**Author: Rozana Rusli**
Date : 08/15/2000
Firewall and Perimeter Protection Practical

A. Introduction

Product Overview
This is a packet filtering Firewall based on the TCP/IP packet filtering software; IP Filter.
It is used as a loadable kernel module in the Open BSD kernel. Open BSD is chosen as it
is touted by a lot of security experts as one of the most secure Operating System available
today.

Network Setup
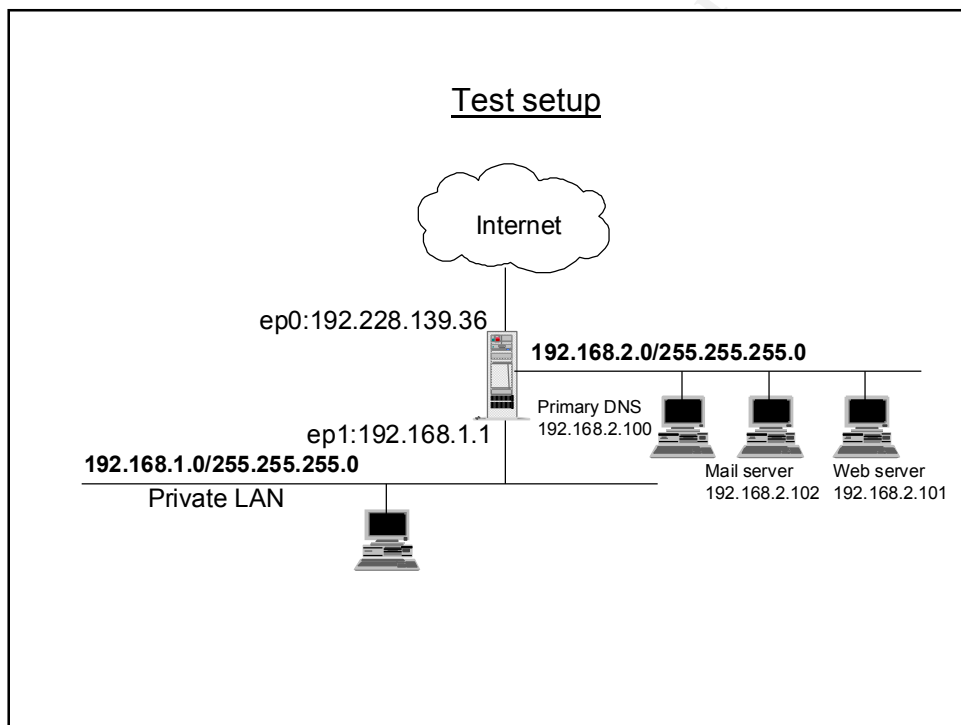
To facilitate the practical assignment, the above network setup is used as a point of
reference. The Firewall is provisioned with three interfaces, one is connected to the
Internet, one to the private network (that needs to be protected) and one is connected to
the screened network. The screened network is to provide access to some of the public
services such as the DNS service, web server and the mail server. The external interface
of the Firewall ep0 with ip address 192.228.139.36 connects to the outside world while
the internal interface ep1 connects with ip address 192.168.1.1 connects to the private
LAN. The hosts that are in the screened network are as follows:

- Primary DNS server with ip address 192.168.2.100. The secondary DNS is located externally at the ISP.
- Mail relay server. This server is used as a relay server for incoming and outgoing mail to the Internet. The ip address is 192.168.2.102
- Web server that host the company's web content which is publicly accessible via the Internet with ip address 192.168.2.101

B. The Ruleset

1.	Block "spoofed" addresses – packets coming from outside your company sourced from internal addresses or private addressess. Also block source routed packets.
Vulnerability and behaviour	Open to Denial of Service attacks
Description/Syntax	# Block traffic in on ep0 from 192.168.1.0 network to any # Packets "from" our network shouldn't be coming in from #outside. block in on ep0 from 192.168.1.0/24 to any # Block incoming traffic on ep0 from any to any with option of loose # source routing block in on ep0 from any to any with opt lsrr
How to apply	Apply this rule at the very top of the rule list before applying the tcp and udp filters to specific ports and hosts.
How to test	Try to access the internal host using the internal IP address. Use nmap to detect if this option is not blocked with the –D option or use hping2 software.
2.	Login services-- telnet (23/tcp), SSH (22/tcp), FTP (21/tcp), NetBIOS (139/tcp), rlogin et al (512/tcp through 514/tcp)

Vulnerability/Behaviour	*telnet (23/tcp) , ssh (22/tcp), rlogin* Allowing this services would mean allowing the possibilities of remote access. However for remote administration the rule is best defined by only allowing from a specific IP address TELNET - Most of the time intruders scan for this port simply to find out more about what operating system is being used. In addition, if the intruder finds passwords using some other technique, they will try the passwords here. SSH – TCP connections to this port might indicate a search for ssh, which has few exploitable features. Many version using the RSA REF library can be exploited if they are configured in a certain fashion. (Suggestion: run ssh on some other port). Also note that the ssh package comes with a program called make-ssh-known-hosts that will scan a domain for ssh hosts. UDP packets directed at this port along with 5632 indicate a scan for PCAnywhere *FTP* – the most common attack happens to "anonymous FTP" servers. These are servers with directories that can be written to and read from where hackers can transfer warez (pirated programs) and pr0n to avoid search engines classifying this document. NetBIOS (139/tcp) – this service (among other things) provide file sharing service, remote management and more. Incoming connections to this port are trying to reach NetBIOS/SMB, the protocols used for Windows "File and Print Sharing" as well as SAMBA. People sharing their hard disks on this port is probably the most common vulnerability on the Internet. Attempts on this port were common at the beginning of 1999, but tapered off near the end. Now at the start of year 2000, attempts on this port have picked up again. Several VBS (IE5 Visual Basic Scripting) worms have appeared that attempt to copy themselves on this port. Therefore, it may be worms that are attempting to propagate on this port. RLOGIN – it provides remote terminal service. It does not require the user to type in the username and if the connection is coming from a trusted host, the receiving computer lets the user log in without typing password. However, trusted hosts introduce security reason; you can't trust a host and you can't trust the user on that host. Trusted hosts and trusted users have been responsible for many security breaches. The trusted host is also vulnerable to IP spoofing.
Description/Syntax	# block and log incoming FTP traffic block in proto tcp from any to any port = 21 # block and log incoming SSH traffic block in proto tcp from any to any port = 22 # block and log incoming telnet traffic block in proto tcp from any to any port = 23 #block and log incoming Netbios traffic block in proto tcp from any to any port = 139 # block and log incoming rlogin traffic block in proto tcp from any to any 511><515

How to apply	The filtering rules above should be applied first before allowing them to specific hosts if the need arise. However it's best to deny these traffic from coming in from the outside world to the external Firewall interface ep0 as part of the Firewall locked down rule in order to avoid tools like nmap and firewalk from detecting the Firewall. For remote administration to the Firewall make sure the source ip is defined via a SSH connection.
How to test	TELNET – connect to the Internet and telnet to any host in the screened subnet. If the service is allowed the server will respond with the OS banner and prompts for password. Otherwise the host will reset the connection. Can try this with other host behind the Firewall. SSH – connect to the Internet and use a SSH client software ie F-Secure to connect to ep0 or other host behind the Firewall. If connection is allowed a SSh connection is established otherwise you'll get a connection reset. FTP – connect to the Internet and run ftp to any host in the screened subnet. If connection is allowed you'll get the FTP server banner. NETBIOS – You can use a utility such as net2bin.exe, nbtdump.exe utility or you can use the DOS command nbstat. This will tell you if such NETBIOS traffic is allowed. RLOGIN – use rsh client from a remote host.
3.	RPC and NFS-- Portmap/rpcbind (111/tcp and 111/udp), NFS (2049/tcp and 2049/udp), lockd (4045/tcp and 4045/udp)

Vulnerability/Behavior	RPC - Remote procedure calls (RPC) allow programs on one computer to execute programs on a second computer. They are widely-used to access network services such as shared files in NFS. Multiple vulnerabilities caused by flaws in RPC, are being actively exploited. This service can be used to survey your hosts for vulnerable RPC services. Access to portmapper is the first step in scanning a system looking for all the RPC services enabled, such as rpc.mountd, NFS, rpc.statd, rpc.csmd, rpc.ttybd, amd, etc. If the intruder finds the appropriate service enabled, s/he will then run an exploit against the port where the service is running. Note that by putting a logging daemon, IDS, or sniffer on the wire, you can find out what programs the intruder is attempting to access in order to figure out exactly what is going on. NFS - the program usually runs at this port. Normally, access to portmapper is needed to find which port this service runs on, but since most installations run NFS on this port, hackers/crackers can bypass portmapper and try this port directly.
Dexcription/Syntax	# Block incoming portmap traffic block in on ep0 proto tcp from any to any port=111 block in on ep0 proto udp from any to any port=111 # Block incoming NFS traffic block in on ep0 proto tcp from any to any port=2049 block in on ep0 proto udp from any to any port=2049
How to apply	Whenever possible turn off and/or remove these services on machines directly accessible from the Internet such as the Web, DNS and Mail servers. Make sure other RPC portmapper programs such as RPC nlockmgr service is also turned off.
How to test	Use "rpcinfo –p [host]". If the traffic is allowed, it will return the listing of all services running.
4.	NetBIOS in Windows NT -- 135 (tcp and udp), 137 (udp), 138 (udp), 139 (tcp). Windows 2000 – earlier ports plus 445(tcp and udp)

Vulnerability/Behavior	135 (tcp/udp) – Microsoft runs its DCE RPC end-point mapper for its DCOM services at this port. This has much the same functionality as port 111 for UNIX systems. Services that use DCOM and/or RPC register their location with the end-point mapper on the machine. When clients remotely connect to the machine, they query the end-point mapper to find out where the service is. Likewise, hackers can scan the machine on this port in order to find out such things as "is Exchange Server running on this machine, and which version?". This port is often hit in order to scan for services (for example, using the "epdump" utility), but this port may also be attacked directly. Currently, there are a few denial-of-service attacks that can be directed at this port.

137 and 138 (udp) - This is the most common item seen by firewall administrators and is perfectly normal. Please refer to explanation on NetBIOS for more details**.**

139 (tcp) - Windows NT comes with its NetBIOS services started by default; these services (among other things) provide the file sharing service, remote management and more. These services should be disabled when connecting a Windows NT machine to the Internet, since they pose a security Windows File Sharing may also be used to enumerate sensitive system information from NT systems. User and Group information (usernames, last logon dates, password policy, RAS information), system information, and certain Registry keys may be accessed via a "null session" connection to the NetBIOS Session Service. This information is typically used to mount a password guessing or brute force password attack against the NT target. |
Description/Syntax	# Block NETBIOS in Windows NT and Windows 2000 block in on ep0 proto tcp from any to any port =135 block in on ep0 proto udp from any to any port =135 block in on ep0 proto udp from any to any port =137 block in on ep0 proto udp from any to any port =138 block in on ep0 proto tcp from any to any port =139 block in on ep0 proto tcp from any to any port =445 block in on ep0 proto udp from any to any port =445
How to apply	NETBIOS is extremely chatty and it's best not to log these traffic simply because the NBT broadcast will be dropped and quickly filling the Firewall.
How to test	Use a utility software net2bin.exe or nbtdump.exe. or use the DOS command "nbtstat"
5. X Windows -- 6000/tcp through 6255/tcp	

Vulnerability/Behavior	X is a popular network-based window system that allows many programs to share a single graphical display. Each graphical device that runs X is controlled by a special program called the X-Windows Server. The X client xterm can exercise complete control over the display if connection is successful. This capability allows the flexibility in creating new clients which gives rich opportunity for Trojan horse programs. However the access is controlled and maintained in the host list via the xhost command. It is not suitable to environment in which workstation or servers are used by more than one person at a time. Server is susceptible to IP Spoofing.
Description/Syntax	# Blocking incoming X-Windows traffic from port 6000 through 6255 block in on ep0 proto tcp from any to any port 5999>< 6256
How to apply	For X-Windows to work it requires the service
How to test	Use xscan or nmap to check if this service is running. For nmap simply run "nmap 6000" or telnet to port 6000.

6.	Naming services-- DNS (53/udp) to all machines which are not DNS servers, DNS zone transfers (53/tcp) except from external secondaries, LDAP (389/tcp and 389/udp)

Vulnerability/ Behavior	DNS - The Berkeley Internet Name Domain (BIND) package is the most widely used implementation of Domain Name Service (DNS). According to a mid-1999 survey, about 50% of all DNS servers connected to the Internet are running vulnerable versions of BIND. In a typical example of a BIND attack, intruders erased the system logs, and installed tools to gain administrative access. They then compiled and installed IRC utilities and network scanning tools, which they used to scan more than a dozen class-B networks in search of additional systems running vulnerable versions of BIND. In a matter of minutes, they had used the compromised system to attack hundreds of remote systems abroad, resulting in many additional successful compromises. Hackers/crackers may be attempting to do zone transfers (TCP), to spoof DNS (UDP), or even hide other traffic since port 53 is frequently neither filtered nor logged by firewalls. An important thing to note is that you will frequently see port 53 used as the *source* UDP port. Stateless firewalls frequently allow such traffic on the assumption that it is a response to a DNS query. Hackers are increasingly exploiting this to pierce firewalls. Buffer overflow in L0pht AntiSniff allows remote attackers to execute arbitrary commands via a malformed DNS response packet
Description/Syntax	# block incoming DNS traffic to all machines which are not DNS servers. Allow DNS queries only to the external DNS and zone transfer from the external secondary DNS. Need to also block the LDAP ports. # block in on ep0 proto udp from any to any port=53 # block and log incoming tcp traffic port 53 to monitor attacks such as # DNS poison. block in log on ep0 proto tcp from any to any port =53 # block LDAP block in on ep0 proto tcp from any to any port = 389 block in on ep0 proto udp from any to any port =389 pass in on ep0 proto udp from any to 192.168.2.100 port =53 pass in on ep0 proto tcp from 161.142.201.17/32 to 192.168.2.100 port =53
How to apply	This is applied to all DNS servers that is hosted at your company while the secondary is located at the external network ie ISP's. It is also best that a split DNS is implemented to make sure that the information in the primary DNS is restricted to the host that you want to allow the outside world to know. To apply this filter block the incoming traffic before allowing it to from and to a specific host for zone transfer and only DNS queries to the the DNS server.

How to test	Use nslookup to test for name resolution. Connect to the Internet and run the following command to test for allowed connection. For DNS query: nslookup ➤ server 192.168.2.100 ➤ [type in the mail hostname] the result is the name will be resolved to an ip address. For DNS zone transfer Nslookup ➤ server 192.168.2.100 ➤ ls –d <your domain name> or you can use *dig* and point it to the name server. Then point it to other than the DNS server to make sure the filter is applied correctly.

7.	Mail-- SMTP (25/tcp) to all machines, which are not external mail relays, POP (109/tcp and 110/tcp), IMAP (143/tcp)

Vulnerability/Behavior	SMTP - Spammers are looking for SMTP server that allow them to "relay" spam. Since spammers keep getting their accounts shut down, they use dial-ups to connect to high bandwidth e-mail servers, then send a single message to the relay with multiple addresses. The relay then forwards to all the victims. There are also numerous holes in many SMTP servers that hackers/crackers can exploit to break in. Also by exploiting the sendmail vulnerability : EXPN and VRFY commands, a malicious user may be able to gather information, such as user names, about user accounts located on the system on which sendmail resides. Using this information, it would then be a relatively simple task for the malicious user to gain access to the system In one of the most common exploits, the attacker sends a crafted mail message to the machine running Sendmail, and Sendmail reads the message as instructions requiring the victim machine to send its password file to the attacker's machine (or to another victim) where the passwords can be cracked. IMAP and POP - are popular remote access mail protocols, allowing users to access their e-mail accounts from internal and external networks. The "open access" nature of these services makes them especially vulnerable to exploitation because openings are frequently left in firewalls to allow for external e-mail access. Attackers who exploit flaws in IMAP or POP often gain instant root-level control.
Description/Syntax	# Block all incoming SMTP traffic except to external mail relay server. # Block also POP and IMAP. # block in on ep0 proto tcp from any to any port =25 block in on ep0 proto tcp from any to any port =109 block in on ep0 proto tcp from any to any port =110 block in on ep0 proto tcp from any to any port =143 pass in on ep0 proto tcp from any to 192.168.2.102/32 port = 25 #

How to apply	Apply the block filters first before allowing the traffic in to the mail server (on the basis of last match wins). It is also a good idea to block incoming "identd" traffic. This service is used a lot by loggers, especially POP, IMAP, SMTP, and IRC servers. In general, if you have any clients accessing these services through a firewall, you will see incoming connection attempts on this port. Note that if you block this port, clients will perceive slow connections to e-mail servers on the other side of the firewall. Many firewalls support sending back a RST on the TCP connection as part of the blocking procedure, which will stop these slow connections. # Send a tcp-reset to connection to the ident port 113 with tcp flag set to syn and # syn ack block return-rst in quick proto tcp from any to any port = 113 flags S/SA
How to test	Telnet to host 192.168.2.102 port 25. The mail software banner will be displayed if connection is allowed. Otherwise, you'll get a connection reset. Use a POP and IMAP mail client to test the allowed connectivity. Point to the external mail server.

8.	Web-- HTTP (80/tcp) and SSL (443/tcp) except to external Web servers, may also want to block common high-order HTTP port choices (8000/tcp, 8080/tcp, 8888/tcp, etc.)
Vulnerability/ Behavior	Most web servers support CGI programs to provide interactivity in web pages. And many of these web servers come with sample CGI programs installed by default. Vulnerable CGI programs present a particular attractive target to intruders because they are relatively easy to locate and operate with the privileges and power of the web server software itself. A CGI hole is the most probable avenue of compromise. Allaire ColdFusion is a web server application package which includes vulnerable sample programs when installed. As a general rule, sample programs should always be removed from production systems. You may see scans for other proxies at the same time, such as at port 8000/8001/8080/8888.
Description/Syntax	# Block all incoming HTTP and SSL traffic except to external web server 192.168.2.101 # block in on ep0 proto tcp from any to any port = 80 block in on ep0 proto tcp from any to any port = 443 block in on ep0 proto tcp from any to any port >=8000 pass in on ep0 proto tcp from any to 192.168.2.101 port =80 pass in on ep0 proto tcp from any to 192.168.2.101 port =443
How to apply	Based on" last match wins" the rule is applied by blocking the incoming http traffic to all hosts before allowing the incoming traffic to the web server only. And to avoid for proxy scanning, apply the filter to deny the high-order http ports > = 8000

How to test	Telnet to the Web server 192.168.2.102 port 80. If connection is allowed the Web software banner is displayed. Otherwise the connection is terminated after timed out.
	Or use a browser client and point the URL to the webserver.

| **9.** | "Small Services"-- ports below 20/tcp and 20/udp, time (37/tcp and 37/udp) |

Vulnerability/Behavior	Some common vulnerabilities that occurs at port below 20 are as follows:
	Port =1 TCPMUX - Irix is the only major vendor that has implemented tcpmux, and it is enabled by default on Irix machines. Irix machines ship with several default passwordless accounts, such as lp, guest, uucp, nuucp, demos, tutor, diag, EZsetup, OutOfBox, and 4Dgifts. Many administrators forget to close these accounts after installation. Therefore, hackers scan the Internet looking first for tcpmux, then these accounts.
	Port = 7 ECHO - You will see lots of these from people looking for fraggle amplifiers sent to addresses of x.x.x.0 and x.x.x.255. A common DoS attack is an echo-loop, where the attacker forges a UDP from one machine and sends it to the other, then both machines bounce packets off each other as fast as they can.
	Port = 11 SYSSTAT - This is a UNIX service that will list all the running processes on a machine and who started them. This gives an intruder a huge amount of information that might be used to compromise the machine, such as indicating programs. With known vulnerabilities or user accounts. It is similar the contents that can be displayed with the UNIX "ps" command.
	Port = 19 CHARGEN - The character generator (chargen) service is designed to simply generate a stream of characters. It is primarily used for testing purposes. Remote users/intruders can abuse this service by exhausting system resources. Spoofed network sessions that appear to come from that local system's echo service can be pointed at the chargen service to form a "loop." This session will cause huge amounts of data to be passed in an endless loop that causes heavy load to the system. When this spoofed session is pointed at a remote system's echo service, this denial of service attack will cause heavy network traffic/overhead that considerably slows your network down. It should be noted that an attacker does not need to be on your subnet to perform this attack as he/she can forge the source addresses to these services with relative ease. Forging UDP packets between two chargen servers, or a chargen and echo can overload links as the two servers attempt to infinitely bounce the traffic back and forth. Likewise, the "fraggle" DoS attack broadcasts a packet destined to this port with a forged victim address, and the victim gets overloaded with all the responses.
Description/Syntax	# Small services. These includes services like tcpmux, echo, sysstat, # chargen, etc. block in on ep0 proto tcp from any to any port < 20 block in on ep0 proto udp from any to any port < 20 block in on ep0 proto tcp from any to any port = 37 block in on ep0 proto udp from any to any port = 37 #
How to apply	Be certain to block ports below 20 and not equal or less then 20. Since port tcp 20 is used for ftp-data.

How to test	All these services can be tested using nmap –p option or telnet to the respective ports.
10.	Miscellaneous-- TFTP (69/udp), finger (79/tcp), NNTP (119/tcp), NTP (123/tcp), LPD (515/tcp), syslog (514/udp), SNMP (161/tcp and 161/udp, 162/tcp and 162/udp), BGP (179/tcp), SOCKS (1080/tcp)

Vulnerability/Behavior	TFTP - Many servers support this protocol in conjunction with BOOTP in order to download boot code to the system. However, they are frequently misconfigured to provide any file from the system, such as password files. They can also be used to write files to the system.
	FINGER - Some finger daemons release information about the user's shell, home directory and group membership. This information may be used by hackers to attack the system. Some of the information can also be used to compromise the user account. For example, information such as the last time a user logged into the system could be used to build a table of usage patterns. Another example is that by knowing a user's home directory and exploiting a vulnerability in the mail system, a hacker could create an entrance into the system.
	NNTP – attempts on this port are usually by people hunting for open USENET servers. Most ISPs restrict access to their news servers to only their customers. Open news servers allow posting and reading from anybody, and are used to access newsgroups blocked by someone's ISP, to post anonymously, or to post spam.
	SYSLOG - an attacker will try to write directly to the syslog daemon. If successful it indicates an attacker could write enough erroneous data to your syslog file to fill your log files and cause hard disk failure. In order to stop this from occurring on your local subnet I suggest you configure your syslog daemon to handle access lists. Certain versions of Solaris syslogd will crash when they receive a syslog message off the network from a host without inverse DNS entries. This allows an attacker to disable security auditing before attacking a host, avoiding detection by programs like TCP wrappers. If the host is vulnerable, it's syslogd will be disabled, and must be re-started via administrative intervention. Obtain the Solaris patch for this problem.
	NTP – is the latest in a long series of protocols designed to let computers on a local or wide area network figure out the time. It can take into account the network delay and the existence of different servers with different clocks. Nevertheless, NTP was not designed to resist attack and several versions of ntpd can be fooled in making significant and erroneous changes to system's clock. Problems that may arise if the system's clock is changed are; attacker can attempt a replay attack, log files will no longer accurately inicate the correct time at which events took place and batch jobs run from cron daemon may not be executed if the system's clock jumps over the time specified in the crontab file.
	SNMP - a very common port that intruders probe for. SNMP allows for remote management of devices. It is widely used by network administrators to monitor and administer all types of network-connected devices ranging from routers to printers to computers. All the configuration and performance information is stored in a database that can be retrieved or set via SNMP. SNMP uses an unencrypted "community string" as its only authentication mechanism. Lack of encryption is bad enough, but the default community string used by the vast majority of SNMP devices is "public". Many managers mistakenly leave this available on the Internet. Crackers will first attempt to use the default passwords "public" and "private" to access the system.. Sniffed SNMP traffic can reveal a great deal about the structure of your network, as well as the systems and devices attached to it. Intruders use such information to pick targets and plan attacks. Attackers can use this vulnerability in SNMP to reconfigure or shut down devices remotely. SNMP packets may be mistakenly directed at your network. Windows machines running HP JetDirect remote management software uses SNMP, and misconfigured machines are frequent. HP OBJECT

Description/Syntax	# **10**. Miscellaneous services – these include services such as block in log on ep0 proto udp from any to any port = 69 block in log on ep0 proto tcp from any to any port = 79 block in log on ep0 proto tcp from any to any port = 119 block in log on ep0 proto tcp from any to any port = 123 block in log on ep0 proto tcp from any to any port = 515 block in log on ep0 proto udp from any to any port = 514 block in log on ep0 proto tcp from any to any port 160><163 block in log on ep0 proto udp from any to any port 160><163 block in log on ep0 proto tcp from any to any port =179 block in log on ep0 proto tcp from any to any port =1080 #
How to apply	Apply the less common or miscellaneous services last for ease of management and update. For most implementation this rules are applied by blocking any traffic that is not mentioned at the end of the ruleset.
How to test	SNMP – use snmp walk FINGER - Use "finger" command. There should be no reutn result if the filter is applied correctly. TFTP – use tftp command. If the traffic is allowed, the connection is established. NNTP – use browser client and point the news server to any host at the screened subnet. Use nmap to also check the other services.
11.	ICMP-- block incoming echo request (ping and Windows traceroute), block outgoing echo replies, time exceeded, and unreachable messages

Vulnerability/Behavior	ICMP is used by intruders to learn information about our network. ICMP is a lightweight set of application that were originally created for network troubleshooting. The purpose is to report errors rather than transferring information. And the most well known ICMP application is certainly the echo request/echo reply or ping. The ICMP echo request is one of the most common mapping technique. Therefore you should block incoming echo request. The "Ping of Death fragmentation attack' uses fragmented icmp packets for denial of service which created an IP packet that exceeds the maximum 65,535 bytes of data allowed by IP specification. This will cause the victim host to crash or freeze.
Description/Syntax	# ICMP can be a source of a lot of trouble for Internet Connected # networks. Blocking out all ICMP packets can be useful, but it will # disable some otherwise useful programs, such as "ping". Filtering on # ICMP type # allows for pings (for example) to work # Block incoming echo request (ping and traceroute) # block in on ep0 proto icmp from any to any icmp-type = 8 # # Block outgoing echo reply block out on ep0 proto icmp all icmp-type echo # # Block ICMP time exceeded # block out on ep0 proto icmp from any to any icmp-type = 11 # # block all ICMP destination unreachable packets which are port-unreachables # block out on ep0 proto icmp from any to any icmp-type unreach code 3
How to apply	This filter should be applied to among the first rules in the Firewall ruleset. Since ICMP is a lightweight protocol, drop these packets first.
How to test	Use traceroute or ping

This section describes the order of how the filters are applied. Prior to each filter a description is provided to explain how it works and the syntax of the filter. The Firewall should block everything first before allowing only which is specifically authorized. By default this Firewall is "deny all". The rule of thumb for this packet filtering implementation is "last match wins". If no rule is found to match, then the Firewall will drop the packet by default. The basic principle in a rule design is:

- keep the rulebase simple (no more than 30)
- limit the number of interface (no more than 5)
- position specific rule before general rule
- position commonly used rules first
- log maximum information possible where necessary

Recommended rules are:

- firewall lockdown rule
- authorised firewall administration
- eliminate "noise" rule

In summary the order of the rules are based on the following thoughts:

Order	Filter Type	Rule number	Explanation
1	Spoofing	1	To block incoming packets with sourced IP of the private LAN. To apply IP filtering rules first before moving on to packet filtering rules.
2	ICMP packets	11	To block all incoming ICMP traffic. However this will disable some other useful programs such as "ping". Filtering on icmp-type allows for ping to work. "Ping" uses icmp-type 0. Should you need to allow such traffic, the ICMP rules must come after the anti-spoofing rules to avoid for nasty smurf attack. Because we block spoofed traffic before the ICMP rules are processed, a spoofed packet never makes it to the ICMP rule-set.
3	Small services	9	Then begin filtering rules based on specific aspects such as port numbers. To determine that the filtering is done on the ports below 20; lower ports before moving on to higher ports.
4	Remote Access and resource sharing	2,3,5	To block any kind of remote access to the Firewall (lockdown rule) as well as to the network behind it and to deny remote resource sharing through the RPC exploits, and X-Windows vulnerability.
5	Windows/NT (noise filters)	4	Eliminate noise rule and to address all the Windows and NT related vulnerabilities.

| 6 | Common services | 6,7,8 | The common services include DNS, Mail and Web. These traffics are only allowed to specific host. In order for it to work correctly, deny all packets first before allowing it to the relevant hosts. |
| 7 | Miscellaneous | 10 | To address all services that are not so common such as network management, routing, time printer, socks services. To group them together for ease of management. |

Note: The following rules are based on the 11 questions of the practical assignment. These filters are only considering outgoing and incoming traffic applied at the external interface of the firewall. For actual implementation the allowed packets need to be defined. For instance like allowing internal users unrestricted Internet access and allowing ssh connection to the firewall for purpose of remote management from a specific source.

```
# The Firewall is running on a gateway with interface ep0 connects to the outside world and interface ep1
# connects to the network 192.168.1.0/255.255.255.0 which needs to be protected.
#
#
# 1. Block spoofing attempts. Packets "from" our network shouldn't be coming in from outside.
#
block in on ep0 from 192.168.1.0/24 to any
#
# The IP options is used here to block source routing. IP options have a bad name for being a general
# security threat. They can be of some use, however, to programs such as traceroute but many find this
# usefulness not worth the risk.
#
block in on ep0 all with opt lsrr
#
# 11. ICMP
# ICMP can be a source of a lot of trouble for Internet Connected networks. Blocking out all ICMP packets
# can be useful, but it will disable some otherwise useful programs, such as "ping". Filtering on ICMP type
# allows for pings (for example) to work.
# Block incoming echo request (ping and traceroute)
#
block in on ep0 proto icmp from any to any  icmp-type = 8
#
# Block outgoing echo reply
block out on ep0 proto icmp all icmp-type echo
#
# Block ICMP time exceeded
#
block out on ep0 proto icmp from any to any  icmp-type = 11
#
# block all ICMP destination unreachable packets which are port-unreachables
#
block out on ep0 proto icmp from any to any icmp-type unreach code 3
```

```
#
# 9. Small services. These includes services like tcpmux,echo,sysstat,chargen, etc.
block in on ep0 proto tcp from any to any port < 20
block in on ep0 proto udp from any to any port < 20
block in on ep0 proto tcp from any to any port = 37
block in on ep0 proto udp from any to any port = 37
#
# 2. Login services – block and log them.
#
block in log on ep0 proto tcp from any to any port =21
block in log on ep0 proto tcp from any to any port =22
block in log on ep0 proto tcp from any to any port =23
block in log on ep0 proto tcp from any to any port 511><515
#
# 3. Remote access and file sharing
# Block portmap/rpcbind, NFS and lockd
block in log on ep0 proto tcp from any to any port =111
block in log on ep0 proto udp from any to any port =111
block in log on ep0 proto tcp from any to any port =2049
block in log on ep0 proto udp from any to any port =2049
block in log on ep0 proto tcp from any to any port =4045
block in log on ep0 proto tcp from any to any port =4045
#
# Block X windows
block in on ep0 proto tcp from any to any port 5999><6256
#
# 4. Windows/NT
block in on ep0 proto tcp from any to any port =135
block in on ep0 proto udp from any to any port =135
block in on ep0 proto udp from any to any port =137
block in on ep0 proto udp from any to any port =138
block in on ep0 proto tcp from any to any port =139
block in on ep0 proto tcp from any to any port =445
block in on ep0 proto udp from any to any port =445
#
# Common services. These includes DNS, Mail, Web
# 6. DNS
#
block in on ep0 proto udp from any to any port=53
pass in on ep0 proto udp from any to 192.168.2.100/32 port =53
block in on ep0 proto tcp from any to any port = 389
block in on ep0 proto udp from any to any port =389
#
# 7. Mail
# Block all incoming SMTP traffic except to external mail relay server. Block also POP and IMAP.
#
block in on ep0 proto tcp from any to any port =25
block in on ep0 proto tcp from any to any port =109
block in on ep0 proto tcp from any to any port =110
block in on ep0 proto tcp from any to any port =143
pass in on ep0 proto tcp from any to 192.168.2.102/32 port = 25
#
# 8. Web
# Block all incoming HTTP and SSL traffic except to external web server 192.168.2.101
#
```

```
block in on ep0 proto tcp from any to any port = 80
block in on ep0 proto tcp from any to any port = 443
block in on ep0 proto tcp from any to any port >=8000
pass in on ep0 proto tcp from any to 192.168.2.101 port =80
pass in on ep0 proto tcp from any to 192.168.2.101 port =443
#
# 10. Miscellaneous services
block in log on ep0 proto udp from any to any port = 69
block in log on ep0 proto tcp from any to any port = 79
block in log on ep0 proto tcp from any to any port = 119
block in log on ep0 proto tcp from any to any port = 123
block in log on ep0 proto tcp from any to any port = 515
block in log on ep0 proto udp from any to any port = 514
block in log on ep0 proto tcp from any to any port 160><163
block in log on ep0 proto udp from any to any port 160><163
block in log on ep0 proto tcp from any to any port =179
block in log on ep0 proto tcp from any to any port =1080
#
# Block anything not mentioned
block in log quick on ep0
#
# Then continue with allowing incoming and outgoing packets
#
```

TIPS and TRICKS

Below is another option of how all of the filters can be applied. Try keeping the rules as simple as possible. The more complex the rulebase, the easier it is to make mistakes. And the simpler the rulebase, the more secure the firewall. If the rulebase is long, this is going to effect the performance of the Firewall.

Order	Filter Type	Explanation
1	Bad packets	Block any inherently bad packets coming in from the outside world. To drop all invalid packets like icmp redirect, short ip fragments and icmp echo to stop outsiders from learning your network.
2	Spoofing	To block incoming packets with sourced IP of the private LAN, block reserved IP. To apply IP filtering rules first before moving on to packet filtering rules.
3	Noise rules	Eliminate noise rule and to address all the Windows and NT related vulnerabilities.
4	Deny TCP/UDP packets	To deny all incoming tcp and udp packets that are not authorized.
5	Allowed Common services	These allow authorized traffic to and from to and from specific host.
6	Deny others not specified	Block all other services not mentioned based on a "last match wins" implementation

Note : The following rule is not complete. The allowed outgoing packets are not addressed here. However the assumption is that the policy allows that the internal users unrestricted Internet access. The rulebase is only considering what needs to be blocked at the external interface of the firewall.

```
# The Firewall is running on a gateway with interface ep0 connects to the outside world and interface ep1
# connects to the network 192.168.1.0/255.255.255.0 which needs to be protected. The ones in blue are tips
filters.
#
```
 # Block any inherently bad packets coming in from the outside world.
 # These include ICMP redirect packets, IP fragments so short the filtering rules won't be able to examine
 # the whole UDP/TCP header, and anything with IP options.
#
block in log quick on ep0 proto icmp from any to any icmp-type redir
block in log quick on ep0 proto tcp/udp all with short
block in log quick on ep0 from any to any with ipopts
```
#
#  ICMP can be a source of a lot of trouble for Internet Connected networks. Blocking out all ICMP packets
#  can be useful, but it will disable some otherwise useful programs, such as "ping". Filtering on ICMP type
#  allows for pings (for example) to work
#  Block incoming echo request (ping and traceroute)
#
block in on ep0 proto icmp from any to any  icmp-type = 8
#
# Block outgoing echo reply
block out on ep0 proto icmp all icmp-type echo
#
# Block ICMP time exceeded
#
block out on ep0 proto icmp from any to any  icmp-type = 11
#
# block all ICMP destination unreachable packets which are port-unreachables
#
block out on ep0 proto icmp from any to any icmp-type unreach code 3
#
#. Block spoofing attempts. Packets "from" our network shouldn't be coming in from outside.
#
block in on ep0 from 192.168.1.0/24 to any
#
```
 # Block reserved address
#
block in log quick from 10.0.0.0/8 to any
block in log quick from 192.168.0.0/16 to any
block in log quick from 172.16.0.0/12 to any
```
#
# The IP options is used here to block source routing. IP options have a bad name for being a general
# security threat. They can be of some use, however, to programs such as traceroute but many find this
# usefulness not worth the risk.
block in on ep0 all with opt lsrr
#
# Noise rules – NBT broadcast and other related Windows/NT traffic
block in on ep0 proto tcp from any to any port =135
block in on ep0 proto udp from any to any port =135
block in on ep0 proto udp from any to any port =137
block in on ep0 proto udp from any to any port =138
block in on ep0 proto tcp from any to any port =139
block in on ep0 proto tcp from any to any port =445
block in on ep0 proto udp from any to any port =445
#.
# TCP/UDP
```

```
#  To allow outgoing traffic. Uncomment this to allow outgoing access to the outside world
#  pass out log on ep0 proto tcp/udp from any to any keep state
#
#  Block all incoming UDP traffic except DNS traffic. NFS and portmap are special-cased and logged
#
block in log on ep0 proto udp from any to any
block in log on ep0 proto udp from any to any port = sunrpc
block in log on ep0 proto udp from any to any port = 2049
pass in quick on ep0 proto udp from any to 192.168.2.100 port = 53
#
```
 # Block and log all incoming TCP traffic connections to known services returning a connection reset so
 # that things like ident won't take forever timing out. But don't log ident (auth port) as it's so common.
#
block return-rst in log on ep0 proto tcp from any to any flags S/SA
block return-rst in on ep0 proto tcp from any to any port = auth flags S/SA
```
#
# Now allow various incoming TCP connections to particular hosts. TCP to the primary nameserver so
# secondaries can do zone transfers. SMTP to the mail host and HTTP to the web server .
#
# DNS
pass in on ep0 proto udp from any to 192.168.2.100/32 port =53
#
# Mail
#
pass in on ep0 proto tcp from any to 192.168.2.102/32 port = 25
#
# Web
#
pass in on ep0 proto tcp from any to 192.168.2.101 port =80
pass in on ep0 proto tcp from any to 192.168.2.101 port =443
#
# Block anything not mentioned
block in log quick on ep0
#
```