies and access lists (inbound and outbound) applied per interface.
- **show ip inspect config**: Displays Cisco IOS Firewall inspection denial-of-service policy, and protocol policies per firewall inspection policy.

Appendix A: Additional Reading for Transparent Firewall

Transparent Firewall Command Reference:
http://www.cisco.com/en/US/products/sw/iosswrel/ps5207/products_feature_guide09186a00801ee193.html

Configuring Transparent Bridging: http://www.cisco.com/en/US/products/ps6350/products_configuration_guide_chapter09186a00800ca767.html

Authentication Proxy

Authentication Proxy (Auth Proxy) offers a mechanism to authenticate and authorize users' access to network resources. Two common applications for Auth Proxy are authenticating users' access from a secure network to the public Internet to avoid network resource abuse, and authenticating user access to sensitive network resources, such as network management resources or human resources and payroll servers. Authentication proxy requires that users provide a valid user name and password before they can access these resources. The user names and passwords may be stored locally on the router, or held on a authentication, accounting, and authorization (AAA) server. AAA servers offer the benefits of offering service to multiple routers, so every router that uses Auth Proxy or other authenticated services does not need to be configured with a new user name any time a user is added on the network. AAA also provides authorization policy information when users provide their credentials, so the users' access to authenticated resources can be filtered on a granular, user-specific basis. Auth Proxy provides HTTP, HTTPS, Telnet, and FTP interfaces to authenticate user access. HTTP and HTTPS provide the benefits of launching a separate browser window for the users' credentials when they try to access HTTP or HTTPS resources protected by the Auth Proxy-enabled router.

Auth Proxy is configured on an interface without direction, as access authentication is always inbound, intercepting the packet before it reaches the inbound ACL. Therefore, an inbound ACL can be configured to block all traffic (`deny ip any any`). Once a user is authenticated by Auth Proxy, ACL Bypass is applied to shunt legitimate traffic around the inbound ACL.

This document does not include AAA server configurations. For more information, please visit:
http://www.cisco.com/en/US/products/ps6350/products_configuration_guide_chapter09186a00804ad9bc.html

HTTP, POP/IMAP, and SMTP/ESMTP Application Inspection

Firewalls and IPSs are becoming increasingly effective at detecting and blocking unauthorized or malicious traffic. Developers of malicious software are making such detection and blocking more difficult by disguising their applications' unwanted traffic as desirable protocols such as HTTP, POP3, IMAP, or SMTP. The unwanted traffic is usually still recognizable as fraudulent, but only after additional, deeper inspection into the data packet to detect indications that the traffic is not legitimate. Cisco IOS Software has introduced several application inspection engine features to address the requirement for deeper packet firewall inspection to block malicious and prohibited application traffic.

This document describes use cases and application backgrounds for Web (HTTP), mail client (POP3 and IMAP), and mail server (SMTP and ESMTP) application inspection services.

Background

Cisco IOS Software Release 12.3(14)T introduced new application inspection engines for three protocols, augmenting the existing ESMTP RFC conformance capability. Most well-known Internet services, such as HTTP, POP3, IMAP, and SMTP/ESMTP are described by RFCs, a step in the Internet Engineering Task Force (IETF) standardization process. RFCs define how Internet services must conduct their activities to ensure compatibility and interoperability in the multivendor environment of the public Internet.

The HTTP application inspection engine offers the greatest range of capabilities by offering the capability to inspect packets for RFC conformance as well as checking various parameters within the content to detect malicious or unauthorized network traffic.

POP and IMAP inspection monitors connection setup to help ensure a secure connection and block unwanted traffic on mail-client ports.

ESMTP inspection has been available since Cisco IOS Software Release 12.3(7)T, augmenting existing SMTP inspection support. ESMTP/SMTP inspection offers protocol compliance checking to block various malicious activities directed at e-mail servers.

HTTP Application Inspection Engine

HTTP is the most commonly used application-layer protocol on the Internet. HTTP offers a flexible, extensible mechanism to support numerous networked applications. Businesses, educational institutions, and government offices that rely on the Internet must allow HTTP traffic through their firewalls to accommodate most Web-based applications. Unfortunately, the pervasive nature of HTTP support has contributed to TCP port 80 being a transmission vector for malicious software such as worms and viruses, as well as offering an effective conduit for concealing other traffic generated by undesirable software such as instant messaging (IM) applications and peer-to-peer (P2P) file-sharing tools.

The Cisco IOS Software HTTP application inspection engine offers flexible application-layer inspection to examine network traffic to detect and take action against malicious or unwanted HTTP traffic. The HTTP application inspection engine offers three fundamental capabilities: protecting servers from malicious clients, protecting clients from malicious or compromised content, and enforcing organizational information systems policies. This document examines three respective examples of some of the capabilities available with HTTP application inspection: HTTP method control, HTTP content verification, and IM and P2P blocking.

All of the HTTP application inspection engine functions offer the option to allow or reset the offending traffic. Furthermore, syslog can send an alarm concerning the violation to a monitoring station. Thus, application inspection can monitor traffic if **allow** is used with a rule, or it can filter out unwanted traffic using the **reset** option.

RFC conformance checking plays an important role in the capability of the HTTP application inspection engine. HTTP was originally defined in RFC 2068, which was superseded by RFC 2616. These RFCs describe the methodology for establishing HTTP sessions and transferring hypertext content. A Web browser and Web server employ a somewhat limited set of requests and responses to carry out their communications over the course of the session.

When malicious traffic is directed at a Web client or Web server, or when non-HTTP applications disguise their communications with a TCP port 80 header, the malicious traffic frequently violates the protocol specification or uses commands outside of the usual command set. Such behavior offers a clear indication that the traffic is somehow atypical and should probably be blocked before entering or leaving the protected network. HTTP application engine RFC conformance checking is an effective solution to stop the most obvious policy-violating traffic, and should be applied with any HTTP application policy.

The HTTP application inspection engine offers several options (Table 2).

Table 2. HTTP Application Inspection Options

Subcommand	Description Summary
audit-trail	Sets application HTTP audit trail
content- length	Detects size of HTTP message
content-type-verification	• Verifies message header • Verifies content type of message data • Verifies content type of “accept” field value (optional)
max-header-length	Detects length of message header
max-uri-length	Detects length of URL in the request message
port-misuse	Detects applictions specified in the HTTP message (P2P, tunneling, IM)
request-method	Detects request methods or extension methods of the message
strict-http	Detects non-HTTP-compliant traffic
timeout	Sets HTTP connection timeout
transfer-encoding	Detects transfer encoding of the message (chunked, compress, or deflate, for example)

The following examples combine several of these options to illustrate application-specific HTTP policy enforcement.

Protecting Servers from Malicious Traffic

Most Web servers on the public Internet are constantly subjected to reconnaissance activity probing for weaknesses in the Web server software or the Web applications they offer. Some of this activity is more effectively addressed with Cisco IPS capabilities, using either a sensor or the IPS features in a Cisco IOS Software router. Examples of inspection more suited to an IPS include:

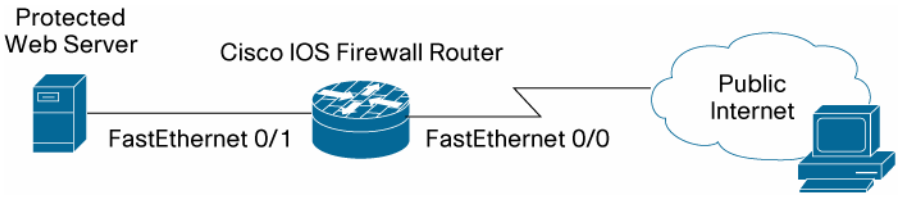
- Searching for a specific text string (for example, worm traffic)
- Watching for activity on multiple ports (for example, port scanning)
- Blocking specific crafted-packet attacks

Web browsers use a standardized group of requests to request and transmit data to and from Web servers. This request is carried in the HTTP header inside the TCP packet, inside the IP packet. By examining the contents of the Web browser’s request to the Web server, application inspection can permit or allow specific types of requests, based on the perceived threat of the different types of requests.

Many attacks against Web servers use specific requests that are fairly uncommon for ordinary Web transactions. An HTTP inspection engine can restrict the particular HTTP methods that Web clients use to communicate with Web servers by checking the HTTP request type field in the HTTP header of the IP packet.

In a simple network, a Web server is separated from the Web client by a Cisco IOS router (Figure 8).

Figure 8. Protect Web Servers with HTTP Application Inspection



To apply the server protection just discussed to this simple network, configure an HTTP application inspection policy:

```
appfw policy-name method-control
 application http
  strict-http action reset alarm
  request-method rfc put action reset alarm
```

This application inspection policy must then be applied to an inspection policy, either an existing inspection set or a minimal set that will only employ application inspection:

```
ip inspect name my-fw appfw method-control
```

Finally, the policy must be applied in the direction that the inspection occurs:

```
interface fastethernet 1/0
 ip inspect my-fw out
```

To see the HTTP application inspection engine configuration, issue this command at the privileged EXEC prompt:

```
sh ip inspect config
```

Protecting Clients from Malicious Content

Web browser clients are vulnerable to malicious Web content that they might mistakenly download through a misleading link or from legitimate Websites that have had their content replaced with malicious material. As with server protection, some client protection tasks are best left to an IPS. Searching for complex worm code strings in Web content is not an appropriate task for HTTP application inspection.

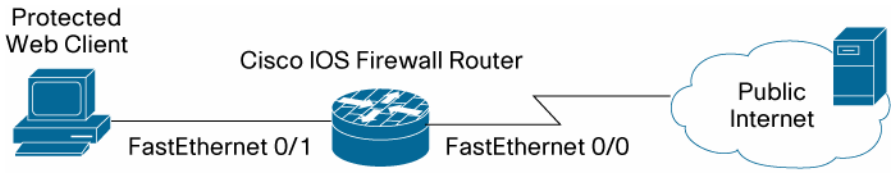
An appropriate use of the HTTP application inspection engine for client protection applies the content-type-verification feature. Several documented Web browser vulnerabilities involve a browser's susceptibility to intentionally mislabeled content. When a Web browser requests content from a Web server, the Web server indicates to the Web browser the type of embedded content in its reply. The Web browser usually passes the content to an application on the workstation that is associated with the type of content that the Web server indicated in its reply. However, if the content is mislabeled, the application that opens the content might try to implement the content according to the content's file header information or the file name extension. As an example, consider an attack that takes advantage of this weakness to distribute a Microsoft Visual Basic script file with an .mpg or .avi extension on the file name (for instance, an attacker applies the name "attack.mpg" to the attack.vbs file). This causes a default installation of Microsoft Internet Explorer to pass the file to Microsoft Windows Media Player. Media Player receives the file from Internet Explorer and recognizes that the content does not have the appropriate attributes to be a genuine .mpg or .avi file, but it makes its best

effort to open the file nonetheless. Media Player might recognize the file as a Visual Basic script and execute it accordingly. If the Visual Basic script contains the appropriate content, the workstation will be compromised.

The content-type-verification feature of the HTTP application engine checks Web content replies to protected browsers to verify that the embedded content matches the content type indicated by the Web server. Like most HTTP application inspection features, content-type-verification can be configured to allow or reset traffic that doesn't pass the content-type check. Obviously, the only appropriate action for invalid traffic is to reset the Web client connection so that the invalid traffic is not allowed to pass through to the Web client.

Once again, this exercise assumes a simple network consisting of a Web server separated from the Web client by a Cisco IOS router (Figure 9).

Figure 9. Protect Web Clients with HTTP Application Inspection



To apply the server protection just discussed to this simple network, configure an HTTP application inspection policy:

```
appfw policy-name chk-content
 application http
  strict-http action reset alarm
  request-method rfc put action reset alarm
```

This application inspection policy must then be applied to an inspection policy, either an existing inspection set or a minimal set that will only employ application inspection:

```
ip inspect name my-fw appfw chk-content
```

Finally, the policy must be applied in the direction that connection was initiated:

```
interface fastethernet 1/0
 ip inspect chk-content in
```

To see the HTTP application inspection engine configuration, issue this command at the privileged EXEC prompt:

```
sh ip inspect config
```

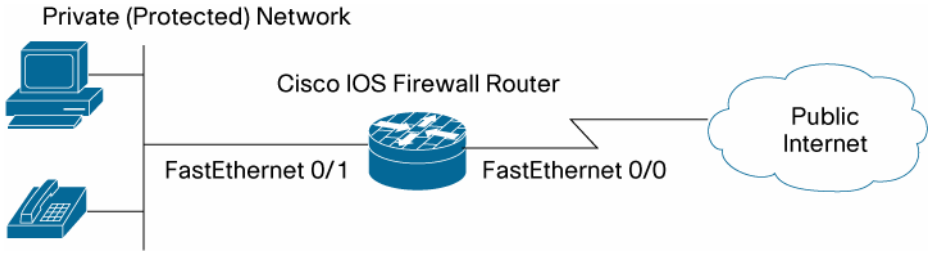
Blocking Instant Messaging Traffic

Instant messaging applications offer substantial productivity gains when they are used for business applications such as business-focused discussion among colleagues. Unfortunately, they can cause substantial productivity losses if employees spend a great deal of time using public IM services such as Yahoo! Messenger or AOL Instant Messenger. P2P file-sharing networks can consume a large portion of an organization's Internet connectivity bandwidth and can place a substantial liability burden on an organization's shoulders if company resources are used to share copyrighted media such as music or films.

Some implementations of IM applications and P2P file sharing software that offer the capability to conceal their traffic within a TCP port 80 (HTTP) header do not implement the complete RFC 2616 dialogue methodology. The Application Inspection Engine's "strict-rfc" option recognizes these applications' traffic as it is clearly not HTTP traffic. However, some IM and P2P applications implement their TCP port 80 traffic with a sufficiently high degree of fidelity to RFC 2616 to make the traffic indistinguishable from legitimate HTTP traffic. The Application Inspection Engine can detect this traffic by enabling the "port-misuse" option, which currently recognizes Yahoo! Messenger IM, KaZaa and Gnutella P2P file sharing, and TCP port 80-based tunneling by HTTPPort/HTTPHost, GNU Httptunnel, GotoMyPC, Firethru, and the Http-tunnel.com client. Applying the port-misuse feature coupled with strict HTTP RFC compliance checking helps assure that valid HTTP dialogue is conducted according to the RFC's specification, and allows recognition of known IM and P2P traffic that closely emulates legitimate HTTP traffic.

This exercise assumes a simple network consisting of a Web browser accessing the Web using a Cisco IOS router (Figure 10).

Figure 10. Blocking Instant Messaging Traffic on a Simple Network



To apply instant messaging and P2P blocking on this simple network, configure an HTTP application inspection policy:

```
appfw policy-name no-abuse
 application http
  strict-http action reset alarm
  port-misuse default action reset alarm
```

This application inspection policy must then be applied to an inspection policy, either an existing inspection set or a minimal set that will only employ application inspection:

```
ip inspect name my-fw appfw no-abuse
```

Finally, the policy must be applied in the direction that the inspection occurs:

```
interface fastethernet 0/1
 ip inspect my-fw in
```

To see the HTTP application inspection engine configuration, issue this command at the privileged EXEC prompt:

```
sh ip inspect config
```

Securing E-Mail Client Service

Cisco IOS Software offers application inspection services for two common Internet e-mail client protocols: IMAP (TCP port 143) and POP3 (TCP port 110). E-mail client application inspection helps ensure that clients negotiate a valid client connection with the server and, if necessary, use secure authentication. E-mail client application inspection addresses two concerns regarding traffic on the e-mail client ports.

The first concern is that the e-mail client ports, which are frequently left open to the public Internet to allow employees to send and receive e-mail when they are away from the office, are not used for other applications such as a “back door” or other unauthorized service on the permitted ports.

The second concern regarding remote e-mail client access is when users have configured their e-mail client to secure their authentication credential exchange. Most POP3 and IMAP client applications default to passing the user’s user name and password to the server as “cleartext,” meaning this information is not encrypted to conceal its credentials. If an attacker intercepts a user name and password being transmitted as cleartext, the attacker could read the user name and password and could masquerade as the authorized user. Such a situation is especially problematic if the credentials are used for multiple services.

To help ensure security on POP3 and IMAP ports and maximize connection security between mail clients and servers, e-mail client traffic inspection monitors the clients’ negotiations with servers to help ensure that the session setup follows the requirements of the respective RFCs. RFC 1939 defines the POP3 client negotiation, and RFC 3501 defines the IMAP client negotiation (Figure 15).

E-mail client inspection can be configured to log and/or reset client negotiations that violate the RFC requirements. If neither **log** nor **reset** is specified, e-mail client application inspection not be useful. Furthermore, if secure login is required, the **secure-login** switch will require that clients ask for a secure negotiation. If the client does not ask for a secure login, the client will be logged out or reset according to the configuration.

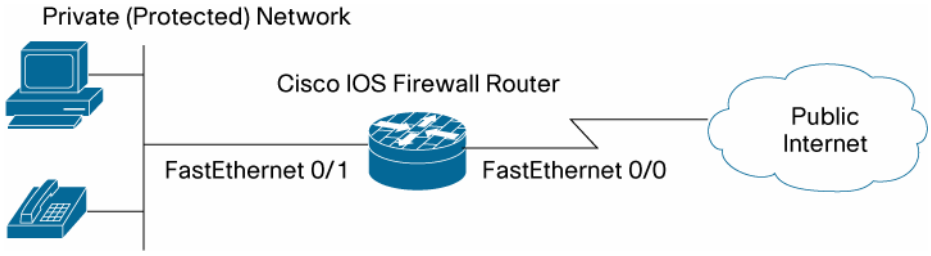
Table 3 shows the options that are available in e-mail application inspection configuration.

Table 3. E-Mail Application Inspection Configuration Options

Option	Description
inspection-name	Names the set of inspection rules. To add a protocol to an existing set of rules, use the same inspection-name as the existing set of rules.
Protocol	Indicates which protocol (POP or IMAP) the inspection engine will monitor.
alert {on | off}	For each inspected protocol, the generation of alert messages can be set be to **on** or **off**. If no option is selected, alerts are generated based on the setting of the **ipv6 inspect alert-off** command. (Optional)
audit-trail {on | off}	For each inspected protocol, **audit-trail** can be set to **on** or **off**. If no option is selected, **audit-trail** messages are generated based on the setting of the **ipv6 inspect audit-trail** command. (Optional)
timeout *seconds*	To override the global TCP or UDP idle timeouts for the specified protocol, specify the number of seconds for a different idle timeout. This timeout overrides the global TCP and UDP timeouts but will not override the global DNS timeout. (Optional)
Reset	Specifies that the TCP connection will be reset if the client enters a nonprotocol command before authentication is complete.
secure-login	Forces client to authorize using a secure method for login. Firewall will not allow plain text login (password in the clear) using this option.

This example assumes a simple network consisting of an e-mail client accessing an e-mail server using a Cisco IOS router (Figure 11).

Figure 11. Simple Network Diagram



To apply the e-mail client traffic protection just discussed to this simple network, define an IP inspection statement in an existing inspection policy or add the statement to an existing inspection set:

```
ip inspect name mail-clients pop3 log reset secure-login
ip inspect name mail-clients imap log reset secure-login
```

Finally, the policy must be applied in the direction that the inspection occurs:

```
interface fastethernet 0/1
 ip inspect mail-client in
```

To show the e-mail client application inspection configuration, issue the following **show** command:

```
show ip inspect config
```

To see e-mail client application inspection engine messages for individual events, enable debug for the appropriate protocol:

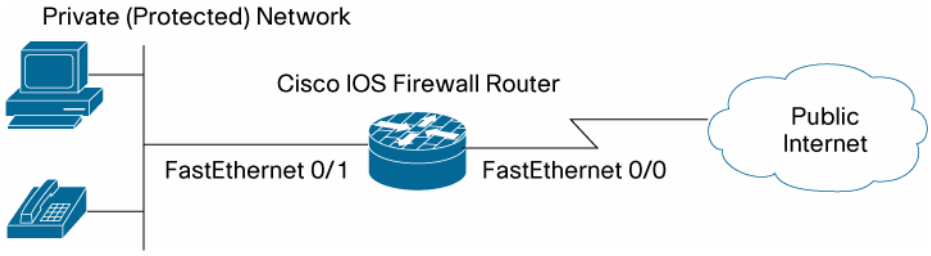
```
debug ip inspect pop3
debug ip inspect imap
```

Securing E-Mail Server Traffic

Cisco IOS Software introduced SMTP inspection in Release 12.0(1)T and augmented the feature to support ESMTP in Release 12.3(7)T. SMTP inspection and ESMTP inspection monitor the connection between the client and server to help ensure that only valid commands that are specified by the RFCs are allowed between the two participants in SMTP and ESMTP dialogues. This restriction prevents unauthorized use of the SMTP and ESMTP port (TCP port 25) so that mail servers are protected from invalid, possibly malicious traffic, and so that exploit software such as back doors and rootkits is not allowed to use TCP port 25.

This example assumes a simple network consisting of an e-mail server connecting to the public Internet using a Cisco IOS router (Figure 12).

Figure 12. Simple Network Diagram



To apply the e-mail server traffic protection to this simple network, define an SMTP or ESMTP inspection statement for use by itself or add the statement to an existing inspection set:

```
ip inspect name mail-server [ smtp | esmtp ]
```

SMTP traffic inspection configuration and ESMTP traffic inspection configuration in any given inspection set contain mutually exclusive commands. ESMTP inspection is a superset of SMTP inspection. SMTP inspection should only be applied if the mail server is to be limited to supporting only SMTP. Otherwise, ESMTP inspection should be applied.

Apply the policy in the direction that the inspection occurs:

```
interface fastethernet 0/1
 ip inspect mail-server in
```

To show SMTP and ESMTP application inspection configuration, issue the following **show** command:

```
show ip inspect config
```

References

ESMTP inspection engine: http://www.cisco.com/en/US/products/sw/iosswrel/ps5207/products_feature_guide09186a00801ed6ee.html

POP and IMAP inspection engine: http://www.cisco.com/en/US/products/sw/iosswrel/ps5207/products_feature_guide09186a00803f85bd.html

HTTP inspection engine: http://www.cisco.com/en/US/partner/products/sw/iosswrel/ps5207/products_feature_guide09186a0080420260.html

Blocking Instant Messaging and Peer-to-Peer File Sharing Applications with Cisco IOS Software Release 12.3(14)T

Most organizations view IM and P2P applications as frivolous consumers of expensive resources—employee time and network bandwidth. Furthermore, some P2P networks can act as a conduit for malicious software such as worms, offering an easy path around firewalls into an organization to compromise desktop computing resources.

Cisco IOS Software Release 12.3(14)T introduced application inspection engines and granular inspection, two critical new features that allow Cisco IOS Firewall to control IM and P2P applications on networks. This document offers some sample configurations to use these features to monitor and block IM and P2P file sharing traffic.

Background

P2P and IM traffic generally offer two modes of operation—a native mode, where the application runs on a uniquely defined set of TCP or UDP ports, and “HTTP cloaked” mode, in which the application masquerades as HTTP (TCP port 80) traffic in order to gain passage through firewalls

and other network policy controls. Some of the more advanced P2P and IM applications implement sufficient RFC 2616 dialogue to appear as a legitimate conversation between a Web browser and a Web server.

Prior to Release 12.3(14)T, Cisco IOS Software was bound by two major restrictions in the control of P2P and IM applications—a limited list of applications that were supported in Cisco IOS Firewall Stateful Inspection (formerly known as Context-Based Access Control [CBAC]), and lack of application inspection capability.

Cisco IOS Firewall Stateful Inspection is the basis of the Cisco IOS Firewall feature set. If a specific application was not built into Cisco IOS Firewall (see the list of original supported protocols in Appendix 1), the **inspect tcp** or **inspect udp** commands were used to watch for any outbound connection activity through a firewall, and anticipated return traffic was subsequently allowed through firewall blocking policies with ACL bypass capability. Unfortunately, these commands allow all traffic that is not specifically filtered out to make a connection with the appropriate server through a firewall, and return traffic is allowed back in. This mode of operation offers little control in allowing or disallowing specific protocols.

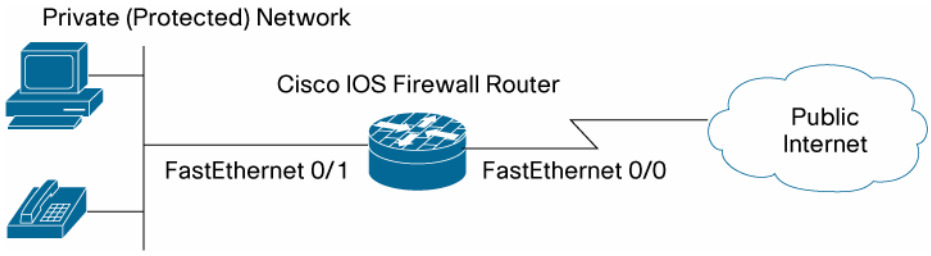
From an application inspection standpoint, HTTP inspection was one of the more thorough protocol inspections that Cisco IOS Firewall offered. However, even if an extremely restrictive Cisco IOS Firewall policy allowing only “HTTP out” was applied, users might still be able to use P2P and IM applications that offered HTTP cloaking.

Cisco IOS Software Release 12.3(14)T introduced application inspection and granular inspection capabilities to address both of these shortcomings.

Example Network

We can examine a simple network to build an example of an effective inspection policy that will prohibit P2P and IM traffic, and that will offer control over cloaked applications that try to exploit TCP port 80 to gain access though the firewall (Figure 13). This network consists of one or more client PCs in a private network, connected to the public Internet through a Cisco IOS router running Cisco IOS Software Release 12.3(14)T.

Figure 13. Example Network

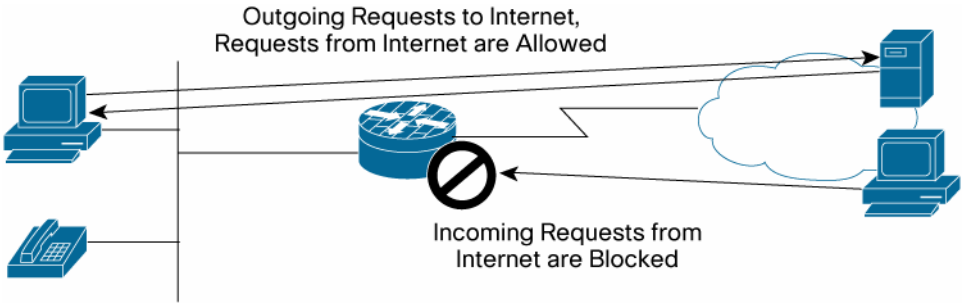


This sample network needs standard services, such as Web access (HTTP and Secure HTTP [HTTPS]), Internet e-mail (SMTP, POP3, and IMAP), packet voice (H.323), DNS lookup, FTP, Network Time Protocol (NTP), and ICMP. Furthermore, the network users employ Virtual Network Computing (VNC), an open-source remote console application that runs by default on TCP port 5900, and they need HTTP access on atypical ports (TCP port 81 and 8080) for connectivity to vendor or customer e-commerce Webpages.

Background

Cisco IOS Firewall uses Cisco IOS Firewall Stateful Inspection to restrict a public network’s access to protected networks, while maintaining the private network’s ability to access resources located in the public network (Figure 14).

Figure 14. Cisco IOS Firewall Stateful Inspection



Cisco IOS Firewall Stateful Inspection protects networks with two basic components. ACLs restrict inbound connections, and stateful inspection examines activity traversing the Cisco IOS Firewall from the protected network to the public network and anticipates the return traffic. Stateful inspection is a mechanism that observes the initiation, maintenance, and closure of network data connections.

Cisco IOS Firewall Stateful Inspection with granular inspection supports several of the specific application protocols listed in Appendix 1. Some of these protocols are common, simple protocols such as HTTP and Telnet, which only use one connection between client and server (or peers) to request and return application data. More complex supported protocols, such as FTP and H.323, employ a control channel to establish communications and a secondary data channel to transmit application data.

Some common protocols are not specifically predefined in IP inspection. Prior to Cisco IOS Software Release 12.3(14)T, **ip inspect tcp** and **ip inspect udp** were used as universal options for any services not covered by specific inspection services to inspect outgoing traffic and allow return traffic through a Cisco IOS Firewall's inbound permission ACL. Unfortunately, the **inspect tcp** option's capability to allow any return traffic is problematic in circumstances where specific protocols must be disallowed, particularly when a complex application such as IM or P2P employs unpredictable port numbers and other mechanisms that make the traffic difficult to detect and block as it leaves the network. Undesired complex applications can be blocked by denying all return traffic except the traffic allowed by specific inspection services.

Controlling P2P and Instant Messaging Applications

Cisco IOS Software Release 12.3(14)T introduced granular protocol inspection, which offers the capability to use PAM protocol definitions with Cisco IOS Firewall Stateful Inspection. PAM allows users to define specific, named protocols. This significantly changes the older paradigm of employing specific inspection statements for advanced protocols that required comprehensive inspection to allow return access back through a firewall (commonly called "fixup" on Cisco PIX® products), then using **inspect tcp** to cover simpler protocols that do not require close scrutiny to allow additional data connections. Granular inspection uses PAM with Cisco IOS Firewall Stateful Inspection to associate user-defined application labels to traffic on specific ports, in order to define the list of desired traffic to be inspected so the return traffic "pinholes" are allowed in the inbound ACL; this protects the private network from unwanted access from the public network. Since the complete list of desired traffic can be specified, there is no need for **inspect tcp** to offer coverage for previously unrecognized application traffic.

Granular inspection is an effective solution for blocking applications using port-hopping techniques that defy ACLs attempting to block the traffic, because the only traffic that is allowed to return through the firewall is running on the specific desired ports that the user allows with existing, predefined inspection capabilities, as well as user-specific, PAM-defined granular inspection policies. **inspect tcp** does not offer this application-specific mechanism to permit traffic—it simply anticipates all traffic running over TCP.

Application Inspection

Granular inspection leaves some openings that advanced P2P and IM applications may exploit. Most networks allow HTTP traffic through their firewalls, as it is the standard transport of many business applications, including ordinary Web traffic. Many IM and P2P applications have developed mechanisms to disguise their traffic within TCP port 80 (HTTP) traffic, thus offering their application an additional mechanism to work

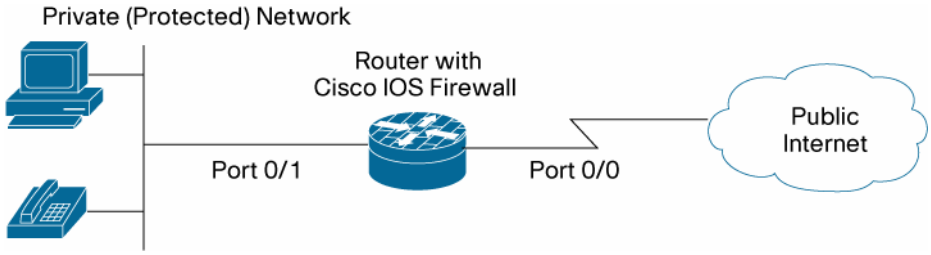
around restrictive firewalls. To address this issue, Cisco IOS Software Release 12.3(14)T introduced application inspection. The HTTP Application Inspection Engine offers the **port-misuse** option to scan traffic for specific known applications that disguise their undesired traffic as legitimate HTTP traffic. Presently, HTTP inspection can recognize Yahoo! Messenger traffic, Gnutella and KaZaA file sharing activity, and some applications that can tunnel other traffic through TCP port 80 to avoid an otherwise restrictive firewall.

By combining granular inspection with the HTTP Application Inspection Engine, network engineers can allow desired protocols' traffic to return to their networks through access lists that protect the private network from unwanted public Internet traffic. Application inspection can control specific unwanted application traffic that has been concealed inside legitimate HTTP traffic.

Configuring Cisco IOS Firewall

Consider a simple network consisting of a Cisco IOS router with two Fast Ethernet ports. Port 0/0 is connected to the public Internet through a broadband connection, and Port 0/1 is connected to an Ethernet switch in the private network (Figure 15).

Figure 15. Example Network



The first configuration step restricts hosts on the public Internet from reaching the protected network with an ACL blocking all traffic from the public Internet, and applying the list to an interface:

```
access-list 101 deny ip any any

interface Fastethernet 0/0
  ip access-group 101 in
```

In the next step, specific inspection statements are configured based on the acceptable traffic that the router will allow out through the firewall, and on the expected return traffic:

```
ip inspect name my-ios-fw http
ip inspect name my-ios-fw https
ip inspect name my-ios-fw esmtp
ip inspect name my-ios-fw pop3
ip inspect name my-ios-fw imap3
ip inspect name my-ios-fw dns
ip inspect name my-ios-fw ftp
ip inspect name my-ios-fw ntp
ip inspect name my-ios-fw icmp
```

Cisco IOS Software supports the most popular Internet protocols, as well as several protocols that require additional effort to accommodate secondary data connections (Appendix 1). This example requires support for VNC, which is not supported by default IP inspection capability;

VNC runs on TCP port 5900 by default. Granular protocol inspection provides the capability to configure inspection for specific protocols that are not natively supported by IP inspection. Configure inspection for VNC by defining the PAM entry for the protocol. Note: User-defined protocol labels must begin with “user-”:

```
ip port-map user-vnc port tcp 5900
```

Next, apply the new protocol to the stateful inspection set:

```
ip inspect name my-ios-fw user-vnc
```

Now that the IP inspection set is complete, apply the inspection policy to the outbound traffic. Since this example protects traffic sourced on the private side of the router, **ip inspect in** is applied to the private interface. The router will inspect traffic passing from the private network to the public Internet, and the appropriate ACL bypass entries will be entered on the public side of the router to allow desired return traffic from the public Internet to pass back to the private network.

```
interface fastethernet 0/1
  ip inspect my-ios-fw in
```

The Cisco IOS Firewall Stateful Inspection configuration that you have defined blocks unwanted connections from the public Internet and allows return traffic for desired applications. Some P2P and IM applications may be able to carry their traffic over TCP port 80, so we’ll use the HTTP Application Inspection Engine to inspect further into TCP port 80 packets, and look for indications of the unwanted P2P and IM traffic.

First, define the application inspection policy name, then configure HTTP inspection. Next, set up the policy. This policy will only inspect TCP port 80 traffic for misuse by non-HTTP traffic; you may wish to define other application inspection features. Check the configuration reference list at the end of this document for details on using other HTTP application inspection features:

```
appfw policy-name abuse-control
  application http
    port-misuse default action reset alarm
```

Apply the application inspection policy to the existing inspection set:

```
ip inspect name my-ios-fw appfw abuse-control
```

This completes the configuration for Cisco IOS Firewall with granular inspection and application inspection.

Verifying Cisco IOS Firewall Capability

You can check the Cisco IOS Firewall configuration and activity with several **show** commands:

`show ip inspect config`	Displays protocol timeouts and limits for Cisco IOS Firewall session activity.
`show ip inspect interfaces`	Displays interfaces with Cisco IOS Firewall rules applied.
`show ip inspect name`	Displays configuration of specific Cisco IOS Firewall rules.
`show ip inspect sessions`	Displays active sessions, including source and destination host addresses and port numbers.
`show ip inspect statistics`	Displays statistics for current active sessions, total sessions reset, session creation rate, number of sessions since Cisco IOS Firewall was configured or the router was rebooted, and other Cisco IOS Firewall statistics.

`show ip inspect all`	Displays all Cisco IOS Firewall information in the previous five commands.

References

Configuring Cisco IOS Firewall Stateful Inspection:
http://www.cisco.com/en/US/partner/products/sw/iosswrel/ps1835/products_configuration_guide_chapter09186a00800ca7c5.html

Configuring HTTP application inspection:
http://www.cisco.com/en/US/partner/products/sw/iosswrel/ps5207/products_feature_guide09186a0080420260.html

Configuring granular protocol inspection:
http://www.cisco.com/en/US/partner/products/sw/iosswrel/ps5207/products_feature_guide09186a008040afd7.html

Configuring PAM:
http://www.cisco.com/en/US/partner/products/sw/iosswrel/ps1831/products_configuration_guide_chapter09186a00800d981c.html

CAVEATS

- This document does not illustrate a security policy that must be followed; rather, it explains how to initiate such a policy.
- Separate documentation outlines IDS design.
- Cisco IOS Firewall does not currently protect ICMP and it is not covered. For the purposes of troubleshooting, ICMP can be allowed. However, as a security measure, it is not advisable to permit ICMP to pass. It is advisable to use ACLs to control the various types of ICMP packets.
- ACLs can be used for IP Spoofing Defense, so that traffic is denied if it comes from outside and has spoofed the inside IP address.
- All firewall states are internal to a single router; therefore, there is currently no provisioning available for redundant firewall routers. Configurations with asymmetric routing (only one direction of each session passes through the router) cause limitations for dynamic modification of ACLs.
- Due to the connectionless nature of UDP traffic, the advanced firewall engine relies on idle timers for closing client-server channels. These timers are not specifically covered in this document, but can be configured for certain types of UDP traffic (i.e. DNS), which can limit very large numbers of open UDP channels.
- A separate document explains Advanced Firewall Inspection/Authentication Proxy with IPSec VPN. Although IPSec VPNs can affect a security policy, Cisco IOS Firewall addresses IPSec type packets directly in the packet flow process.

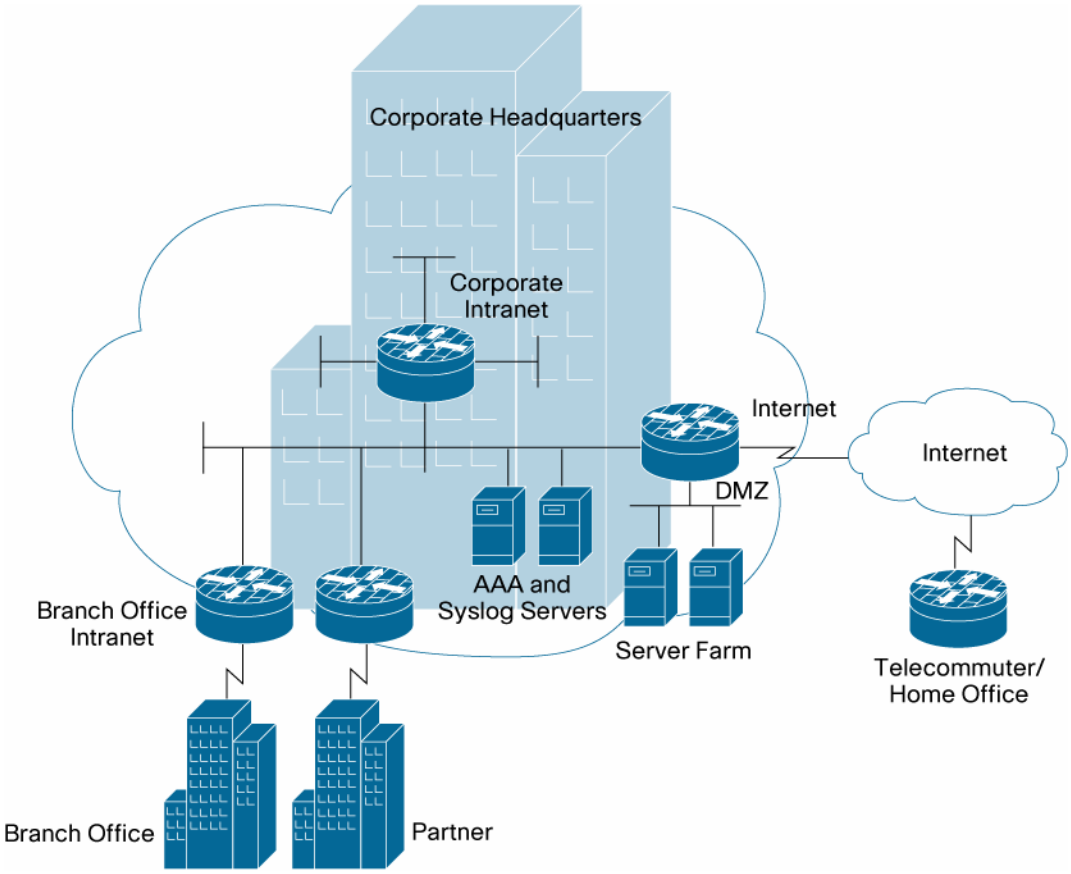
DEPLOYMENT SCENARIOS

Corporations today can implement the Cisco IOS Firewall to secure the following perimeters (Figure 21):

- **Internet Perimeter**—Connect to the Internet via WAN
- **Corporate Intranet Perimeter**—Segmentation of corporate intranet by departments
- **Branch Office Intranet**—Connect to the branch office via WAN, such as Frame Relay
- **Extranet Perimeter/Partner**—Connect to the partners or suppliers via WAN, such as Frame Relay
- **Home Office/Telecommuter**—Connect to the Internet and corporate intranet via IPSec secure LAN-LAN tunnel

Figure 16 details several key components of a typical enterprise network. Each component is broken down into sections to better understand the configuration tasks and security implications for each segment within the network. Security policies and intruder concerns vary widely in each scenario. These implications often involve “political” relationships among departments and/or partners, which affects the technical implementation of the defined security policies for the firewall. Each policy should be reviewed periodically to best address access and security concerns.

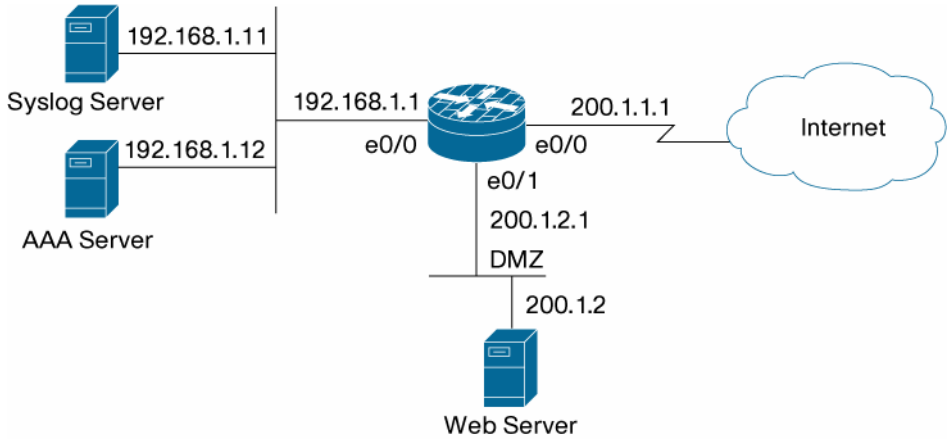
Figure 16. Corporate Network



Internet Perimeter

Cisco IOS Firewall used as corporate Internet firewall is shown in Figure 17.

Figure 17. Internet Perimeter Firewall



- The Web server is on the DMZ network to protect the inside network from outside access through the Web server, in the event the Web server were compromised.
- Cisco IOS Firewall provides real-time log messages, including alerts, to the syslog server
- Auth Proxy with an AAA server is used to control Internet and DMZ access per user
- Simple example: Static NAT and PAT

Internet Firewall Policy

This policy outlines a typical set of protocols that might be used within a corporate network when accessing the Internet. Care has been taken to specifically limit traffic from the general inbound Internet. A DMZ is one method to segment off general corporate users from Internet-accessible servers. This method limits the liabilities that can be involved with any one attack to any specific part of the network.

The company assigns everyone a user ID and password for authentication. The protocols the company wants to secure are SMTP, FTP, H.323, DNS, Telnet, SQLnet, and RealAudio. The protocols—HTTP, SSL, FTP, VDOlive, NetShow, and H.323—are allowed for outside to access the Web server in DMZ. As a policy, no one is allowed to initiate any connection from DMZ.

- Authentication proxy is enabled for outbound access from inside.
- Allow the firewall to inspect and secure the following protocols for all authenticated inside users that go out to the Internet and DMZ.
  - Protocols to be inspected from inside: SMTP, FTP, H.323, TCP, UDP, SQLnet, RealAudio.
  - HTTP need not be specified unless Java blocking is desired. Specifying ‘TCP’ allows inspections for single channel TCP protocols that include HTTP and Telnet. ‘UDP’ is specified for DNS.
- All of the following protocol traffic should be allowed into the DMZ from the outside; it should not be allowed inside:
  - HTTP, FTP, VDOlive, NetShow, and H.323.
- Cisco IOS Firewall does not currently inspect SSL. TCP port 443 will be opened to permit SSL.
- No traffic initiated from the DMZ is to be allowed outside of the DMZ.

Internet Firewall Sample Configuration

```
FWRouter#sh run
Building configuration...

Current configuration:
!
version 12.0
service timestamps debug uptime
service timestamps log uptime
service password-encryption
!
hostname FWRouter
!
! Enable authentication proxy globally.
!
boot system flash
aaa new-model
aaa authentication login default group tacacs+
aaa authentication login console_line line none
```

```
aaa authentication login vty_line line none
aaa authorization exec vty_line none
aaa authorization auth-proxy default group tacacs+
enable secret 5 $1$VZGK$Els0ipcmt8Pe0R98RMEds0
enable password 7 0822455D0A16544541
!
username cisco password 7 121A0C041104
!
ip subnet-zero
no ip source-route
no ip finger
!
! INFIRE is configured for traffic destined for the internet or the DMZ. Inspection
! is configured inbound on the inside interface (e0/0)
!
ip inspect name INFIRE smtp
ip inspect name INFIRE ftp
ip inspect name INFIRE tcp
ip inspect name INFIRE udp
ip inspect name INFIRE sqlnet
ip inspect name INFIRE realaudio
ip inspect name INFIRE h323
!
! OUTFIRE is setup for traffic heading from the internet. This traffic is can go
! ONLY to the DMZ and is applied to inbound traffic on the outside interface (s0/0)
!
ip inspect name OUTFIRE tcp
ip inspect name OUTFIRE ftp
ip inspect name OUTFIRE vdolive
ip inspect name OUTFIRE netshow
ip inspect name OUTFIRE h323
ip auth-proxy name PROXY http    ! Enables authentication proxy for http traffic.
!
! e0/0 is the inside interface to the Corp network.
!
interface Ethernet0/0
 ip address 192.168.1.1 255.255.255.0
 ip access-group 101 in    ! TACACS+ traffic
 ip access-group 102 out ! lock-down acl
 no ip directed-broadcast
 no ip proxy-arp
 ip nat inside
 ip inspect INFIRE in    ! firewall inspection for inbound traffic.
```

```
ip auth-proxy PROXY    ! Associates authentication proxy on the inside interface.
 no cdp enable
!
! s0/0 is the interface closest to the internet. The outside interface.
!
interface Serial0/0
 ip address 200.1.1.1 255.255.255.0
 ip access-group 121 in    ! allows internet initiated traffic.
 no ip directed-broadcast
 ip nat outside
 ip inspect OUTFIRE in    ! firewall inspection for traffic coming from the internet.
 no cdp enable
!
! this is the DMZ interface.
!
interface Ethernet0/1
 ip address 200.1.2.1 255.255.255.0
 ip access-group 111 in    ! DNS traffic.
 ip access-group 112 out ! allows specific traffic to the DMZ server.
 no ip directed-broadcast
 no ip proxy-arp
 no cdp enable
!
! an unsed interface.
!
interface Serial0/1
 no ip address
 no ip directed-broadcast
 shutdown
 no cdp enable
!
! Inside addresses 192.168.1.21-25 will be translated to 200.1.1.21-25 respectively.
! static nat is used as a simple example. Dynamic nat or a combination can be used
! depending on the address requirements.
!
ip nat inside source static 192.168.1.25 200.1.1.25
ip nat inside source static 192.168.1.24 200.1.1.24
ip nat inside source static 192.168.1.23 200.1.1.23
ip nat inside source static 192.168.1.22 200.1.1.22
ip nat inside source static 192.168.1.21 200.1.1.21
ip classless
! configures the HTTP server to allow AAA authentication.
ip http server
```

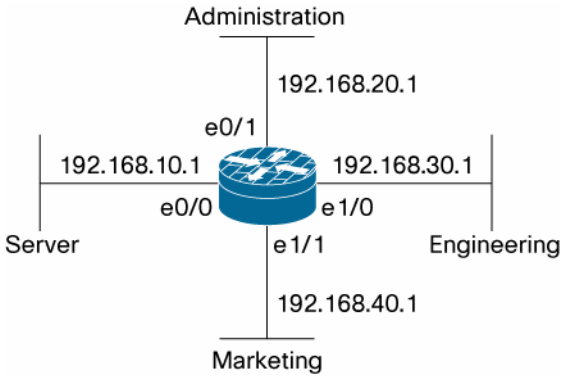
```
ip http authentication aaa
ip http access-class 10
!
! syslog is configured for reporting etc.
!
logging facility syslog
logging 192.168.1.11
access-list 10 deny any
!
! acl 101 blocks all traffic except TACACS+ communications with the AAA server.
! With successful user authentication, Authentication Proxy will override this ACL
! and open connections based on the AAA ACL policy. This ACL is applied to the
! inbound on the inside interface (e0/0)
access-list 101 permit tcp host 192.168.1.12 eq tacacs host 192.168.1.1
! acl 102 locks down traffic heading to the inside net. It is applied to the
! inside interface for outbound traffic.
access-list 102 deny ip any any
! acl 111 permits DNS requests from the HTTP server on the DMZ net. (e0/1)
access-list 111 permit udp host 200.1.2.11 any eq domain
! acl 112 permits internet traffic inspected by the firewall destined to the DMZ.
access-list 112 permit tcp any host 200.1.2.11 eq www
access-list 112 permit tcp any host 200.1.2.11 eq ftp
access-list 112 permit tcp any host 200.1.2.11 eq 7000
access-list 112 permit tcp any host 200.1.2.11 eq 1755
access-list 112 permit tcp any host 200.1.2.11 eq 1720
! acl 121 corresponds to acl 112. it allows internet traffic inspected by the
! firewall to the server on the DMZ. It is applied to inbound traffic on the
! outside interface (s0/0)
access-list 121 permit tcp any host 200.1.2.11 eq www
access-list 121 permit tcp any host 200.1.2.11 eq ftp
access-list 121 permit tcp any host 200.1.2.11 eq 7000
access-list 121 permit tcp any host 200.1.2.11 eq 1755
access-list 121 permit tcp any host 200.1.2.11 eq 1720
no cdp run
! configure the TACACS server’s address and key.
tacacs-server host 192.168.1.12
tacacs-server key cisco
!
! The following is down in order to secure access to the router via the console
! before enter AAA configuration commands. note: AAA commands take effect before
! saving out of config mode.
!
line con 0
```

```
 exec-timeout 0 0
 password 7 0822455D0A16
 login authentication console_line
 transport input none
line aux 0
line vty 0 4
 password 7 01100F175804
 login authentication vty_line
!
!
!
End
```

Corporate Intranet Perimeter

Cisco IOS Firewall used as a corporate intranet firewall is shown in Figure 18.

Figure 18. Corporate Intranet Perimeter Firewall



- Each interface can be configured so that each subnet has its own policy
- Sub-interfaces of WAN protocols are supported (i.e. ATM and Frame Relay)

Corporate Intranet Firewall Policy

Each department has an individual policy of permissible protocols, including the standard protocols Telnet, FTP, and UDP. As a policy, no one will be allowed to initiate any session from the server subnet.

The administrative network has Web servers that need to be accessed by all of the company and permit only HTTP in. Marketing maintains servers and needs to allow the Telnet, FTP, SMTP, UDP, and HTTP inbound protocols. Similar to the Internet perimeter scenario, only specific protocols that are necessary for each department are allowed. This approach not only limits the number of resources that can be attacked at any one time, but also layers security into administrative areas. The server subnet has only servers that will require Telnet, FTP, and UDP.

- No users will reside on the server subnet
- Administrative subnet will permit HTTP inbound only
- Marketing subnet will permit Telnet, FTP, SMTP, UDP, and HTTP inbound

- Engineering subnet will permit Telnet, FTP, and HTTP inbound
- Everyone on the administrative, marketing, and engineering subnet is permitted all outbound Telnet, SMTP, FTP, H.323, DNS (TCP and UDP), SQLnet, and RealAudio

Corporate Intranet Firewall Sample Configuration

```
FWRouter#sh run
Building configuration...

Current configuration:
!
version 12.0
service timestamps debug uptime
service timestamps log uptime
service password-encryption
!
hostname FWRouter
!
boot system flash
enable secret 5 $1$VZGK$Els0ipcmt8Pe0R98RMEds0
enable password 7 0822455D0A16544541
!
username cisco password 7 121A0C041104
!
!
no ip source-route
no ip finger
!
! inspection is configured per interface. Traffic traversing interfaces depends on
! the configuration of the outbound ACLs.
!
ip inspect audit-trail
! inspection for the admin, eng and mktng subnets. This traffic can go anywhere and
! is applied to traffic heading into the interface. Note: tcp inspection will
! account for TELNET traffic and HTTP which are single channel protocols. UDP
! inspection takes care of DNS.
ip inspect name CorpFIREin tcp
ip inspect name CorpFIREin udp
ip inspect name CorpFIREin ftp
ip inspect name CorpFIREin smtp
ip inspect name CorpFIREin realaudio
ip inspect name CorpFIREin h323
ip audit notify log
ip audit po max-events 100
```

```
!
! e0/0 is the server subnet. Note: no inspection rules are applied due to
! inspection on all the other interfaces. All other traffic accessing this network
! is therefore “secure”.
!
interface Ethernet0/0
 ip address 192.168.10.1 255.255.255.0
 ip access-group 101 in    ! no initiated traffic from the server subnet.
 ip access-group 102 out    ! allows SPECIFIC traffic to the server subnet.
 no ip directed-broadcast
 no ip proxy-arp
 no cdp enable
!
! e0/1 is the administrative subnet.
!
interface Ethernet0/1
 ip address 192.168.20.1 255.255.255.0
 ip access-group 112 out    ! allows web only to the Admin subnet.
 no ip directed-broadcast
 no ip proxy-arp
 ip inspect CorpFIREin in    ! firewall inspection applied to inbound traffic
! according to the Corp. security policy.
 no cdp enable
!
! e1/0 is the engineering subnet.
!
interface Ethernet1/0
 ip address 192.168.30.1 255.255.255.0
 ip access-group 122 out    ! allows SPECIFIC traffic to the Eng subnet.
 no ip directed-broadcast
 no ip proxy-arp
 ip inspect CorpFIREin in    ! firewall inspection applied to inbound traffic
! according to the Corp. security policy.
 no cdp enable
!
! e1/1 is the marketing subnet.
!
interface Ethernet1/1
 ip address 192.168.40.1 255.255.255.0
 ip access-group 132 out    ! allows SPECIFIC traffic to the Mkt subnet.
 no ip directed-broadcast
 no ip proxy-arp
 ip inspect CorpFIREin in    ! firewall inspection applied to inbound traffic
```

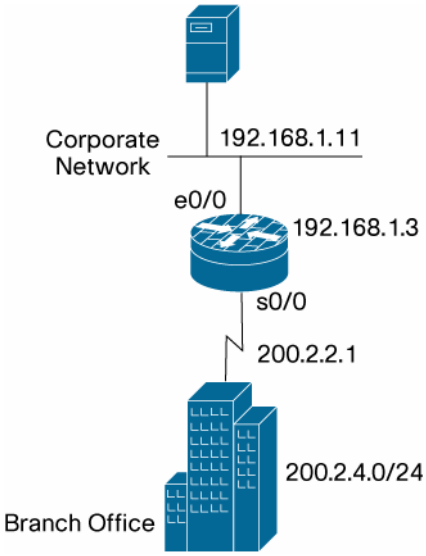
```
! according to the Corp. security policy.
 no cdp enable
!
ip classless
!
logging facility syslog
logging 192.168.1.11
! acl 101 rejects any traffic initiated FROM the server subnet. It’s applied
! to ther Server interface (e0/0) for inbound traffic.
access-list 101 deny ip any any
! acl 102 allows specific traffic to the Server subnet. It’s applied to outbound
! traffic on the server interface. (e0/0)
access-list 102 permit tcp any any eq telnet
access-list 102 permit tcp any any eq ftp
access-list 102 permit udp any any
! acl 112 allows only HTTP traffic to the Admin subnet. (e0/1)
! Note: outbound ACLs 112, 122 and 132 setup implicit denies to the Admin., Eng.,
! and Mkt. interfaces. Firewall inspection inbound on the interfaces permits return
! traffic through the acls. This accounts for telnet, smtp, ftp, h323, dns, sql and
! realaudio for users within the Admin, Eng, and Mkt subnets. (see firewall
! policies)
access-list 112 permit tcp any any eq www
! acl 122 allows only specific traffic to the Eng. subnet. (e1/0)
access-list 122 permit tcp any any eq telnet
access-list 122 permit tcp any any eq ftp
access-list 122 permit tcp any any eq www
! acl 132 allows only specific traffic to the Mkt. Subnet. (e1/1)
access-list 132 permit tcp any any eq telnet
access-list 132 permit tcp any any eq ftp
access-list 132 permit udp any any
access-list 132 permit tcp any any eq www
access-list 132 permit tcp any any eq smtp
no cdp run
!
line con 0
 exec-timeout 0 0
 password 7 14141B180F0B
 login
 transport input none
line aux 0
line vty 0 4
 password 7 045802150C2E
 login
```

```
!
End
```

Branch Office Intranet Perimeter

Cisco IOS Firewall used as a branch office intranet firewall is shown in Figure 19.

Figure 19. Branch Office Intranet Firewall



- Static NAT is used with the private addresses
- Similar to the extranet perimeter firewall, but without the Auth Proxy

Branch Office Intranet Firewall Policy

The protocols allowed on the branch office intranet firewall are Telnet, FTP, and HTTP for both inside and outside traffic. Only the branch office subnet, 200.2.4.0/24, is allowed inside.

- Only the subnet 200.2.4.0/24 is permitted for Telnet, FTP, and HTTP into inside.
- Inside has Telnet, FTP, and HTTP access to the branch office.

The branch office policy limited not only protocols, but also the source address of the network. This limit also restricts the types of attacks that can be generated.

Note the placement of this router relative to the corporate intranet perimeter. This placement, along with the security policy defined per department, helps to limit liabilities throughout the whole network. This builds a layered security approach.

Branch Office Intranet Firewall Sample Configuration

```
FWRouter#sh run
Building configuration...
```

Current configuration:

```
!
```

```
version 12.0
service timestamps debug uptime
service timestamps log uptime
service password-encryption
!
hostname FWRouter
!
boot system flash
enable secret 5 $1$VZGK$Els0ipcmt8Pe0R98RMEds0
enable password 7 0822455D0A16544541
!
username cisco password 7 121A0C041104
!
!
!
!
no ip source-route
no ip finger
!
! Firewall inspection is setup for bi-directionally for traffic to/from the Corp.
! and Branch networks. Note: tcp inspection will account for TELNET traffic and
! HTTP which are single channel protocols.
!
!
ip inspect name BranchFIRE ftp
ip inspect name BranchFIRE tcp
ip audit notify log
ip audit po max-events 100
!
!
! e0/0 is the corporate subnet.
!
interface Ethernet0/0
 ip address 192.168.1.3 255.255.255.0
 ip access-group 101 in    ! allows specific traffic from the corp subnet. Also implicitly
denies unwanted traffic to the branch.
 no ip directed-broadcast
 no ip proxy-arp
 ip nat inside
 ip inspect BranchFIRE in    ! firewall inspection for traffic FROM the corp.
! subnet.
 no cdp enable
!
```

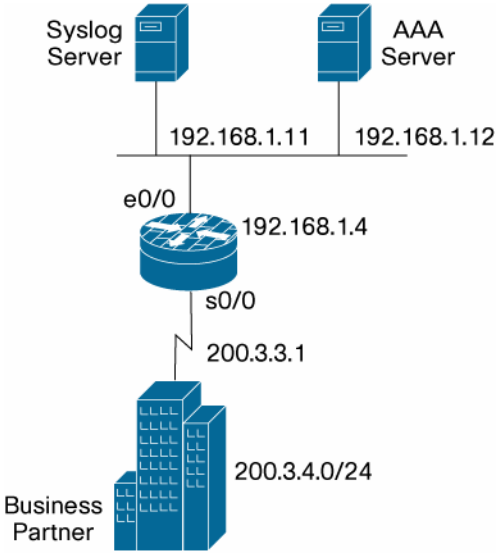
```
! s0/0 is the serial interface to the branch network.
!
interface Serial0/0
 ip address 200.2.2.1 255.255.255.0
 ip access-group 121 in    ! allows specific traffic from the branch office. Also implicitly
denies unwanted traffic to the corp. network.
 no ip directed-broadcast
 no ip proxy-arp
 ip nat outside
 ip inspect BranchFIRE in    ! firewall inspection for traffic from the branch
! office.
 no cdp enable
!
! Inside addresses 192.168.1.41-45 will be translated to 200.2.2.41-45 respectively.
! static nat is used as a simple example. Dynamic nat or a combination can be used
! depending on the address requirements.
!
ip nat inside source static 192.168.1.45 200.2.2.45
ip nat inside source static 192.168.1.44 200.2.2.44
ip nat inside source static 192.168.1.43 200.2.2.43
ip nat inside source static 192.168.1.42 200.2.2.42
ip nat inside source static 192.168.1.41 200.2.2.41
ip classless
!
! syslog is configured for reporting etc.
!
logging facility syslog
logging 192.168.1.11
! acl 101 allows the initial packets sourced from the Corporate subnet. Packets are
! then inspected by the firewall rules. The inspection engine allows for the
! dynamic acl to be built and added to the input acl associated with interface to
! the branch network (s0/0). Implicit denies prevent other unwanted traffic from
! traversing the router.
access-list 101 permit tcp 192.168.1.0 0.0.0.255 any eq telnet
access-list 101 permit tcp 192.168.1.0 0.0.0.255 any eq ftp
access-list 101 permit tcp 192.168.1.0 0.0.0.255 any eq www
! similar to acl 101, acl 121 allows the initial packets sourced from the Branch
! office to be inspected. The firewall engine thus allows return traffic to get
! through acl 101.
access-list 121 permit tcp 200.2.2.0 0.0.0.255 any eq telnet
access-list 121 permit tcp 200.2.2.0 0.0.0.255 any eq ftp
access-list 121 permit tcp 200.2.2.0 0.0.0.255 any eq www
no cdp run
```

```
!
line con 0
 exec-timeout 0 0
 password 7 0822455D0A16
 login
 transport input none
line aux 0
line vty 0 4
 password 7 01100F175804
 login
!
!
end
```

Extranet Perimeter

Cisco IOS Firewall used as a corporate extranet firewall is shown in Figure 20.

Figure 20. Extranet Perimeter Firewall



- Static NAT is used with the private addresses
- Auth Proxy will be used for inside access from business partner

Extranet Firewall Policy

Corporate establishes tighter security control by authenticating the partners before granting them access. Only one subnet, 200.3.4.0/24, is allowed to enter. The partner allows Telnet and FTP to their network.

- Auth Proxy is enabled on serial 0/0 for inbound access
- Only the subnet 200.3.4.0/24 is permitted for Telnet and FTP
- Inside has Telnet and FTP access to the business partner

Much like the branch office, the extranet security policy further limits protocol access. It adds user-level authentication to further verify access to these limited resources. Extranet policies are often the most restrictive parts to a corporate network. Topology placement helps to build the overall corporate security policy, as defined in the corporate Internet perimeter router.

Extranet Firewall Sample Configuration

```
FWRouter#sh run
Building configuration...
```

Current configuration:

```
!
version 12.0
service timestamps debug uptime
service timestamps log uptime
service password-encryption
!
hostname FWRouter
!
boot system flash
!
! Enable authentication proxy globally.
!
aaa new-model
aaa authentication login default group tacacs+
aaa authentication login console_line line none
aaa authentication login vty_line line none
aaa authorization exec vty_line none
aaa authorization auth-proxy default group tacacs+
enable secret 5 $1$VZGK$Els0ipcmt8Pe0R98RMEds0
enable password 7 0822455D0A16544541
!
username cisco password 7 121A0C041104
!
!
!
!
no ip source-route
no ip finger
!
ip inspect name CorpFire ftp
ip inspect name CorpFire tcp
ip auth-proxy name PROXY http
ip audit notify log
```

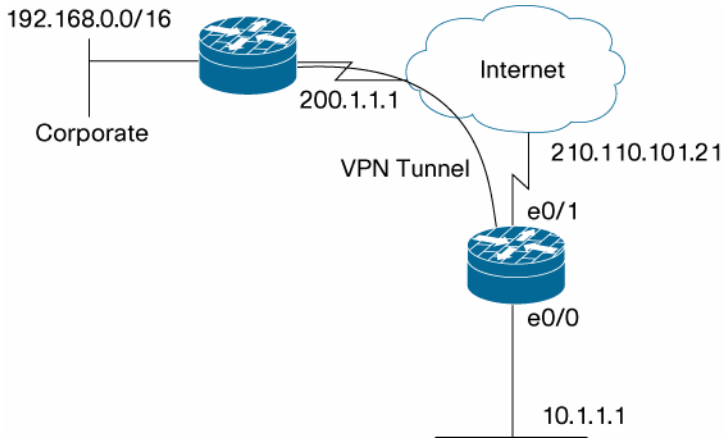
```
ip audit po max-events 100
!
! e0/0 is the inside interface to the corporate subnet.
!
interface Ethernet0/0
 ip address 192.168.1.4 255.255.255.0
 ip access-group 101 in
 no ip directed-broadcast
 no ip proxy-arp
 ip nat inside
 ip inspect CorpFire in
 no cdp enable
!
! s0/0 is the interface to the extranet partner.
!
interface Serial0/0
 ip address 200.3.3.1 255.255.255.0
 ip access-group 111 in
 no ip directed-broadcast
 no ip proxy-arp
 ip nat outside
 ip inspect CorpFire in
 ip auth-proxy PROXY    ! Associates authentication proxy on the interface to the
! extranet partner.
 no cdp enable
!
! Inside addresses 192.168.1.31-35 will be translated to 200.3.3.31-35 respectively.
! static nat is used as a simple example. Dynamic nat or a combination can be used
! depending on the address requirements.
!
ip nat inside source static 192.168.1.35 200.3.3.35
ip nat inside source static 192.168.1.34 200.3.3.34
ip nat inside source static 192.168.1.33 200.3.3.33
ip nat inside source static 192.168.1.32 200.3.3.32
ip nat inside source static 192.168.1.31 200.3.3.31
ip classless
! configures the HTTP server to allow AAA authentication.
ip http server
ip http authentication aaa
ip http access-class 10
!
! sys log is configured for reporting etc.
!
```

```
logging facility syslog
logging 192.168.1.11
access-list 10 deny any
!
! acl 101 blocks all traffic except TACACS+ communications with the AAA server.
! With successful user authentication, Authentication Proxy will override this ACL
! and open connections based on the AAA ACL policy. This ACL is applied to the
! inbound on the inside interface (e0/0)
access-list 101 permit tcp host 192.168.1.12 eq tacacs host 192.168.1.4
access-list 101 permit tcp 192.168.1.0 0.0.0.255 any eq telnet
access-list 101 permit tcp 192.168.1.0 0.0.0.255 any eq ftp
!
! acl 111 denies all initiated traffic from the extranet partner that is not
! authenticated. Note: IOS Firewall’s order of operations checks authentication
! before acls and inspection.!
access-list 111 deny ip any any
no cdp run
! configure the TACACS server’s address and key.
tacacs-server host 192.168.1.12
tacacs-server key key
!
! The following is done in order to secure access to the router via the console
! before enter AAA configuration commands. note: AAA commands take effect before
! saving out of config mode.
!
line con 0
 exec-timeout 0 0
 password 7 0822455D0A16
 login authentication console_line
 transport input none
line aux 0
line vty 0 4
 password 7 01100F175804
 login authentication vty_line
!
!
end
```

Telecommuter/Home Office

Cisco IOS Firewall used as a home office perimeter firewall is shown in Figure 21.

Figure 21. Telecommuter/Home Office Firewall with VPN



- NAT overload/PAT is used with private addresses
- Cisco IOS Firewall is used to protect the home network
- An IPSec LAN-LAN tunnel is configured to the corporate network

Internet Firewall Policy

The telecommuter is granted secure access to the corporate network, using Inspect tunneling. Security to the home network is accomplished through firewall inspection. The protocols that are allowed are TCP, UDP, RTSP, H.323, NetShow, FTP, and SQLnet. There are no servers on the home network, so no traffic is allowed that is initiated from outside. IPSec tunneling secures the connection from the home LAN to the corporate network.

Like the Internet Firewall policy, HTTP need not be specified, as Java blocking is not necessary. Specifying TCP inspection allows for single-channel protocols like Telnet and HTTP. UDP is specified for DNS.

Telecommuter/Home Office Sample Configuration

```
!
hostname telecommuter-fwvpn
!
enable secret 0 xxxx
!
ip subnet-zero
no ip domain lookup
ip domain name telcom.com
ip name-server 210.110.100.1    ! Provides DNS to for the router
ip dhcp excluded-address 10.0.0.1    ! Excludes the router interface in the DHCP pool.
!
ip dhcp pool Client    ! Provides DHCP assignments to all local workstations
 import all
 network 10.0.0.0 255.255.255.0
 default-router 10.0.0.1
```

```
 dns-server 210.110.100.1
 domain-name telecom.com
 !
! Firewall inspection is setup for all tcp and udp traffic as well as specific application
protocols as defined by the security policy.
ip inspect name firewall tcp
ip inspect name firewall udp
ip inspect name firewall rtsp
ip inspect name firewall h323
ip inspect name firewall netshow
ip inspect name firewall ftp
ip inspect name firewall sqlnet
!
crypto isakmp policy 1    ! defines the key association and authentication for ipsec tunnel.
hash md5
authentication pre-share
crypto isakmp key cisco123 address 200.1.1.1
!
!
crypto ipsec transform-set set1 esp-3des esp-md5-hmac ! defines encryption and transform
set for the ipsec tunnel.
!
crypto map to_corporate 1 ipsec-isakmp    ! associates all crypto values and peering address
for the ipsec tunnel.
 set peer 200.1.1.1
 set transform-set set1
 match address 105
!
!!
interface Ethernet0    ! e0 is the internal home network
 ip address 10.1.1.1 255.255.255.0
 ip nat inside
 ip inspect firewall in    ! inspection examines outbound traffic
 no cdp enable
!
interface Ethernet1    ! e1 is the outside or internet exposed interface.
 ip address 210.110.101.21 255.255.255.0
 ip access-group 103 in    ! acl 103 permits ipsec traffic from the corp. router as well as
denies internet initiated traffic inbound.
 ip nat outside
 no cdp enable
 crypto map to_corporate    ! applies the ipsec tunnel to the outside interface.
!
```

```
ip nat inside source list 102 interface Ethernet1 overload    ! utilize nat overload in order
to make best use of the single address provided by the isp.
ip classless
ip route 0.0.0.0 0.0.0.0 210.110.101.1
no ip http server
!
!
! acl 102 associated addresses used for nat.
access-list 102 permit ip 10.1.1.0 0.0.0.255 any
! acl 103 defines traffic allowed from the peer for the ipsec tunnel.
access-list 103 permit udp host 200.1.1.1 any eq isakmp
access-list 103 permit udp host 200.1.1.1 eq isakmp any
access-list 103 permit esp host 200.1.1.1 any
access-list 103 permit icmp any any    ! allow icmp for debugging but should be disabled due
to security implications.
access-list 103 deny ip any any    ! prevents internet initiated traffic inbound.
! acl 105 matches addresses for the ipsec tunnel to/from the corporate network.
access-list 105 permit ip 10.1.1.0 0.0.0.255 192.168.0.0 0.0.255.255
no cdp run
!
line con 0
 exec-timeout 120 0
 stopbits 1
line vty 0 4
 exec-timeout 0 0
 password 0 xxxx
 login local
end
```

GUIDELINES FOR SECURING CISCO IOS FIREWALL

As with all networking devices, controlling access into the firewall is essential. This section highlights some techniques that can secure it. Password and privilege setup is described in “Configuring Passwords and Privileges”. Additionally, AAA can be used to set up user authentication, authorization, and accounting (see Cisco documentation for more information).

Access

- Keep the firewall in a secured (locked) room.
- Think about access control *before* connecting a console port to the network in any way, including attaching a modem to the port. Be aware that a *break character* on the console port might give total control of the firewall, even with access control configured.

- Apply access lists and password protection to all virtual terminal ports. Use access lists to restrict network addresses that are allowed to Telnet into the router.
  - When setting passwords for privileged access to the firewall, use the **enable secret** command rather than the **enable password** command, which does not have as strong an encryption algorithm.
  - Put a password on the console port. In AAA environments, use the same authentication for the console as for elsewhere. In a non-AAA environment, at a minimum configure the **login** and **password** commands.
- Configure the **no ip proxy-arp** command to prevent internal addresses from being revealed. (This is important to do if you do not already have NAT configured to prevent internal addresses from being revealed.)

Services

Any enabled service could present a potential security risk. A determined, hostile party might be able to find creative ways to misuse the enabled services to access the firewall or the network. Do not enable any local service (such as SNMP or NTP) that will not be used.

- Cisco Discovery Protocol (CDP) is enabled by default. To turn off CDP, enter the **no cdp run** global configuration command, or apply **no cdp enable** on public or vulnerable interfaces if CDP is required for specific interfaces.
- For local services that are enabled, protect against misuse by configuring the services to communicate only with specific peers, and protect by configuring access lists to deny packets for the services at specific interfaces.
- Disable minor services. For IP, enter the **no service tcp-small-servers** and **no service udp-small-servers** global configuration commands.
- If NTP is essential, configure it only on required interfaces to listen only to certain peers, and use authenticated NTP if NTP poisoning is a concern. To disable the NTP server on specific interfaces, enter the **ntp disable** interface configuration command on each interface where NTP will not be served.
- Disable all HTTP services except those required for network function. Service restriction is configured with “ip http active-session-modules” command, as documented on Cisco.com: http://www.cisco.com/en/US/products/ps6350/products_configuration_guide_chapter09186a0080455929.html

Cisco IOS offers two different options to “lock down” routers. AutoSecure is available at the command line by typing “auto secure” at the router’s enable prompt, and Cisco Router and Security Device Manager (SDM) offers the Router Security Audit tool in a GUI format. Both of these tools can lead the network engineer through a dialogue to disable services that may increase the router’s vulnerability profile, set up stronger authentication, and configure other settings to improve router security.

Routing and Spoofing

Cisco IOS Software includes many features to prevent malicious application of various IP features. These features can protect the router and the network against activity that is trying to exploit an alternative route around devices where network policy may be applied, or to:

- Protect against spoofing: protect the networks on both sides of the firewall from being spoofed from the other side. You could protect against spoofing by configuring input access lists at all interfaces to pass only traffic from expected source addresses, and to deny all other traffic. Many network attacks (SYN attacks in particular) use a “bogus” source address, originating from one of the reserved address ranges. The reserved address space list is somewhat dynamic, as the Internet Assigned Numbers Authority grants and revokes address space to Internet network service providers. The list of reserved or unassigned IP addresses is known as the “bogon” list. A current bogon list is available at: http://www.cymru.org
- IP Source Routing allows a traffic source to specify that a packet must pass through certain hosts on the Internet on the way to its destination. Source routing rarely has legitimate use on the public Internet, and is more frequently used for network attack or abuse. Under most circumstances, all routers in a network should have source routing disabled. Disable source routing. For IP, enter the **no ip source-route** global configuration command.

Directed Broadcast

- Directed broadcasts are rarely required by IP networks. Directed broadcasts should be disabled for all applicable protocols on your firewall and on all your other routers. To disable it for IP, use the **no ip directed-broadcast** command.
- Directed broadcasts can be misused to multiply the power of DoS attacks, because every DoS packet sent is broadcast to every host on a subnet. Furthermore, some hosts have other intrinsic security risks present when handling broadcasts.

CONFIGURING AND MANAGING THE CISCO IOS FIREWALL

Configuration

Most small to medium-sized networks can employ *element-based* configuration and management, where a network engineer/administrator individually configures and monitors devices on the network. Element-based configuration and monitoring/management is accomplished with the router's CLI, or with Cisco SDM if an easier-to-use graphical interface is desired. The CLI and Cisco SDM can only configure one device at a time, and do not provide any multiple-device correlation to compile networkwide rules' impact on traffic.

As the number of devices on the network increases, determination of the *network's* policy for a given traffic flow becomes more complex. Larger networks usually benefit from application of *network-based* configuration and management systems, such as Cisco Security Manager to integrate networkwide provisioning, configuration, and management into one system, and Cisco Security Monitoring, Analysis, and Response System (MARS) for network monitoring and security event correlation.

Management and Monitoring

Cisco IOS Firewall monitoring is accomplished by syslog messages for alarms, and CLI interaction to view statistics of firewall activity. Consult product documentation for relevant "show" and "debug" commands for monitoring and troubleshooting IOS Firewall activity.

REFERENCES

- Cisco IOS Firewall Overview
- Security Command Reference
- Cisco IOS Firewall Command Reference
- Implementing Authentication
- SDM 2.2 User's Guide , including several pages for configuring IOS Firewall with SDM

Corporate Headquarters
Cisco Systems, Inc.
170 West Tasman Drive
San Jose, CA 95134-1706
USA
www.cisco.com
Tel: 408 526-4000
800 553-NETS (6387)
Fax: 408 526-4100

European Headquarters
Cisco Systems International BV
Haarlerbergpark
Haarlerbergweg 13-19
1101 CH Amsterdam
The Netherlands
www-europe.cisco.com
Tel: 31 0 20 357 1000
Fax: 31 0 20 357 1100

Americas Headquarters
Cisco Systems, Inc.
170 West Tasman Drive
San Jose, CA 95134-1706
USA
www.cisco.com
Tel: 408 526-7660
Fax: 408 527-0883

Asia Pacific Headquarters
Cisco Systems, Inc.
168 Robinson Road
#28-01 Capital Tower
Singapore 068912
www.cisco.com
Tel: +65 6317 7777
Fax: +65 6317 7799

Cisco Systems has more than 200 offices in the following countries and regions. Addresses, phone numbers, and fax numbers are listed on **the Cisco Website at www.cisco.com/go/offices.**

Argentina • Australia • Austria • Belgium • Brazil • Bulgaria • Canada • Chile • China PRC • Colombia • Costa Rica • Croatia • Cyprus Czech Republic • Denmark • Dubai, UAE • Finland • France • Germany • Greece • Hong Kong SAR • Hungary • India • Indonesia • Ireland • Israel Italy • Japan • Korea • Luxembourg • Malaysia • Mexico • The Netherlands • New Zealand • Norway • Peru • Philippines • Poland • Portugal Puerto Rico • Romania • Russia • Saudi Arabia • Scotland • Singapore • Slovakia • Slovenia • South Africa • Spain • Sweden • Switzerland • Taiwan Thailand • Turkey • Ukraine • United Kingdom • United States • Venezuela • Vietnam • Zimbabwe

205233.BP_ETMG_KS_11.05

Printed in the USA