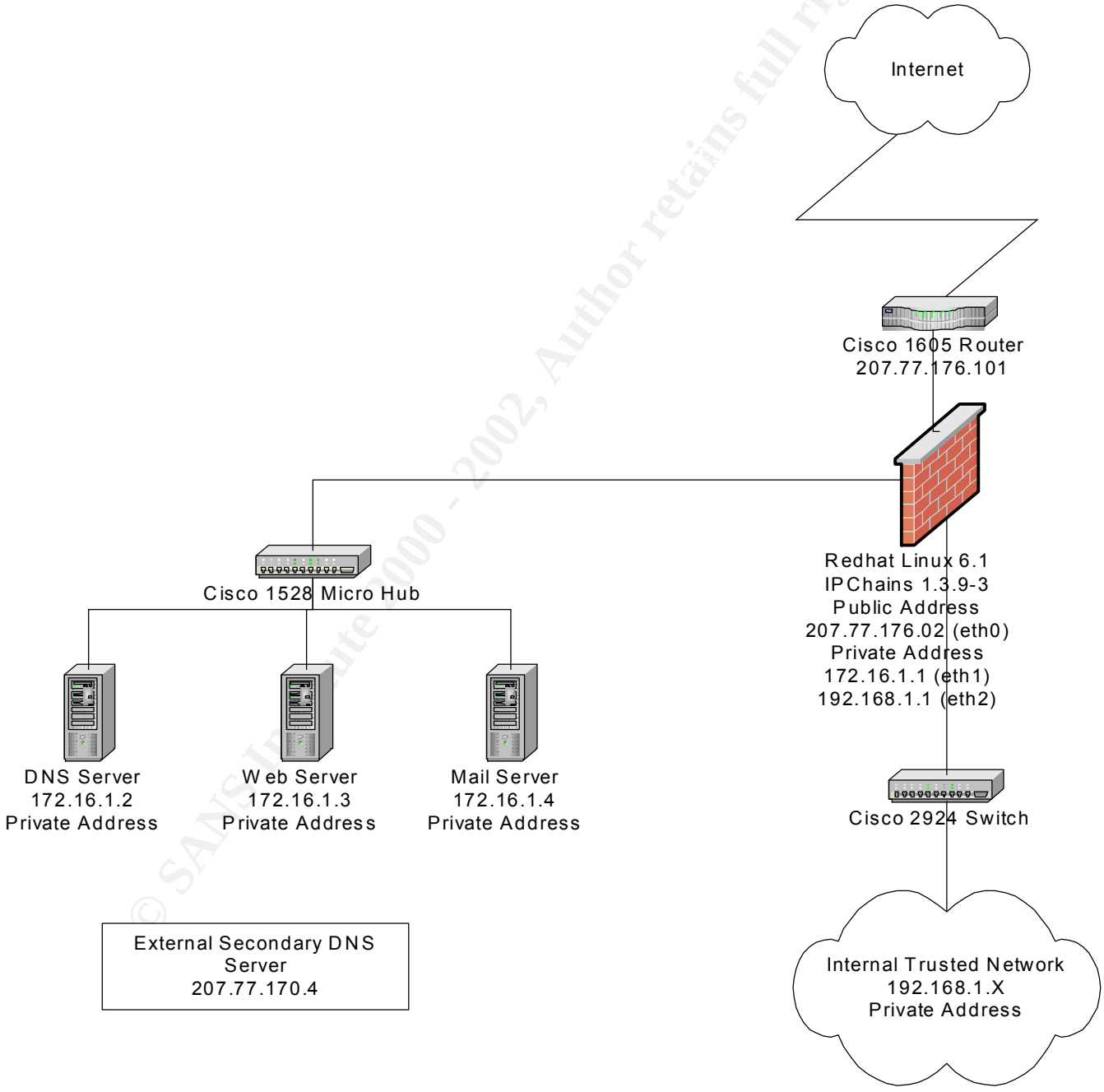
Global Information Assurance Certification Paper

Network Overview

Below is the design of my hypothetical network. It consists of a trusted internal network, a screened sub-network that includes the web, mail and external DNS server, and the connection to the Internet. All public IP addresses are fictional and hopefully, do not correspond to any real network locations.



Rather than deal with each of the 11 requirements individually, I have taken a holistic approach.

The default policy for the network is to deny all except specifically approved traffic. This will take care of most of the blocks required by this assignment, and will leave only the permitted traffic to be dealt with. This also will provide strong protection with a simple rule base in the firewall and simple ACL's in the router.

The border router will be used to filter as much traffic from the Internet as possible to reduce the load on the firewall. The firewall will be used to block as much outbound traffic from the internal and screened networks as possible.

Inbound traffic that will be allowed to the screened network:

1. DNS Queries (53/UDP to 172.16.1.2 only).
2. DNS Zone Transfer (53/TCP to 172.16.1.2 from 207.77.170.2 only).
3. Web Traffic (HTTP 80/TCP, SSL 443/TCP to 172.16.1.3 only).
4. SMTP Mail (25/TCP to 172.16.1.4 only).
5. POP Mail (109/TCP and 110/TCP to 172.16.1.4 from 192.168.1.x only).

Outbound traffic that will be allowed from the screened network:

1. DNS responses (53/UDP from 172.16.1.2 only).
2. DNS Zone Transfers (53/TCP from 172.16.1.2 to 207.77.170.4 only).
3. Web HTTP and SSL Traffic (80/TCP and 443/TCP).
4. SMTP Mail (25/TCP to the Internet only)
5. POP responses (109/TCP and 110/TCP to 192.168.1.X).

The effect of this policy is to limit the network's exposure to the minimum services that are required. DNS queries will be serviced from all addresses. However, DNS Zone Transfer requests will only be allowed if they originate from the ISP's DNS server that is acting as the external secondary DNS server. All Web requests will be serviced but only at the Web server. All SMTP mail will be serviced by the mail server only, and it will only service POP requests from the internal network.

Ramifications to this design include:

1. Company road warriors and telecommuters will not have access to company email while out of the office.
2. Limited extranet capabilities for key customers and vendors since there is no interactive access from the Web server to the internal network.
3. More difficult management of screened and external network devices with SNMP or TFTP capabilities. Console administration will be required for the external and screened network devices since no remote login services will be allowed.
4. Since no file transfers are allowed, all updates to the Web server will have to be done physically. Additionally, any backup of the screened servers will have to be done directly on them.

Inbound traffic that will be allowed to the internal network:

1. DNS responses (source port 53/UDP from 172.16.1.2 and 207.77.170.4 only).
2. POP mail (source ports 109/TCP or 110/TCP from 172.16.1.4 only).
3. HTTP and SSL (source ports 80/TCP or 443/TCP).

Outbound traffic that will be allowed from the internal network:

1. DNS queries (53/UDP to 172.16.1.2 and 207.77.170.4 only).
2. POP (109/TCP and 110/TCP traffic to the 172.16.1.4 only).
3. HTTP (80/TCP) and SSL (443/TCP).

The effect of this policy is to limit our internal clients to utilizing our mail server for in/outbound mail. The only other functions that would be allowed is Web browsing.

One downside to allowing access to the Web is that our clients can still access external public mail services such as AOL or Hotmail. If the intent of limiting access to the screened mail server was to simplify monitoring of email communications, allowing access to the Web works against that objective.

Inbound traffic from the Internet traffic that will be dropped regardless of the destination address or port:

1. Source routed traffic.
2. Small Services
3. Finger
4. IP traffic with a source address from the internal and screened address space.
5. IP traffic with a source address from a private address.
6. ICMP echo reply, echo request, time-exceeded, and unreachable packets..

The effect of this policy is to eliminate inbound spoofed traffic and various network mapping techniques. All traffic affected by this policy will be dropped silently to provide as much stealth as possible to the network.

Outbound traffic that will be blocked from the internal network regardless of the destination address or port:

1. Source routed traffic
2. Small services
3. Any traffic not from the firewall.

The effect of this policy is to prevent our network from being used as a source of spoofed traffic for any type of connection.

Configuring the Border Router

1. Update the router to Cisco IOS Release 12.1T
2. Install ACL list.

ACL Commands

1 Drop source routed packets
1.1. Command: no ip source-route
1.2. Description: Global configuration command to stop IP packets with source routing header options.
1.3. Order Precedence: Should be placed with other global configuration commands at the top of the ACL list. Since all source route packets are going to be dropped, it will minimize the processing of these packets.

2 Drop small services
2.1. Command: no service tcp-small-servers
no service udp-small-servers
2.2. Description: Disables all services below 20/TCP and 20/UDP (echo, discard, chargen, and daytime).
2.3. Order Precedence: Should be placed with other global configuration commands at the top of the ACL list. It is best to do it as soon as possible to minimize the processing of these packets.

3 Drop finger requests
3.1. Command: no service finger
3.2. Description: Drops all finger connection attempts that can be used to identify who is logged on and from where.
3.3. Order Precedence: Should be placed with other global configuration commands at the top of the ACL list. It is best to do it as soon as possible to minimize processing these types of packets.

4 Disable bootp
4.1. Command: no ip bootp
4.2. Description: Disables the bootp server service. This is part of the router hardening process of turning off unneeded services.
4.3. Order Precedence: Should be placed with other global configuration commands at the top of the ACL list. It is best to do it as soon as possible to minimize processing the packet.

5 Disable http
5.1. Command: no ip http
5.2. Description: Disables the http server service. This is part of the router hardening process of turning off unneeded services.
5.3. Order Precedence: Should be placed with other global configuration commands at the top of the ACL list.

6. Disable outbound ICMP unreachable packets
6.1. Command: no ip unreachables
6.2. Description: Disables the generation of ICMP unreachable messages.
6.3. Order Precedence: Should be placed with other global configuration commands at the top of the ACL list. Used to thwart port and address scans used to map the network.

7. Disable outbound ICMP echo-reply packets
7.1. Command: access-list 100 deny ICMP any echo-reply
7.2. Description: Disables all inbound echo-reply packets.

7.3. Order Precedence: Should be placed soon after the global configuration commands to minimize processing times. All subsequent references to access-list 100 are to the serial outbound group.

8. Disable outbound ICMP time-exceeded packets
8.1. Command: access-list 100 deny ICMP any time-exceeded
8.2. Description: Disables outbound traceroute response packets.
8.3. Order Precedence: Should be placed soon after the global configuration commands to minimize processing times.

9. Enable all remaining outbound traffic.
9.1. Command: access-list 100 permit IP any any
9.2. Description: Allows all other types of IP traffic from any source to any destination. Required because once an ACL is defined for an interface the router defaults to disabling all traffic except that which is specifically allowed.
9.3. Order Precedence: Should be placed at the end of the access-list so that all blocking actions have been taken.

10. Block all inbound echo request traffic
10.1. Command: access-list 101 deny ICMP any echo-request
10.2. Description: Drops all inbound requests for echo replies. This is important to prevent various network mapping techniques. It also prevents the broadcast addresses from being used for network mapping or possibly DoS attacks. All subsequent references to access-list 101 are to the serial inbound group.
10.3. Order Precedence: Should be placed at the start of the ACL list to minimize processing requirements for these packets.

11. Block all inbound traffic that comes from a private address
11.1. Commands: access-list 101 deny 10.0.0.0 0.255.255.255 any
access-list 101 deny 172.16.0.0 0.15.255.255 any
access-list 101 deny 192.168.0.0 0.0.255.255 any
11.2. Description: Drops all inbound traffic that has a private source address as defined by RFC 1918. Any inbound traffic from these addresses is obviously spoofed.
11.3. Order Precedence: Should be placed near the start of inbound access-list to minimize processing by immediately dropping spoofed traffic.

12. Block all inbound traffic that comes from the loop back network (127.X.X.X)
12.1. Command: access-list 101 deny 127.0.0.0 0.255.255.255 any
12.2. Description: Drops all inbound traffic that has a loop back source address because traffic from this address space is obviously spoofed.
12.3. Order Precedence: Should be placed at the start of inbound access-list to minimize processing by immediately dropping spoofed traffic.

13. Block all inbound login traffic
13.1. Commands: access-list 101 deny tcp any eq 23
access-list 101 deny tcp any eq 22
access-list 101 deny tcp any eq 21
access-list 101 deny tcp any eq 512
access-list 101 deny tcp any eq 513
access-list 101 deny tcp any eq 514
13.2. Description: Drops all inbound traffic for the Telnet, SSH, FTP, Rlogin and syslog ports. A common DoS is to flood the syslog files in hopes of filling the disk drive that it resides on. The other services are important to protect because they can provide command line access if they are enabled and especially if there is an improperly configured .rhost file somewhere on the network.

13.3. Order Precedence: No particular order precedence applies.

14. Block all inbound RPC and NFS traffic
14.1. Commands: access-list 101 deny tcp any eq 111
access-list 101 deny udp any eq 111
access-list 101 deny tcp any eq 2049
access-list 101 deny udp any eq 2049
access-list 101 deny tcp any eq 4045
access-list 101 deny udp any eq 4045
14.2. Description: Drops all inbound traffic for the RPC, NFS and lockd ports.
14.3. Order Precedence: No particular precedence applies.

15. Block all inbound NETBIOS traffic
15.1. Commands: access-list 101 deny tcp any eq 135
access-list 101 deny udp any eq 135
access-list 101 deny tcp any eq 137
access-list 101 deny udp any eq 137
access-list 101 deny tcp any eq 138
access-list 101 deny udp any eq 138
access-list 101 deny tcp any eq 139
access-list 101 deny udp any eq 139
access-list 101 deny tcp any eq 445
access-list 101 deny udp any eq 445
15.2. Description: Drops all inbound traffic for the NETBIOS ports. Port 445 is a recent addition to the list of ports associated with NETBIOS due to the release of Windows 2000.
15.3. Order Precedence: No particular precedence applies.

16. Block all inbound LDAP traffic
16.1. Command: access-list 101 deny tcp any eq 389
access-list 101 deny udp any eq 389
16.2. Description: Drops all inbound traffic for the Lightweight Directory Access Protocol.
16.3. Order Precedence: No particular precedence applies.

17. Block all inbound POP and IMAP traffic
17.1. Command: access-list 101 deny tcp any eq 109
access-list 101 deny tcp any eq 110
access-list 101 deny tcp any eq 143
17.2. Description: Drops all inbound POP and IMAP mail traffic.
17.3. Order Precedence: No particular precedence applies.

18. Block all inbound time traffic
18.1. Command: access-list 101 deny tcp any eq 37
access-list 101 deny udp any eq 37
18.2. Description: Drops all inbound traffic for the time ports.
18.3. Order Precedence: No particular precedence applies.

19. Block all inbound TFTP traffic
19.1. Command: access-list 101 deny udp any eq 69
19.2. Description: Drops all inbound traffic for the TFTP port. This is important because the router's ACL can be updated via TFTP. This prevents the router from being reconfigured from the outside.
19.3. Order Precedence: No particular precedence applies.

20. Block all inbound NNTP traffic
20.1. Command: access-list 101 deny tcp any eq 119
20.2. Description: Drops all traffic for the Network News Transport Protocol.
20.3. Order Precedence: No particular precedence applies.

21. Block all inbound NTP traffic
21.1. Command: access-list 101 deny tcp any eq 123
21.2. Description: Drops all inbound traffic for the Simple Network Time Protocol.
21.3. Order Precedence: No particular precedence applies.

22. Block all inbound LPD traffic
22.1. Command: access-list 101 deny tcp any eq 515
22.2. Description: Drops all traffic for the LPD port. This is an important port to block because there is a plethora of buffer overflow exploits aimed at LPD and some lead to root access.
22.3. Order Precedence: No particular precedence applies.

23. Block all inbound SNMP traffic
23.1. Commands access-list 101 deny tcp any eq 161
access-list 101 deny udp any eq 161
access-list 101 deny tcp any eq 162
access-list 101 deny udp any eq 162
23.2. Description: Drops all inbound Simple Network Management Protocol traffic. SNMP allows for the remote management of network devices including the router. Therefore, it is important to protect the router from this type of traffic.
23.3. Order Precedence: No particular precedence applies.

24. Block all inbound BGP traffic
24.1. Command: access-list 101 deny tcp any eq 179
24.2. Description: Drops all inbound Border Gateway Protocol traffic. Since BGP is a router to router protocol, it is important to verify that the router supports an alternate routing protocol such as OSPF.
24.3. Order Precedence: No particular precedence applies.

25. Block all inbound SOCKS traffic
25.1. Command: access-list 101 deny tcp any eq 1080
25.2. Description: Drops all traffic for the SOCKS service that allows systems from behind the firewall to access the Internet. If the SOCKS port is visible from the Internet, it acts as an opening to the internal network.

26. Enable all remaining inbound traffic.
26.1. Command: access-list 101 permit IP any any
26.2. Description: Allows all other types of IP traffic from any source to any destination. Required because once an ACL is defined for an interface the router defaults to disabling all traffic except that which is specifically allowed.

26.3. Order Precedence: Should be placed at the end of the access-list so that all blocking actions have been taken.

27. Only allow traffic from the firewall into the inbound Ethernet. Drop all traffic from any other source.
27.1. Command: access-list 102 permit ip 207.77.176.2 0.0.0.0 any
27.2. Description: Permits traffic from only the external interface of the firewall into the inbound Ethernet adapter of the router. The firewall will be providing NAT services for the internal and screened network so only traffic from it should be going to the router. Since an ACL has been defined for this access-list, the router now defaults to denying all other traffic not specifically allowed.

No ACL is necessary for the outbound Ethernet because the router defaults to allowing all traffic through unless an ACL has been specified.

A concern I have with the network policy in this assignment is that it makes no provision for logging traffic that is dropped. This leaves you flying blind because you do not know what type of activity may be targeting your organization. It also vastly diminishes the opportunity to tighten security if it appears that an attempt is being made to penetrate the network. I would suggest adding logging to this ACL as well.

To set up logging perform the following:

1 Open port 514 from the router to the logging server
access-list permit tcp 207.77.176.1 0.0.0.0 207.77.176.2 0.0.0.0 eq 514
2 Enable logging from the router
logging 207.77.176.2 (This assumes the firewall will also be the syslogd. A better solution would be to have a dedicated syslogd server.)
3 Add the "log" keyword to any ACL that should be logged.

CONFIGURING THE FIREWALL

The router has now been configured to eliminate as much traffic as is possible. The next step is for the firewall to complete the process of determining whether the traffic should be allowed into the network and which server is responsible for handling it. Of course, the firewall should be hardened by applying all appropriate patches, uninstalling all unnecessary software, eliminating all services that are not needed, and securing any remaining ports. Once this has been done, it can then be set up as a firewall.

Since the firewall is running Redhat Linux 6.1, ipchains will be used to control the flow of traffic through it. Ipchains is normally configured within a shell script using a text editor.

The kernel should be configured with the following modules: CONFIG_EXPERIMENTAL, CONFIG_MODULES, CONFIG_NET, CONFIG_FIREWALL, CONFIG_IP_FORWARD, CONFIG_IP_NOSR, CONFIG_SYN_COOKIES, CONFIG_IP_FIREWALL, CONFIG_IP_FIREWALL_VERBOSE, CONFIG_IP_MASQUERADE.

For our purposes the script will be named "fw.rules" and will be located in /etc. The file should be owned by root and its permission should be set to read/write for root only (600).

To begin execution of the firewall, a simple script should be created in /etc/rc.d/init.d named "fwstart". The sole command in fwstart would be: ipchains-restore < /etc/fw.rules.

The script should be owned by root with read/write/execute permissions for root only (700). A symbolic link should be created to the script in rc3.d directory which is the boot directory where Linux traditionally starts network services. The link name should be in the format Snnfwstart where "nn" is the lowest number possible to start the firewall as early in the boot process as possible.

All commands should be executed in the order they are presented below.

IPCHAINS Commands

1 Set the command path
- 1.1 Command: PATH=/bin:/sbin:/usr/bin:/usr/sbin
- 1.2 Description: Ensures that all commands will be found when they are executed.

2 Shut down all the chains while the rule base is created
- 2.1 Commands:
```
ipchains -A input  -i ! lo -j DENY
ipchains -A output -i ! lo -j DENY
ipchains -A forward -j DENY
```
- 2.2 Description: The default chains are set to deny all traffic except for the loopback interface which is left open for the input and output chains.

3 Protect against spoofed addresses
- 3.1 Commands:
```
for FILE in /proc/sys/net/ipv4/conf/*/rp_filter
do
        echo 1 > $FILE
done
```
- 3.2 Description: rp_filter files indicate whether or not IP spoofing should be checked for an interface. A value of "1" indicates spoofing should be checked.

4 Turn on IP forwarding
- 4.1 Command: echo 1 > /proc/sys/net/ipv4/ip_forward
- 4.2 Description: IP forwarding is controlled by the ip_forward file. A value of "1" indicates that IP forwarding is activated.

5 Turn on SYN Flooding protection
- 5.1 Command: echo 1 > /proc/sys/net/ipv4/tcp_syncookies
- 5.2 Descripton: SYN cookies enable legitimate users to continue to connect when the network is under a SYN flood attack by issuing a cryptographic challenge. SYN cookies work transparently so no modification is required to the client's software.

6 Break the rule base into manageable chunks
- 6.1 Commands:
```
ipchains -N internal-dmz
ipchains -N internal-external
ipchains -N dmz-internal
ipchains -N dmz-external
ipchains -N external-dmz
ipchains -N external-internal
```
- 6.2 Descripton: Creates new chains for each possible combination of traffic. This will simplify the process of creating the rule base and make it more understandable.

7 Define what traffic the internal-dmz will be responsible for
- 7.1 Command: ipchains -A forward -s 192.168.1.0/24 -i eth1 -j internal-dmz
- 7.2 Description: Forward traffic from the internal network on the DMZ interface to the internal-dmz chain.

John M. Millican

8 Define what traffic the internal-external chain will be responsible for
- 8.1 Command: ipchains -A forward -s 192.168.1.0/24 -i eth0 -j internal-external
- 8.2 Description: Forward traffic from the internal network on the external interface to the internal-external chain.

9 Define what traffic the dmz-internal chain will be responsible for
- 9.1 Command: ipchains -A forward -s 172.16.1.0/255.15.0.0 -i eth2 -j dmz-internal
- 9.2 Description: Forward traffic from the DMZ on the internal interface to the dmz-internal chain.

10 Define what traffic the dmz-external chain will be responsible for
- 10.1 Command: ipchains -A forward -s 172.16.1.0/255.15.0.0 -i eth0 -j dmz-external
- 10.2 Description: Forward traffic from the DMZ on the internal interface to the dmz-external chain.

11 Define what traffic the external-dmz chain will be responsible for
- 11.1 Command: ipchains -A forward -i eth1 -j external-dmz
- 11.2 Description: Forward the remaining traffic on the DMZ interface to the external-dmz chain.

12 Define what traffic the external-internal chain will be responsible for
- 12.1 Command: ipchains -A forward -i eth2 -j external-internal
- 12.2 Description: Forward the remaining traffic on the internal interface to the external-internal chain.

13 Drop all remaining traffic
- 13.1 Command: ipchains -A forward -j DENY
- 13.2 Description: Drop any remaining traffic on any interface.

Filter traffic from the internal network to the DMZ

14 Accept DNS queries from the internal network to the DMZ
- 14.1 Command: ipchains -A internal-dmz -p udp -d 176.16.1.2 53 -j ACCEPT
- 14.2 Description: Allow DNS queries from the internal network to the DNS server in the DMZ.

15 Accept Web connections from the internal network to the DMZ
- 15.1 Commands: ipchains -A internal-dmz -p udp -d 176.16.1.2 80 -j ACCEPT
  ipchains -A internal-dmz -p tcp -d 176.16.1.3 445 -j ACCEPT
- 15.2 Description: Allow HTTP and SSL connections from the internal network to the Web server in the DMZ.

16 Accept SMTP traffic from the internal network to the DMZ
- 16.1 Command: ipchains -A internal-dmz -p tcp -d 176.16.1.4 25 -j ACCEPT
- 16.2 Description: Allow SMTP connections from the internal network to the Mail server in the DMZ.

17 Accept POP traffic from the internal network to the DMZ
- 17.1 Commands: ipchains -A internal-dmz -p tcp -d 176.16.1.4 109 -j ACCEPT
  ipchains -A internal-dmz -p tcp -d 176.16.1.4 110 -j ACCEPT
- 17.2 Description: Allow POP connections from the internal network to the Mail server in the DMZ.

18 Drop all remaining traffic from the internal network to the DMZ
- 18.1 Command: ipchains -A internal-dmz -j REJECT
- 18.2 Description: Self-explanatory. Since this is a connection from the internal network, we will allow error messages to be generated through the use of the REJECT keyword.

Filter traffic from the internal network to the Internet

19 Allow DNS queries from the internal network to the Internet
- 19.1 Command: ipchains -A internal-external -p udp -d 207.77.170.4 53 -j MASQ
- 19.2 Description: Allow DNS queries from the internal network to the secondary DNS server at the ISP only. Masquerade the internal client's IP address.

20 Allow Web traffic from the internal network to the Internet
- 20.1 Command: ipchains -A internal-external -p tcp --dport 80 -j MASQ
  ipchains -A internal-external -p tcp --dport 445 -j MASQ
- 20.2 Description: Allow all HTTP and SSL traffic from the internal network to the Internet. Masquerade the internal client's IP address.

21 Drop any other traffic from the internal network to the Internet
- 21.1 Command: ipchains -A internal-external -j REJECT
- 21.2 Description: Self-explanatory. Since this is a connection from the internal network, we will allow error messages to be generated through the use of the REJECT keyword.

Filter traffic from the DMZ to the internal network

22 Allow DNS responses from the DNS server in the DMZ to the internal network
- 22.1 Command: ipchains -A dmz-internal -p udp -s 172.16.1.2 53 -j ACCEPT
- 22.2 Description: Allows DNS responses from the external DNS server only. Since only UDP traffic is allowed, any DNS responses greater than 512 bytes will be lost since DNS switches over to TCP on those longer responses. This loss is accepted because opening up 53/TCP would allow DNS zone transfers from any internal DNS servers.

23 Allow Web traffic from the Web server to the internal network
- 23.1 Command: ipchains -A dmz-internal -p tcp ! -y -s 172.16.1.3 80 -j ACCEPT
  ipchains -A dmz-internal -p tcp ! -y -s 172.16.1.3 445 -j ACCEPT
- 23.2 Description: Allows Web traffic from the Web server in the DMZ to the internal network, but blocks connection attempts from the HTTP and SSL ports of the Web server to systems in the internal network.

24 Allow POP traffic from the Mail server to the internal network
- 24.1 Command: ipchains -A dmz-internal -p tcp ! -y -s 172.16.1.4 109 -j ACCEPT
  ipchains -A dmz-internal -p tcp ! -y -s 172.16.1.4 110 -j ACCEPT
- 24.2 Description: Allows POP traffic from the Mail server in the DMZ to the internal network, but blocks connection attempts from the POP ports of the Mail server to systems in the internal network.

25 Drop any other traffic from the DMZ to the internal network
- 25.1 Command: ipchains -A dmz-internal -j DENY
- 25.2 Description: Self-explanatory.

Filter traffic from the DMZ to the Internet

26 Allow the DNS server in the DMZ to send DNS traffic
- 26.1 Command: ipchains -A dmz-external -p udp -s 172.16.1.2 53 -j MASQ
- 26.2 Description: Allows DNS traffic to be sent to the Internet. Masquerades the DNS server's IP address. Since only UDP traffic is allowed, any DNS traffic greater than 512 bytes will be lost since DNS switches over to TCP on this longer traffic. This loss is accepted because opening up 53/TCP would allow DNS zone transfers from the external DNS server to the Internet.

27 Allow the DNS server in the DMZ to send DNS zone transfers
- 27.1 Command: ipchains -A dmz-external -p tcp -s 172.16.1.2 53 -d 207.77l.170.4 53 -j MASQ
- 27.2 Description: Allows the external DNS server to respond to DNS zone transfer requests from the secondary DNS server at the ISP only. Masquerades the DNS server's IP address.

28 Allow the Web server in the DMZ to send Web traffic
- 28.1 Commands: ipchains -A dmz-external -p tcp ! -y -s 172.16.1.3 80 -j MASQ
  ipchains -A dmz-external -p tcp ! -y -s 172.16.1.3 445 -j MASQ
- 28.2 Description: Allows Web traffic from the Web server in the DMZ to the Internet, but blocks connection attempts from the HTTP and SSL ports of the Web server to systems in the Internet. Masquerade the Web server's IP address.

29 Allow the Mail server in the DMZ to send SMTP
- 29.1 Command: ipchains -A dmz-external -p tcp -s 172.16.1.4 25 -j MASQ
- 29.2 Descripton: Allows SMTP traffic from the Mail server in the DMZ to the Internet, but blocks connection attempts from the SMTP port of the Mail server to systems in the Internet.

30 Drop all remaining traffic from the DMZ to the Internet
- 30.1 Command: ipchains -A dmz-external -j DENY
- 30.2 Description: Self-explanatory.

Filter traffic from the Internet to the DMZ

31 Allow DNS traffic from the Internet
- 31.1 Command: ipchains -A external-dmz -p udp -d 176.16.1.2 53 -j ACCEPT
- 31.2 Description: Allows DNS traffic from the Internet. Since only UDP traffic is allowed, any DNS traffic greater than 512 bytes will be lost since DNS switches over to TCP on

this longer traffic. This loss is accepted because opening up 53/TCP would allow DNS zone transfers requests to be accepted from the Internet.

32 Allow DNS zone transfers from the Internet
- 32.1 Command: ipchains -A external-dmz -p tcp -d 176.16.1.2 53 -s 207.77.170.4 53 -j ACCEPT
- 32.2 Description: Allows DNS zone transfer requests from the secondary DNS Server at the ISP only.

33 Allow Web requests from the Internet
- 33.1 Commands: ipchains -A external-dmz -p tcp -d 176.16.1.3 80 -j ACCEPT
  ipchains -A external-dmz -p tcp -d 176.16.1.3 445 -j ACCEPT
- 33.2 Description: Allows systems from the Internet to make Web requests to the Web server only.

34 Allow SMTP traffic from the Internet
- 34.1 Command: ipchains -A external-dmz -p tcp -d 176.16.1.4 25 -j ACCEPT
- 34.2 Description: Allows SMTP connections from the Internet to the Mail server only.

35 Drop all remaining traffic from the Internet to the DMZ
- 35.1 Command: ipchains -A external-dmz -j DENY
- 35.2 Description: Self-explanatory.

Filter traffic from the Internet to the internal network

36 Drop any traffic from the Internet to the internal network
- 36.1 Command: ipchains -A external-internal -j DENY
- 36.2 Description: Do not allow any connections from the Internet to any system within the internal network.

Take off the blocks and let the rule base do its job

37 Remove the blocks on the default chains
- 37.1 Commands: ipchains -D input 1
  ipchains -D forward 1
  ipchains -D output 1
- 37.2 Description: Deletes the chains created in step 2 above. The "1" refers to rule number 1 that was created.

Please note that the X-Windows services are implicitly blocked since they are not explicitly allowed.

As with the router and for the same reasons, I am concerned that no logging is being done.

To implement logging the following should be done. Edit /etc/syslog.conf and add the following entries:

- Kern.* /var/log/kernel.log
- kern.info /var/log/messages
- mail.* /var/log/maillog

Logging for most of the interesting events in the firewall rule set can be done by adding the –l options to the ipchains commands that have the DENY keyword.

SIMPLIFIED RESTATEMENT OF THE RULE BASE WITH MINIMAL COMMENTS

```
#  Shutdown all traffic while setting up the rule base
ipchains -A input  -i ! lo -j DENY
ipchains -A output -i ! lo -j DENY
ipchains -A forward        -j DENY


#  Turn on IP Forwarding
echo 1 > /proc/sys/net/ipv4/ip_forward

#  Turn on SYN Flooding protection
echo 1 > /proc/sys/net/ipv4/tcp_syncookies

#  Break up the problem into manageable chunks
ipchains -N internal-dmz
ipchains -N external-dmz
ipchains -N internal-external
ipchains -N dmz-internal
ipchains -N dmz-external
ipchains -N external-internal

#  Define possible combinations of traffic
ipchains -A forward -s 192.168.1.0/24 -i eth1 -j internal-dmz
ipchains -A forward -s 192.168.1.0/24 -i eth0 -j internal-external
ipchains -A forward -s 172.16.1.0/255.15.0.0 -i eth0 -j dmz-external
ipchains -A forward -s 172.16.1.0/255.15.0.0 -i eth2 -j dmz-internal
ipchains -A forward -i eth1 -j external-dmz
ipchains -A forward -i eth2 -j external-internal
ipchains -A forward -j DENY

#  Filter traffic from the internal network to the DMZ
ipchains -A internal-dmz -p udp -d 176.16.1.2 53  -j ACCEPT
ipchains -A internal-dmz -p tcp -d 176.16.1.3 80  -j ACCEPT
ipchains -A internal-dmz -p tcp -d 176.16.1.3 445 -j ACCEPT
ipchains -A internal-dmz -p tcp -d 176.16.1.4 25  -j ACCEPT
ipchains -A internal-dmz -p tcp -d 176.16.1.4 109 -j ACCEPT
ipchains -A internal-dmz -p tcp -d 176.16.1.4 110 -j ACCEPT
ipchains -A internal-dmz -j REJECT

#  Filter traffic from the internal network to the Internet
ipchains -A internal-external -p udp -d 207.77.170.4 53 -j ACCEPT
ipchains -A internal-external -p tcp --dport 80  -j MASQ
```

John M. Millican
Firewall and Perimeter Defenses
SANS DC 2000

```
ipchains -A internal-external -p tcp --dport 445 -j MASQ
ipchains -A internal-external -j REJECT

#  Filter traffic from the DMZ to the internal network
ipchains -A dmz-internal -p udp      -s 172.16.1.2 53  -j ACCEPT
ipchains -A dmz-internal -p tcp ! -y -s 172.16.1.3 80  -j ACCEPT
ipchains -A dmz-internal -p tcp ! -y -s 172.16.1.3 445 -j ACCEPT
ipchains -A dmz-internal -p tcp ! -y -s 172.16.1.4 109  -j ACCEPT
 ipchains -A dmz-internal -p tcp ! -y -s 172.16.1.4 110  -j ACCEPT
ipchains -A dmz-internal -j DENY

#  Filter traffic from the DMZ to the Internet
ipchains -A dmz-external -p udp      -s 172.16.1.2 53  -j MASQ
ipchains -A dmz-external -p tcp      -s 172.16.1.2 53  -d 207.77l.170.4 53 -j MASQ
ipchains -A dmz-external -p tcp ! -y -s 172.16.1.3 80  -j MASQ
ipchains -A dmz-external -p tcp ! -y -s 172.16.1.3 445 -j MASQ
ipchains -A dmz-external -p tcp      -s 172.16.1.4 25  -j MASQ
ipchains -A dmz-external -j DENY

#  Filter traffic from the Internet to the DMZ
ipchains -A external-dmz -p udp -d 176.16.1.2 53  -j ACCEPT
ipchains -A external-dmz -p tcp -d 176.16.1.2 53  -s 207.77.170.4 53 -j ACCEPT
ipchains -A external-dmz -p tcp -d 176.16.1.3 80  -j ACCEPT
ipchains -A external-dmz -p tcp -d 176.16.1.3 445 -j ACCEPT
ipchains -A external-dmz -p tcp -d 176.16.1.4 25  -j ACCEPT
ipchains -A external-dmz -j DENY

#  Filter traffic from the Internet to the internal network
ipchains -A external-internal -j DENY

#  Take the blocks and let the rule base do its job
ipchains -D input   1
ipchains -D forward 1
ipchains -D output  1
```