Global Information Assurance Certification Paper

Mark Degner
GSEC Practical v1.3

Securing Your Network With An Internet Access Router
(or Getting Your Money's Worth From Your Cisco Gear)

Introduction

When designing Internet defenses, administrators and security practitioners frequently overlook easy, inexpensive means of providing "Defense in Depth." One of these methods is taking advantage of the security features built into the Cisco Internetwork Operating System (IOS). Cisco routers are quite prevalent in the market today, and many organizations use them for Internet access. Although the author's familiarity with Cisco equipment will be the focus of this paper, the same techniques can also be applied to equipment from other vendors.

William Cheswick and Steven Bellovin covered the idea of this cost effective technique in Firewalls and Internet Security:

> Packet filters can provide a cheap and useful level of gateway security. Used by themselves, they are cheap: the filtering abilities come with the router software. Since you probably need a router to connect to the Internet in the first place, there is no extra charge (Cheswick and Bellovin, p. 54-55).

A basic secure router configuration is easy to apply, and should become standard practice when deploying any new router. Routers make up the backbone of increasingly critical corporate networks, so it is imperative that they always operate the way we expect.

In this document, we will cover the configurations that should be applied to nearly any Cisco router, and routers deployed for Internet access in particular. First, we will look at how to secure the Internet border router from malicious attack. Then, we will discuss how we can utilize the router to further protect the internal network. A sample configuration will also be an appendix at the end of this document to provide an easily implemented change.

Conventions used in the text of this document:
The body of the paper will be in 12 point Times New Roman.
`Configuration commands will use the 10 point Courier New font.`
`Router output will be displayed in italicized 10 point Courier New font.`
The interface on network devices referred to as 'inside' is the interface that is facing the network we are protecting. Interfaces defined as 'outside' are those interfaces facing the Internet.

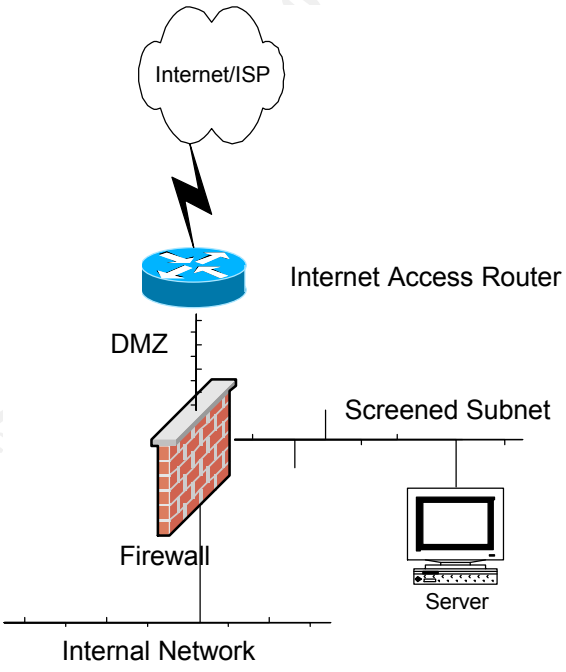
Basic Design Concepts

Security designs almost always include some sort of firewall protecting the organization's network from the Internet. In general, a properly configured, modern firewall will provide much more security than a simple filtering router. However, the filtering ability of the router can add a significant amount of security to the network when coupled with the firewall.

This design is described clearly in the following excerpt:

> The exterior screening router acts as a first-level filter to permit or deny traffic coming in from the Internet to the internal campus. It validates most incoming traffic before passing it on to the firewall. The firewall then provides the more CPU-intensive function of packet-by-packet inspection (Kaeo, p. 258).

We can also see this common design illustrated in the next diagram.



Most often, the connection to the ISP is some sort of serial line, but it can also be a high-speed line such as Ethernet or ATM. We will call this the outside interface of the router.

The inside interface of the router is most often an Ethernet interface which is connected to the same segment as the outside interface of the firewall. This segment is referred to as the DMZ, in reference to the military function of the same name. The screened subnet is where services accessed from the public Internet are located, in order to isolate them from the internal network. The internal network is where users and services internal to the organization are located.

An Intrusion Detection System may also be located on the DMZ network to monitor Internet traffic.

Securing the Router

In order for a router to be effective at providing additional network security, the integrity of the configuration must first be guaranteed by hardening the router itself against attack. This hardening includes limiting who has access to the router configuration, and making the router resistant to network-based exploits. If an attacker cannot get access to the router console, or if it is secured against remote attacks, then he or she cannot make changes to the router configuration.

Reducing the services running on the router will also aid in further securing the device. The Cisco IOS is really just a specialized operating system, and like any other operating system, running services may be susceptible to exploits. These vulnerabilities can be another vector for an attacker to gain control of the router, or make unauthorized changes to the configuration.

First, password encryption must be enabled to obscure passwords from casual onlookers. Be aware that this is not a strong encryption scheme, and the passwords should still be kept confidential, even in their encrypted form. If the configuration is copied to a file, the passwords should be scrubbed from the text.

```
service password-encryption
```

Best practice is to configure the privileged mode (often called enable mode) password as 'secret.' This stores the password in stronger encryption than if it was just configured as 'enable password.' This will require extra care in the event the password is lost, and the administrator must utilize the password recovery procedure. The password cannot be recovered, but a new password can be created.

```
enable secret <password>
no enable password
```

For the same legal reasons that banners are recommended for computer systems, a

banner is used at the login prompt on routers. Using the 'banner login' command causes the banner to be displayed before the user logs in, rather than after.

```
banner login #
This is a private system.

Use by unauthorized persons is prohibited.

All accesses to this service are logged.
#
```

Take care to not reveal too much information about the system or organization in the banner. A simple statement such as that above will suffice.

Limiting Access

There are a few things a network administrator can do to limit access to the router console. The first, and easiest thing to do is limit access by the source IP address. Another way to limit access is to enforce users to use unique login names, the same way that system administrators should not always log in to their systems using the superuser account.

By IP address

The administrators accessing the router are usually a small, known group. This also means the IP addresses they operate from are generally known. The telnet service on the router must be limited to as few addresses as possible. These may be the addresses of the administrator's workstations, a Network Address Translation address on the firewall, or in special cases, a remote management address.

By configuring this small, known set of addresses in an access list and tying it to the virtual terminal lines, we have a lot of control who can telnet to the router. We see an example below. Only addresses listed in access-list 99 will be permitted to telnet to the router, and the accesses will be logged. All other traffic will be explicitly denied, and all unauthorized access attempts will also be logged.

```
access-list 99 permit <admin IP address> log
access-list 99 deny any log
```

Access-list 99 is then applied to the VTY lines to limit access with the 'access-class in' command. Only the host address specified in the access list will be permitted to telnet to the router.

```
line vty 0 4
 access-class 99 in
```

```
exec-timeout 15 0
password <choose a password>
login
transport input telnet (or ssh if used)
```

There are a couple other configuration commands we have added here. The 'exec-timeout' command disconnects the VTY session in 15 minutes and 0 seconds, so sessions are not inadvertently left open for long periods of time. The 'login' command forces the user to authenticate when they telnet to the router, using the password assigned. 'Transport input' limits the protocols that are allowed to access the virtual terminal. Limit this to telnet, refusing all other protocols. If SSH is used on the router, allow that here, as well.

Extended access lists can also be used to log additional information about the session. The 'log-input' command will record the input interface and source MAC address of the session in the log.

```
      access-list 100 permit tcp host <source address> host 0.0.0.0
eq 23 log-input
      access-list 100 deny ip any any log-input
```

Only traffic from the configured source address is permitted through this access list. All other traffic is denied, and all traffic is logged. This access list is then applied in the same manner as the standard access list.

A note about VTY lines, it is possible to configure different access privileges on any of the VTY lines on the router. In general, the configuration for all the VTY lines will be the same, such as we saw above. The access lists we configured above are applied to VTY lines 0 through 4. Sometimes it is useful to give different access to one or more of the VTY lines. For instance, in an organization where many people have access to the router, the senior administrator may reserve a VTY line for his or herself, in the event the other VTYs are being used. This would just be a matter of writing a new access list, and only configuring it on the last VTY line; line 4.

By user

Cisco routers also have the capability of controlling access by login names and passwords. Most configurations utilize the built in user level and priveleged level access controls. While this works at controlling access to the router console, it does not provide for any accountability. Requiring administrators to use login names makes it easier to track who made changes to the configuration and when, and to determine if a username/password pair has been compromised.

The easiest way to implement user authentication is to explicitly configure the users on the router itself. This is done with the following command:

```
username <user> password <password>
```

Once a username is configured on the router, the router will then prompt the administrator for their username, before requesting the password.

For more advanced user management, the router can use Authentication, Authorization and Accounting (AAA) systems. The user database can be stored at one central location, and network devices can then query the AAA server, instead of having users explicitly configured on each network device. With an Internet router on the other side of the firewall, make sure to permit the authentication traffic in the firewall security policy.

AAA requires a remote user database, running an authentication protocol, often something like RADIUS, or TACACS. A sample configuration is shown below. This configuration uses a TACACS+ server.

This first command enables the AAA system on the router.

```
aaa new-model
```

The next command utilizes the TACACS+ server as the primary authentication method, and if it is not available, uses case-sensitive local authentication information. This would makes use of the username/password configuration that was discussed above.

```
aaa authentication login default group tacacs+ local-case
```

To ensure the privilege mode access is also controlled by the AAA system, the next configuration command is entered. The group keyword instructs the AAA feature to utilize the servers listed with the 'tacacs-server' host command.

```
aaa authentication enable default group tacacs+ enable
```

The AAA server will authorize privileged configuration commands, before the user is allowed to execute them.

```
aaa authorization commands 15 default group tacacs+ local
```

The following two commands will log when the user exits user mode and privileged mode respectively.

```
aaa accounting exec default stop-only group tacacs+
aaa accounting commands 15 default stop-only group tacacs+
```

To log all network related service requests, the following line would be used.

```
aaa accounting network default stop-only group tacacs+
```

The two remaining AAA commands log any system events not explicitly related to users, and any connections initiated from the router itself. This latter command can be useful in determining if the router was used as a man-in-the-middle for a malicious attack.

```
aaa accounting system
aaa accounting connection
```

Finally the IP addresses of the AAA servers are listed, and the encryption key used to secure the AAA data.

```
tacacs-server host <ip address>
tacacs-server key <shared secret key>
```

Although more complex than the typical user level and privilege level passwords, AAA systems give administrators and security professionals much more control and information in regards to the users that are logging into the router.

Securing the Router Physically

Physical security is just as important as network security in ensuring the integrity of the router. The router should be in a secured room, limiting access to the console port. With physical access, an attacker can make configuration changes to the router without using the configure authentication mechanisms on the router. The router can be power cycled, and the configuration can be overridden using the password recovery procedures for that particular model of router.

To deter the less sophisticated hacker, or casual adventurer, a password should also be configured on the console port to prevent unauthorized access. If it is not being used, the Auxiliary port must be disabled. An administrator would disable this port like this:

```
line aux 0
 no exec
 exec-timeout 0 10
 transport input none
```

Hardening the Router Against Exploits

Early versions of the Cisco IOS had many common network services enabled by default. More recent versions have disabled many of these services by default, in order to provide extra security, and conserve resources. In this section we will discuss these services

As specialized as the Cisco IOS is, it is still not immune to programming errors and security exploits. Here is a short list of security issues that have been found in Cisco routers in the last couple years:

Cisco IOS HTTP Server vulnerability - Cisco bug ID CSCdr36952
Cisco IOS Software TELNET Option handling vulnerability - Cisco Bug ID CSCdm70743
Cisco IOS Data Leak with Cisco Express Forwarding Enabled – Cisco Bug ID CSCdu20643
Cisco IOS Malformed SNMP Message-Handling Vulnerabilities – Multiple Bug IDs
Cisco IOS ARP Table Overwrite Vulnerability - Cisco Bug ID CSCdu81936
Cisco IOS Software Multiple SNMP Community String Vulnerabilities – Multiple Bug IDs

You can see from this list that Cisco routers are as susceptible to vulnerabilities as other network devices, and that it is important to limit the running services, and the access to those services, to protect against known and unknown exploits.

There are several services, analogous to those running on computer systems that should be turned off. They either do not provide any real functionality, or do not provide enough functionality to warrant the risk. Depending on the software version, some services may not be available, and there is variation as to whether they are enabled or disabled by default.

Next are listed several of the services offered by the IOS and a quick explanation as to what they are, and the reasoning for disabling them.

The finger service is one of these services that should be turned off; to hide the identity of the user logged in, and to shut down an unneeded service that could potentially create vulnerability.

```
no service finger
```

The bootp server is not needed for Internet routers, and is disabled to avoid possible vulnerability exploits.

```
no ip bootp server
```

Since the Cisco IOS HTTP server does not provide any extra functionality, it has been disabled to prevent remote access to the router via the HTML interface, and to protect the router from HTTP vulnerabilities.

```
no ip http server
```

UDP and TCP small servers, which provide services such as CHARGEN, have no real purpose on a router, and must be disabled.

```
no service udp-small-servers
no service tcp-small-servers
```

Disabling the config service prevents malicious configuration from being fed to the router via a TFTP server. If the service is enabled, the router will look on the network for a configuration, before booting the local configuration.

```
no service config
```

PAD is only needed with the X.25 protocol, and since this router is in an IP only configuration, we will not leave this service running.

```
no service pad
```

Although it has some uses in certain environments, there is no need to use the DHCP server in this design, so we will disable it.

```
no service dhcp
```

Ensure that the TFTP server is deactivated on the router.

```
no tftp-server
```

There is also no need to run the IDENTD service.

```
no ip identd
```

As you can see, quite a few services are available in the Cisco IOS, none of which we need in this design. Disabling them will only add to the security and stability of our Internet access router.

Protecting Internal Hosts

Now that the device has been made more secure, we will now focus on utilizing IOS features to protect the rest of the network. This involves a combination of commands entered in global configuration mode, and interface configuration mode. First we will cover those commands that are configured system wide on the router, then the interface specific commands will be discussed.

The no ip source-route command prevents the router from forwarding packets with source-routing information. Source routed IP packets are rarely used in practice, and is more often used as a vulnerability exploit.

```
no ip source-route
```

CDP is a Cisco proprietary protocol that is used for discovering other devices on the same segment. Since CDP tends to reveal more information than we want to make available, and there really isn't any reason to discover devices on the outside of the network, it should be disabled on our Internet router. CDP is disabled system wide with this command in global configuration mode:

```
no cdp run
```

The inside Ethernet interface of our access router is configured with additional commands.

```
interface Ethernet0/0
 description Inside Interface to DMZ
```

No ip redirects prevents the router from passing redirect messages, so routing decisions can not be altered.

```
 no ip redirects
```

No ip unreachables prevents the transmission of icmp unreachable packets, which may be used by an attacker to probe the access list or firewall configuration

```
 no ip unreachables
```

No ip directed-broadcast prevents the network from being used in a 'smurf' attack, which takes advantage of the tendency of all devices on a network to respond to a ping to the network broadcast address.

```
 no ip directed-broadcast
```

No ip proxy arp is disabled to prevent the flow of packets from being altered, similar to ip redirects. This sometimes exposes problems with the design of a network when disabled. If end stations are not configured with the proper default gateway for their network, proxy arps will still allow the packets to be routed. When disabled, network problems may occur, so the end stations should be checked for proper configuration.

```
 no ip proxy-arp
```

The mroute-cache command is used for enhancing the performance of multicast traffic. Our configuration does not provide for multicast traffic, so this command is disabled.

```
 no ip mroute-cache
```

Since CDP is unneeded in this configuration, it is disabled, since information about the router can be discovered using the CDP protocol.

```
no cdp enable

no ip mask-reply

ip accounting access-violations
```

TCP Intercept

For versions of IOS software that support it, TCP Intercept can be enabled to help protect against SYN flood attacks.

A SYN flood attack is defined in this way in the Cisco documentation:

> A SYN-flooding attack occurs when a hacker floods a server with a barrage of requests for connection. Because these messages have unreachable return addresses, the connections cannot be established. The resulting volume of unresolved open connections eventually overwhelms the server and can cause it to deny service to valid requests, thereby preventing legitimate users from connecting to a web site, accessing e-mail, using FTP service, and so on (Cisco IOS 12.1 Documentation).

This feature of the IOS resists SYN floods by validating TCP connection requests. During a TCP connection, the router will first complete the TCP three-way handshake with the requesting system, then complete the handshake with the destination server, and tie the two connections together transparently. The router will then protect the server from a barrage of half-open connections, leaving it available to service legitimate requests.

These configurations and values are those that are recommended by Rob Thomas in his document "Secure IOS Template Version 2.3."

```
access-list 120 permit tcp any <screened subnet> <screened
subnet wildcard mask>

ip tcp intercept list 120
ip tcp intercept connection-timeout 60
ip tcp intercept watch-timeout 10
ip tcp intercept one-minute low 1500
ip tcp intercept one-minute high 6000
```

The access list defines which services are being protected by the TCP Intercept feature. The first 'tcp intercept' statement then applies the access list to the feature. The

remainder of the commands then monitor the connection for 60 seconds, shut down half-open connections after 10 seconds, and set a low threshold of 1500 open connections pre minute and a high threshold of 6000 connections per minute respectively.

Depending on the amount of traffic service by the organization, these thresholds most likely will need to be tuned to be effective.

Anti-spoofing

The next step taken in the configuration is having the router shun spoofed traffic. Spoofing is the act of crafting a packet in such a way that it appears to have come from an address other than the host that actually transmitted the packet. It is not possible to prevent all spoofed traffic, since someone may spoof a valid host, but we will attempt to filter out as much as we can.

Configuring anti-spoofing involves filtering spoofed traffic on ingress (inbound) and egress (outbound).

Ingress Filtering

The following access list entries filter out traffic that enters the serial interface, which is sourced from illegal addresses, and violations are logged. This extended access list is named 'inbound-filter' to describe the direction the filtered traffic is flowing with regards to the organization's network.

```
ip access-list extended inbound-filter
```

Our network should not be the source address of inbound traffic.

```
deny   ip <local networks> <local networks wildcard> any log
```

Drop any packets sourced from network 0.

```
deny   ip 0.0.0.0 0.255.255.255 any log
```

Drop packets sourced from an address of all 1s.

```
deny   ip host 255.255.255.255 any log
```

Deny traffic sourced from the loopback network.

```
deny   ip 127.0.0.0 0.255.255.255 any log
```

Unless necessary, drop multicast traffic.

```
deny   ip 224.0.0.0 15.255.255.255 any log
```

No Class E addresses should be seen.

```
deny   ip 240.0.0.0 7.255.255.255 any log
```

Drop traffic sourced from RFC 1918 addresses, as these are not routable on the Internet.

```
deny   ip 10.0.0.0 0.255.255.255 any log
deny   ip 172.16.0.0 0.15.255.255 any log
deny   ip 192.168.0.0 0.0.255.255 any log
```

Deny traffic from the test-net network.

```
deny   ip 192.0.2.0 0.0.0.255 any log
```

169.254/16 is reserved as the 'end node autoconfig' network, and isn't present on the Internet.

```
deny   ip 169.254.0.0 0.0.255.255 any log
```

Deny traffic from a host with address of all 0s.

```
deny ip host 0.0.0.0 any log
```

Permit traffic that is destined for the organization's networks.

```
permit ip any <local networks> <local networks wildcard>
```

Drop all other traffic that doesn't match these filters, and log it.

```
deny udp any range 1 65535 any log
deny tcp any range 1 65535 any log
deny ip any any log
```

The ingress filter would then be applied to the outside interface, in the inbound direction:

```
ip access-group inbound-filter in
```

And outbound on the inside interface:

```
ip access-group inbound-filter out
```

Egress Filtering

A responsible Internet netizen will attempt to prevent its network from being used as the source of an attack, whenever possible, as much as it would to protect against being the target of an attack. Some outbound traffic filtering can be done to ensure only valid addresses are being transmitted from the local network to the Internet.

```
ip access-list extended outbound-filter
```

The router will accept IP traffic sourced only from the local network, since all other address that reach the Ethernet interface of the router are spoofed, or not translated properly.

```
permit ip <local networks> <local networks wildcard> any
 deny udp any range 1 65535 any log
 deny tcp any range 1 65535 any log
 deny ip any any log
```

Once configured, this access list would then be applied to the inside interface in the inbound direction:

```
ip access-group outbound-filter in
```

and the outside interface in the outbound direction:

```
ip access-group outbound-filter out
```

ICMP

Some Internet Control Message Protocol (ICMP) traffic is used for exploiting systems, so only allow ICMP traffic that is vital to the operation of IP, or necessary for troubleshooting connectivity is permitted through the router. The ICMP types allowed will allow for MTU messages to be sent and received, and will allow the 'ping' and 'traceroute' commands to operate. All other ICMP traffic is then blocked and logged. These statements would be added to the ingress and egress filters that were built for anti-spoofing.

Ingress

```
 permit icmp any <local networks> <local networks wildcard>
packet-too-big
 permit icmp any <local networks> <local networks wildcard>
echo-reply
 permit icmp any <local networks> <local networks wildcard>
echo
 permit icmp any <local networks> <local networks wildcard> ttl-
exceeded
 deny   icmp any any
```

Egress

```
 permit icmp <local networks> <local networks wildcard> any
packet-too-big
 permit icmp <local networks> <local networks wildcard> any
echo
 permit icmp <local networks> <local networks wildcard> any
echo-reply
 permit icmp <local networks> <local networks wildcard> any ttl-
exceeded
deny icmp any any
```

Depending on the organization's architecture, other ICMP message type may be needed, so be aware of this when the ICMP filters are constructed.

SNMP

If SNMP is desired on the router, permit SNMP access only from those hosts that require it. Recent SNMP exploits have underscored the need to limit access to SNMP services as much as possible.

In the commands listed below, an access list is created that only allows an explicitly defined host to access the SNMP service on the router. This access is read-only, and any SNMP traps sent from the router will be authenticated with a configured authentication string.

```
access-list 90 permit <SNMP host> log
access-list 90 deny any log

snmp-server community <string> RO 90
snmp-server trap-authentication
snmp-server host <snmp ip> <authentication string>
```

If SNMP access is not required, disable the SNMP server with the following command:

```
no snmp-server
```

Make sure not to use the default communities of public and private on the system. They are too well known, and should be removed from the configuration before a new read-only community string is chosen. Read-write access should not be enabled at all, to prevent unauthorized reconfigurations.

Extending Security Policy from the Firewall to the Internet Access Router

Apart from the filtering we are already doing, the router can also be configured to be an extension of the organization's security policy, mimicking the filtering rules on the firewall. This requires extra management in keeping the filters synchronized, but can have

the benefit of reducing the load on the firewall, which is inspecting more information in the packet than the router, and by reducing much of the noise seen by an IDS sensor on the DMZ.

A sample access list is shown below:

```
ip access-list extended firewall-filter-in
permit tcp any host <web server address> eq www
permit tcp any host <mail server address> eq smtp
```

We can also further secure the public servers by ensuring that any traffic generated from them is in the response to an already established TCP connection using the 'established' keyword in an access list.

```
ip access-list extended firewall-filter-out
permit tcp host <web server address> eq www any established
permit tcp host <mail server address> eq smtp any established
```

This access list would then be applied on the inside interface of the router, in the inbound direction.

Although this may seem to be useless redundancy, it can actually be quite beneficial to the organization. In the even that one of the security devices fails, or is exploited, the other device will still be applying the organization's security policy to traffic. It also prevents unwanted traffic due to misconfiguration, since the same configuration must be applied to both devices.

Logging

Logging is quite helpful in monitoring the health of network devices, and for monitoring malignant and benign network traffic. Close examination of the logs will tell the administrator if their configuration is working as expected, and if it should be modified to provide for more security, or more access.

The first step is to turn on the logging buffer to trace problems and violations of the access-lists. This log is on the router itself, so when at the console, we can review the buffered messages that have been written to it. Logging to the console should be disabled for performance reasons until it is necessary to enable it for troubleshooting.

```
logging buffered 32768 informational
no logging console
logging trap debugging
logging facility local7
logging <syslog server IP address>
```

The configuration above also makes mention of a Syslog server. The router can send messages to a logging server to store all of the messages in one location. By default, the router uses Syslog facility 7, but this can be changed by modifying the configuration.

The timestamps service was configured to save all log entries in local-time, to make it easier to trace logs. This ties in with the operation of NTP, which will be discussed next, to ensure the synchronization of time and logs. Both normal logging and debug information will have timestamps which will help us determine when the event happened.

```
service timestamps log datetime msec localtime show-timezone
service timestamps debug datetime msec localtime show-timezone
```

Setting the timezone of all devices as GMT make correlating logs less confusing and easier.

```
clock timezone GMT 0
```

NTP

Network Time Protocol is useful for ensuring the router has the correct time when logging information to the buffer, or a Syslog server. Some models of routers, such as the Cisco 2500, do not have a realtime clock, so they cannot store the time information themselves. They must either have it manually configured each time the router is rebooted, or must synchronize their time with a network source. Make sure you use trusted, reliable NTP sources. The best sources are those run by the organization, using some other time synchronization method, such as radio, or GPS. The NTP configuration can then be set up in such a way that authentication can be used between NTP devices.

```
ntp server <server IP address> key 1
ntp authenticate
ntp authentication-key <key number> md5 <string>
ntp trusted-key 1
ntp update-calendar
```

Other Configurations

In addition to everything we have covered so far, a few other recommended commands aid in the management of the router.

The TCP keepalives service is enabled to end disconnected telnet sessions to prevent them from using resources, and tying up the terminal lines.

```
service tcp-keepalives-in
```

IP domain-lookup can be turned off as a convenience to prevent the console from being tied up doing a DNS lookup when a command is entered incorrectly.

```
no ip domain-lookup
```

Since the Internet no longer uses the concept of 'classess', other functionality can be enabled by the router.

The Zero subnet can now be used, and can be enabled on the router in global configuration mode.

```
ip subnet-zero
```

And classless routing is enabled to prevent any problems with default routing.

```
ip classless
```

IDS Issues

As any security professional knows, Intrusion Detection System (IDS) sensor placement is crucial to its effectiveness. When securing your network on the access router as we are discussing in this document, certain aspects of your IDS implementation are going to be affected. By filtering out certain traffic and protocols before it reaches your firewall, some malicious traffic is not going to be seen by your Intrusion Detection System, hampering your ability to identify probes and attacks.

The following statement emphasizes the impact on IDS that a filtering router can have:

> Intrusion detection sensors are usually placed outside the firewall in the DMZ. The DMZ, or Demilitarized Zone, is the area between an ISP and the outermost Firewall interface. This arrangement allows the sensor to see all attacks coming in from the Internet. However, if the attack is TCP, and the firewall or filtering router blocks the attack, the intrusion detection system might not be able to detect the attack. Many attacks can only be detected by matching a string signature. The string will not be sent unless the TCP three-way handshake is completed (Northcutt, p. 42)

The advantages and disadvantages should be weighed against each other, however, before a decision is made whether or not to filter traffic at the access router. Although our IDS is not going to see some of the malicious traffic now blocked by our filters and access lists, this also means that this malicious traffic is not making it to your network. In addition, the amount of traffic the IDS sensor must process is now reduced, increasing its ability to

monitor the traffic that does make it into the network.

This will make the job of the IDS administrator easier, as the number of false alarms will be substantially reduced for those services that are not offered by the organization. With the reduction in traffic, the IDS can be tuned more effectively, analyzing only the traffic that makes it through the access router, before it gets through the more sophisticated firewall.

Performance Concerns

Filtering traffic with a router requires that router to process more of the traffic, making it unable to utilize some of the performance enhancing features on a Cisco router. For this reason, it is more beneficial to be conscious of the order in which an access list entries are created. A Cisco IOS access list is processed top down, and first match, so the entries corresponding to the greatest percentage of traffic should be at the top of the access list. This will allow the router to identify and process the traffic more quickly, minimizing the impact of the access list on network performance.

Once the access list is created, it is possible to monitor the number of hits each line of the access list receives, so tuning can be done. Following is an example of how to determine if the order of the access list entries is correct.

```
Router#sho access-lists 100
Extended IP access list 100
    permit tcp 192.168.1.0 0.0.0.255 any eq www (12 matches)
    permit tcp 192.168.1.0 0.0.0.255 any eq telnet (129
matches)
```

We can see from this example that the second access list entry permitting telnet traffic has more hits than the first entry permitting HTTP traffic. This access list would be more effective if the position of each line were switched. The router would then have to process less of the access list, making it more efficient.

Conclusion

In this document we have covered a lot of compelling reasons to secure Internet routers, and using them to secure the rest of your network. Once an administrator understands the reasoning for some of these design choices, the choice of applying more security to a router becomes a simple one. The configurations aren't that difficult, and there are plenty of resources out there to help someone develop their own secure configurations.

In an ideal world, many of these security filters would be applied on the ISP end of our Internet link, to prevent bandwidth from being wasted by undesired traffic, but until then,

we can at least prevent this traffic from leaking into our network.

We have touched on many security issues in this text, but the security of the equipment can be taken further with more specialized configurations. Some of these more involved, model specific configurations are discussed in the references used for this text. The reader is encouraged to read some of these references. The topics and configurations discussed herein are drawn from several different sources available on the Internet, and the reader is encouraged to read the texts listed in the reference.

Appendix- Sample Internet Access Router Configuration

```
service timestamps debug datetime msec localtime show-timezone
service timestamps log datetime msec localtime show-timezone
service password-encryption
no service udp-small-servers
no service tcp-small-servers
no service config
no service pad
no service dhcp
service tcp-keepalives-in
!
hostname <hostname>
!
username <user> password <password>
!
enable secret <password>
no enable password
!
aaa new-model
aaa authentication login default group tacacs+ local-case
aaa authentication enable default group tacacs+ enable
aaa authorization commands 15 default group tacacs+ local
aaa accounting exec default stop-only group tacacs+
aaa accounting commands 15 default stop-only group tacacs+
aaa accounting network default stop-only group tacacs+
aaa accounting system
aaa accounting connection
tacacs-server host <ip address>
tacacs-server key <shared secret key>
!
ip subnet-zero
no ip domain-lookup
no service finger
no ip bootp server
no ip http server
no tftp-server
no ip identd
no ip source-route
!
clock timezone GMT 0
!
interface Ethernet0\0
 ip address <ip address> <netmask>
 description Inside Interface to DMZ
 no ip redirects
 no ip unreachables
 no ip proxy-arp
 no ip mroute-cache
 no cdp enable
 no ip mask-reply
 ip access-group inbound-filter out
 ip access-group outbound-filter in
 no ip directed-broadcast
```

```
 ip accounting access-violations
!
interface Serial0\0
ip address <ip address> <netmask>
 description Outside Interface to Internet
 no ip redirects
 no ip unreachables
 no ip proxy-arp
 no ip mroute-cache
 no cdp enable
 no ip mask-reply
 ip access-group inbound-filter in
 ip access-group outbound-filter out
 no ip directed-broadcast
 ip accounting access-violations
!
ip classless
!
no cdp run
!
ip route 0.0.0.0 0.0.0.0 <next hop router>
!
banner login #
This is a private system.

Use by unauthorized persons is prohibited.

All accesses to this service are logged.
#
!
logging buffered 32768 informational
no logging console
logging trap debugging
logging facility local7
logging <syslog server IP address>
!
access-list 90 permit <SNMP host> log
access-list 90 deny any log
access-list 99 permit 199.80.46.15 log
access-list 99 deny any log
ip access-list extended inbound-filter
deny ip <local networks> <local networks wildcard> any log
deny ip 0.0.0.0 0.255.255.255 any log
deny ip host 255.255.255.255 any log
deny ip 127.0.0.0 0.255.255.255 any log
deny ip 224.0.0.0 15.255.255.255 any log
deny ip 240.0.0.0 7.255.255.255 any log
deny ip 10.0.0.0 0.255.255.255 any log
deny ip 172.16.0.0 0.15.255.255 any log
deny ip 192.168.0.0 0.0.255.255 any log
deny ip 192.0.2.0 0.0.0.255 any log
deny ip 169.254.0.0 0.0.255.255 any log
deny ip host 0.0.0.0 any log
permit icmp any <local networks> <local networks wildcard>
packet-too-big
```

```
 permit icmp any <local networks> <local networks wildcard>
echo-reply
 permit icmp any <local networks> <local networks wildcard>
echo
 permit icmp any <local networks> <local networks wildcard> ttl-
exceeded
 deny   icmp any any
permit ip any <local networks> <local networks wildcard>
deny udp any range 1 65535 any log
deny tcp any range 1 65535 any log
deny ip any any log
ip access-list extended outbound-filter
permit icmp <local networks> <local networks wildcard> any
packet-too-big
 permit icmp <local networks> <local networks wildcard> any
echo
 permit icmp <local networks> <local networks wildcard> any
echo-reply
 permit icmp <local networks> <local networks wildcard> any ttl-
exceeded
deny icmp any any log
permit ip <local networks> <local networks wildcard> any
deny udp any range 1 65535 any log
deny tcp any range 1 65535 any log
deny ip any any log
access-list 120 permit tcp any <screened subnet> <screened
subnet wildcard>
!
ip tcp intercept list 120
ip tcp intercept connection-timeout 60
ip tcp intercept watch-timeout 10
ip tcp intercept one-minute low 1500
ip tcp intercept one-minute high 6000
!
snmp-server community <string> RO 90
snmp-server trap-authentication
snmp-server host <snmp ip> <authentication string>
!
line con 0
 transport input none
line aux 0
 no exec
 exec-timeout 0 10
 transport input none
line vty 0 4
 access-class 99 in
 exec-timeout 15 0
 password <choose a password>
 login
 transport input telnet ssh
!
ntp server <server IP address> key <key number>
ntp authenticate
ntp authentication-key <key number> md5 <string>
ntp trusted-key <key number>
```

```
ntp update-calendar
!
end
```

References

Cisco Systems. “Cisco Connection Documentation.” Cisco Connection Online. 11 March 2002. URL: http://www.cisco.com/univercd/home/home.htm (1 Apr 2002).

Cisco Systems. “Internet Security Advisories.” Cisco Connection Online. 27 March 2002. URL: http://www.cisco.com/warp/public/707/advisory.html (1 Apr 2002).

Cisco Systems. “Improving Security on Cisco Routers.” Cisco Connection Online. 14 March 2002. URL: http://www.cisco.com/warp/public/707/21.html (1 Apr 2002).

National Security Agency. “Router Security Configuration Guide.” Cisco Router Security Recommendation Guides. 27 December 2001.
URL: http://nsa2.www.conxion.com/cisco/download.htm (1 Apr 2002)

Eldridge, Brett. “Building Bastion Routers Using Cisco IOS.” Phrack Magazine, Volume 9, Issue 55 (1999): article 10.
URL: http://www.phrack.com/show.php?p=55&a=10 (1 Apr 2002).

Thomas, Rob. “Secure IOS Template Version 2.3.” 16 October 2001.
URL: http://www.cymru.com/~robt/Docs/Articles/secure-ios-template.html (1 Apr 2002).

Jones, George M. “Benchmark Version 1.0 for Cisco IOS Routers.” 5 February 2002. Center for Internet Security, 2002.

Northcutt, Stephen. Network Intrusion Detection: An Analyst’s Handbook. Indianapolis: New Riders Publishing, 1999. 42.

Cheswick, William R. and Bellovin, Steven M. Firewalls and Internet Security. Reading: Addison Wesley Longman, Inc. 1994. 54-56.

Kaeo, Merike. Designing Network Security. Indianapolis: Cisco Press/Macmillan Technical Publishing, 1999. 257-268.

Last Updated: May 23rd, 2017

Upcoming Training

SANS Stockholm 2017	Stockholm, Sweden	May 29, 2017 - Jun 03, 2017	Live Event
SANS San Francisco Summer 2017	San Francisco, CA	Jun 05, 2017 - Jun 10, 2017	Live Event
Security Operations Center Summit & Training	Washington, DC	Jun 05, 2017 - Jun 12, 2017	Live Event
SANS Houston 2017	Houston, TX	Jun 05, 2017 - Jun 10, 2017	Live Event
Community SANS Ottawa SEC401	Ottawa, ON	Jun 05, 2017 - Jun 10, 2017	Community SANS
SANS Charlotte 2017	Charlotte, NC	Jun 12, 2017 - Jun 17, 2017	Live Event
SANS Rocky Mountain 2017 - SEC401: Security Essentials Bootcamp Style	Denver, CO	Jun 12, 2017 - Jun 17, 2017	vLive
SANS Secure Europe 2017	Amsterdam, Netherlands	Jun 12, 2017 - Jun 20, 2017	Live Event
Community SANS Portland SEC401	Portland, OR	Jun 12, 2017 - Jun 17, 2017	Community SANS
SANS Rocky Mountain 2017	Denver, CO	Jun 12, 2017 - Jun 17, 2017	Live Event
SANS Minneapolis 2017	Minneapolis, MN	Jun 19, 2017 - Jun 24, 2017	Live Event
SANS Columbia, MD 2017	Columbia, MD	Jun 26, 2017 - Jul 01, 2017	Live Event
SANS Cyber Defence Canberra 2017	Canberra, Australia	Jun 26, 2017 - Jul 08, 2017	Live Event
SANS Paris 2017	Paris, France	Jun 26, 2017 - Jul 01, 2017	Live Event
SANS London July 2017	London, United Kingdom	Jul 03, 2017 - Jul 08, 2017	Live Event
Cyber Defence Japan 2017	Tokyo, Japan	Jul 05, 2017 - Jul 15, 2017	Live Event
SANS Cyber Defence Singapore 2017	Singapore, Singapore	Jul 10, 2017 - Jul 15, 2017	Live Event
Community SANS Minneapolis SEC401	Minneapolis, MN	Jul 10, 2017 - Jul 15, 2017	Community SANS
SANS Los Angeles - Long Beach 2017	Long Beach, CA	Jul 10, 2017 - Jul 15, 2017	Live Event
SANS Munich Summer 2017	Munich, Germany	Jul 10, 2017 - Jul 15, 2017	Live Event
Community SANS Phoenix SEC401	Phoenix, AZ	Jul 10, 2017 - Jul 15, 2017	Community SANS
Mentor Session - SEC401	Ventura, CA	Jul 12, 2017 - Sep 13, 2017	Mentor
Mentor Session - SEC401	Macon, GA	Jul 12, 2017 - Aug 23, 2017	Mentor
Community SANS Colorado Springs SEC401	Colorado Springs, CO	Jul 17, 2017 - Jul 22, 2017	Community SANS
Community SANS Atlanta SEC401	Atlanta, GA	Jul 17, 2017 - Jul 22, 2017	Community SANS
SANSFIRE 2017	Washington, DC	Jul 22, 2017 - Jul 29, 2017	Live Event
SANSFIRE 2017 - SEC401: Security Essentials Bootcamp Style	Washington, DC	Jul 24, 2017 - Jul 29, 2017	vLive
Community SANS Charleston SEC401	Charleston, SC	Jul 24, 2017 - Jul 29, 2017	Community SANS
Community SANS Fort Lauderdale SEC401	Fort Lauderdale, FL	Jul 31, 2017 - Aug 05, 2017	Community SANS
SANS San Antonio 2017	San Antonio, TX	Aug 06, 2017 - Aug 11, 2017	Live Event
SANS Prague 2017	Prague, Czech Republic	Aug 07, 2017 - Aug 12, 2017	Live Event