Global Information Assurance Certification Paper

SANS Firewall Practical

Rick Dreger
1/17/2005

Introduction:

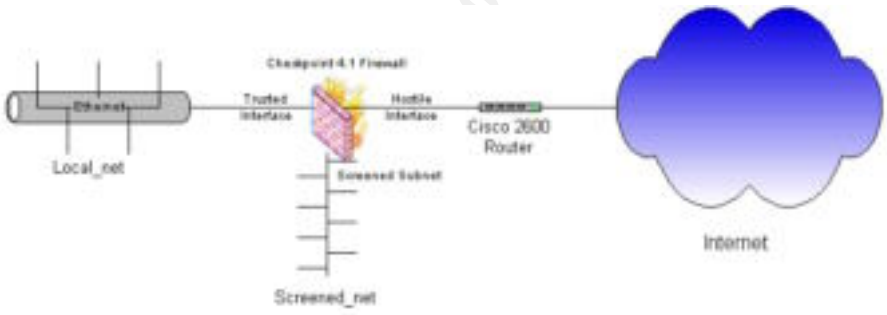
When one wishes to implement a firewall, or any kind of perimeter protection, understanding the target environment is essential. The client's requirements must be accurately defined so that the implemented solution will correctly address the needs. For the purpose of this Firewall practical, the assumption is made that all the recommended permit/denies will be followed, and then some additional rules have been added to supplement the policy. Further, detailed comments and caveats will be stated for each of the rules and the eleven recommendations. Please note that the scope of this security analysis has been narrowed to specifically target the perimeter defense design, and does not address host security, IDS, or other key layers in very much detail.

Perimeter Overview:

In order to implement perimeter security for this project the following choices were made:

- The firewall will be implemented using Checkpoint 4.1 running on a Nokia IP330.
- The firewall has three interfaces: one hostile (Internet) interface, one trusted (internal) interface, and one screened subnet interface.
- The firewall is sitting behind a Cisco 2600 router running IOS 12.x.
- The client is not using NAT and all local_net and screened_net IP addresses are Public.
- Routing between the router and firewall will be done using static routes.

Perimeter Design Diagram:



Firewall Information:

Note: This paper does not go into detail on configuring the Cisco router to be a useful perimeter protection device. Some router configuration steps are mentioned (such as for anti-spoofing or source-route blocking), but no significant time is spent in this document. The following web pages should provide useful information for securing the router:

- http://www.cisco.com/univercd/cc/td/doc/product/software/ios113ed/113ed_cr/secur_c/scoverv.htm
- http://www.cisco.com/univercd/cc/td/doc/product/software/ios113ed/113ed_cr/secur_c/scprt3/screflex.htm
- http://www.cert.org/ftp/tech_tips/packet_filtering

First Steps:

If we are starting out with a brand new IP330, with the FW-1 code running in a non-distributed configuration (i.e.: all modules residing on this one box), the information below *highlights* some of the key first steps:

- The Nokia IP330 appliance runs a pre-hardened, modified variant of the Free BSD OS called IPSO. Configuring the IP330 will initially involve connecting via a console cable (Tera Term Pro software works very well) and then configuring the network interfaces and services via lynx – a text based web browser. The interface displayed by lynx is the text form of Nokia's Voyager interface (i.e. Voyager can be thought of as IPSO's configuration GUI).
- Once the interfaces have been correctly configured, the Firewall package should be addressed next. The IPSO version of the Firewall-1 code already ships with the appliance, however new versions can be downloaded from the support site and then loaded/upgraded via the command line. (One must set up a login account at http://support.iprg.nokia.com before using the FAQ archive or download resources)
- Next, license and configure the firewall. This includes typing in license codes, adding remote GUI clients, adding administrator information, and choosing how IP forwarding should work. Once done, make sure the FW-1 daemons have correctly started (otherwise you will not be able to connect the remote GUI in the next step).
- For configuring the actual Firewall-1 policy, the following is suggested: Buy an inexpensive mini-hub or make use of a currently unused Ethernet hub in the office. Now plug the interface that's going to connect to the internal network into one of the ports. Make sure nothing else is connected into this mini hub as the firewall is at a very vulnerable stage! Connect a laptop, correctly addressed and loaded with the correct version of the FW-1 GUI into another port of the hub. Access should now be available via the FW-1 GUI and direct telnet or ssh access to the IP330.

Rule 0 Information:

Upon connecting the FW-1 GUI to the Firewall, one of the first steps is to lock down the Rule 0 "pseudo-rules." Although version 4.1 provides better logging functionality and opens up less services by default, for consistency it is useful to have all of one's rules directly in the rule base. See http://www.geek-speak.net/fw1/fw1_properties.html for some useful information on version FW-1 v. 4.x Rule 0 properties.

For this particular client the image below shows the configuration of the properties file. The key points are:

- All the implied rules have been removed.
- Rules checking will be applied Inbound
- Time out values have been left at the default values.

Properties Setup – Rule 0 configuration

Rules Base:

Now that the pseudo-rules have been removed, the next step is to build the rule base itself. In order to do this for the specified client, a combination of "good firewall policy practices" were used in combination with all of the eleven recommendations. A very useful resource on building or modifying a rule base comes from Lance Spitzner's "Building Your Firewall Rulebase" white paper (http://www.enteract.com/~lspitz/rules.html).

The rule base below represents the final rule base image for this project. Specific lines from this policy are referenced by items 1 through 11 in the "Addressing the Recommended Rules" section later in this paper.

Notes:

- Groups have been given clear names, though the items in those groups will not always be explicitly annotated. For example: the "secureadmins" group will contain workstations for admins allowed to connect to the firewall, though each workstation in that group will not be individually shown.
- Groups that contain multiple Services that directly relate to the eleven recommended actions, will however be explained in their appropriate section below.
- Most of the objects and some of the services will have to be created, though these steps are not all "spelled out" here.
- Many items have their own rule line for clarity purposes, though a few have multiple recommendations implemented on the same line.

Firewall Policy

No.	Source	Destination	Service	Action	Track	Install On	Time	Comment
1	secureadmins	Firebox	FireWall1 ssh	accept	Short	Gateways	Any	Allow Security Admins ssh and FW1 type access (GUI, etc)
2	Any	Firebox	Any	drop	Short	Gateways	Any	Drop all other traffic going to the firewall.
3	local_net	Any	http https	accept		Gateways	Any	Accept outbound web traffic.
4	Any	webserver_ext	http https	accept		Gateways	Any	Allow web traffic to web server on screened subnet.
5	Any	smtp_mailbox	smtp	accept		Gateways	Any	Allow connection via smtp to mail server.
6	smtp_mailbox	Any	smtp	accept		Gateways	Any	Allow mail server to send smtp requests
7	local_net	dns_ext	domain-udp	accept		Gateways	Any	Allow all but internal net to do nslookups on the external name server.
8	dns_ext	dns_int	dns	accept	Short	Gateways	Any	Allow external dns to do zone transfers from the internal dns (allow tcp and upd 53)
9	dns_int	Any	domain-udp	accept		Gateways	Any	Allow internal name server to do queries but no zone-xfers (53/tcp) or large queries.
10	secureadmins	screened_net	icmp_subnet	accept	Short	Gateways	Any	Allow security admins to query external boxes using ICMP tools
11	screened_net	secureadmins	icmp_reply_subnet	accept	Short	Gateways	Any	Allow screened subnet boxes to reply back to the admins' queries.
12	local_net screened_net	Any	echo-reply dest-unreach time-exceeded	drop	Short	Gateways	Any	Drop outbound icmp echo replies, unreachables, and time exceeded.
13	Any	local_net screened_net	echo-request time-exceeded traceroute	drop	Short	Gateways	Any	Deny incoming pings and traceroutes
14	Any	Any	login_svcs	drop	Short	Gateways	Any	Drop telnet, ssh, ftp, netbios and r-services
15	Any	Any	rpc_nfs_svcs	drop	Short	Gateways	Any	Drop: Portmap/rpcbind (111/tcp and 111/udp), NFS (2049/tcp and 2049/udp), lockd (4045/tcp and 4045/udp)
16	Any	Any	netbios_svcs	drop	Short	Gateways	Any	Block all netbios ports 135-139 tcp and udp plus 445 tcp/udp.
17	netmgmt	screened_net	snmp_161	accept	Short	Gateways	Any	Allow net mgmt box snmp access to screened_net
18	screened_net	netmgmt	snmp_traps	accept		Gateways	Any	Allow snmp-traps back to only the net management system
19	Any	Any	Xwindows Ldap_group getmail_group smallsvc_time_group	drop	Short	Gateways	Any	Drop: X Windows -- 6000/tcp through 6255/tcp Deny tcp/udp for ldap Deny POP, IMAP Deny small services and time.
20	Any	Any	misc_group	drop	Short	Gateways	Any	Deny services like: tftp, finger, NNTP, etc
21	Any	Any	tcp-high-ports	drop	Long	Gateways	Any	Use to monitor activity targetting high ports.
22	Any	Any	Any	drop	Long	Gateways	Any	Use to catch other traffic that doesnt match any of our rules.

Final Rule Base

Brief Rules Explanation:

Information on each of the rules, including any important information on their ordering, is briefly mentioned below, with much more complete information coming in the following sections. Although many of the deny rules could have been accounted for simply by using the "ANY, ANY, deny and log" rule, they have been explicitly stated to illustrate how they would be created. More complex policies would certainly call for more specific accept/deny rule combinations, so this rule base is a bit longer than it needs to be, but it is clearer as to what is being targeted.

- Rule 1: Allow security administrators to access the firewall both by ssh and the FW-1 GUI. Put this first so that the lock-down rule can be put second to secure the firewall.
- Rule 2: Deny any other traffic directly to the firewall. (lock-down rule)
- Rule 3: For this customer, web traffic from local_net to the Internet will be allowed. Since web traffic will probably be the most abundant type of traffic, this rule is the very first permit rule to try and enhance rule base performance. Note: Outbound web traffic is allowed, though web monitoring and screening could also be implemented per corporate requirements.
- Rule 4: Inbound web traffic to the external web server should also be abundant. This rule is set as the next accept rule to help minimize performance issues. See recommendation #8 in the "Addressing the Recommended Rules" section below for detailed information.
- Rule 5/6: Allow SMTP mail traffic. Since mail traffic is also very common, it is the next permit rule in the policy. See recommendation #7 in next section.
- Rule 7,8,9: Configure DNS access. DNS requests should also be fairly usual, so these represent the next accept rules in the policy. See recommendation #6 in next section.
- Rule 10- 13: Configure ICMP access. It was chosen that a small group of security admins should be allowed to use ICMP tools to the screened_net devices, and that those devices should be able to reply back to the admins. All other ICMP traffic should be denied (some explicitly stated, others dropped by the last rule). Since, ICMP ping enumeration "queries" are fairly common, these were chosen as the next deny rules. See recommendation #11 in next section for more information.
- Rule 14: Limit login services. See recommendation #2 in next section.
- Rule 15: Limit RPC services. See recommendation #3 in next section.
- Rule 16: Limit NetBIOS. See recommendation #4 in next section.
- Rule 17, 18: For Network management of the screened_net systems, some very limited SNMP access has been allowed for the netmgmt server only. See recommendation #10, SNMP sub-section for more information. Placement of this rule is flexible, but it must be above the SNMP drop function performed by Rule 20.
- Rule 19: Limit a variety of other services. See recommendations #5, 6, 7, and 9 in next section.
- Rule 20: Limit other miscellaneous services. See recommendation #10 in next section.
- Rule 21: Explicit deny all of tcp-high-ports. Useful to see what kind of trojan scans and other "high port" activity is hitting the firewall. This rule was placed second to last to ensure that all high ports not explicitly allowed in other rules were dropped.
- Rule 22: The explicit deny all and log rule. Useful to see what other kind of traffic is attempting to access resources. This rule is always the very last one. (Clean up rule)

Addressing the Recommended Rules

1. Spoofed Addresses:

Original Recommendation:

Block "spoofed" addresses-- packets coming from outside your company sourced from internal addresses or private (RFC1918 and network 127) addresses. Also block source routed packets.

Vulnerability:

Three address ranges fall into the category of Private IP address ranges, they are:

- 10.0.0.0/8
- 172.16.0.0/12
- 192.168.0.0/16

Although these address ranges provide a very useful way to conserve Public IP addresses (when mated to Network Address Translation), packets addressed with Private IP's coming from the Internet are anomalous. Per RFC 1918 (http://sunsite.cnlab-switch.ch/ftp/doc/standard/rfc/19xx/1918) devices with a Destination IP address in one of these ranges cannot be routed to and should be rejected. Hence, the response to any traffic entering a Publicly addressed network with a Private IP address in the SRC field cannot be routed back to. Therefore, this kind of traffic coming from or going to the Internet should not be considered valid traffic and should be dropped as it will just consume resources.

Source routed packets should also be blocked at the firewall in order to prevent hostile hosts from pretending to be trusted hosts. If a hostile device sends a packet with the SRC IP address of a trusted host and is using source routing, the reply from the host will navigate the source route back to the spoofing device. Denying source route traffic helps prevent this trust manipulation.

How to block it:

Since spoofed packets and source routed packets should never be allowed into the network, the Cisco router should be configured to drop them. Create a new extended access list and apply it in the INBOUND direction on the External router interface. To drop spoofed packets the ACL should include the following lines:

```
access-list 101 deny   ip 10.0.0.0 0.255.255.255 any log
access-list 101 deny   ip 172.16.0.0 0.15.255.255 any log
access-list 101 deny   ip 192.168.0.0 0.0.255.255 any log
```

To prevent source routed packets from entering the network use the following on the router:

```
no ip source-route
```

Order Dependencies:

As these rules are on the router and not the Checkpoint firewall they have been put at the top of an extended Access Control List (ACL) and then applied to the appropriate interface.

The router will drop the spoofed traffic as soon as it matches on the rule in the ACL, keeping the router from evaluating the packet against the rest of the List. Note that these lines are only the first few in the ACL. As it stands now, all inbound traffic is blocked. Other lines should be added to the ACL to allow appropriate traffic, such as "permit ip any any".

Testing:

The best way to test this is to try and send crafted packets from the Internet through the router to the protected network. These crafted packets should have Private IP's set in the SRC field or have the source route options set, depending on which test you are doing. The router's logs should then be checked to verify that the spoofed packet matched against the correct deny rule. The logs from the firewall (the next device in line) should also be checked to make sure that the source routed traffic and the spoofed traffic did not get to it.

2. Login Services

Original Recommendation:

Login services-- telnet (23/tcp), SSH (22/tcp), FTP (21/tcp), NetBIOS (139/tcp), rlogin et al (512/tcp through 514/tcp).

Vulnerability:

These services represent various remote access methods. Telnet is used for remote terminal access, where all the session information (login, passwords, data) is presented in clear text across the network. Secure shell (SSH) can be thought of as "encrypted telnet" where login, password, and data are all encrypted. Both of these access methods could allow remote users full command line access to systems, allowing for a full system compromise. Both services should be denied (at least to Internet hosts) by the firewall.

File Transfer Protocol (FTP) allows for the remote copying of data. FTP access could be abused by Internet users storing various warez or undesirable files on corporate systems. Further, if permissions on the FTP enabled system are not carefully set, users might modify files to exploit the r-services (rsh, rexec) or upload password files to crack and exploit.

Per the SANS top ten security threat page (http://www.sans.org/topten.htm): "For Windows NT systems, prevent anonymous enumeration of users, groups, system configuration and registry keys via the "null session" connection. Block inbound connections to the NetBIOS Session Service (tcp 139) at the router or the NT host." In this case, the Checkpoint firewall will be blocking the service.

The various r-services are: exec 512/tcp, login 513/tcp, and shell 514/tcp. These three services can pose a significant security risk to UNIX systems. The r-services are based upon a trust model which can provide access simply based upon the address of the host that is communicating with it. Thus, a user with access to a system without direct authentication.

The r-services vulnerability ties in with an unsecured FTP server and the possibility of manipulating the .rhosts and /etc/hosts.equiv files to gain access.

Firewall Rule:

Refer to rule #14 in the rulebase. The service group login_svcs contains: telnet (23/tcp), SSH (22/tcp), FTP (21/tcp), NetBIOS (139/tcp), and the r-services rlogin et al (512/tcp through 514/tcp).

Testing:

Once again, using an nmap scan of these ports in combination with checking the firewall log to ensure packets were dropped provides a good first test. A slightly slower, but more precise method would be to then try inbound telnets, ssh, r-services (rsh, etc), ftp and NetBIOS connections from an Internet host to ensure that access is blocked.

3. RPC / NFS / lockd

Original Recommendation:

RPC and NFS-- Portmap/rpcbind (111/tcp and 111/udp), NFS (2049/tcp and 2049/udp), lockd (4045/tcp and 4045/udp)

Vulnerability:

The portmap service is used to keep track of Remote Procedure Call services. These RPC's notify portmap as to which high ports that service is going to be using. By being able to connect to port 111, users can discern which RPC's are running on which ports. This information can then be exploited as the situation permits.

Network File Systems allows a remote machine to mount local file systems, much like the concept of sharing in Windows. Misconfiguring mount permissions can unintentionally permit unauthorized machines to mount the drive and access data. The lockd process relates to NFS in that it is responsible for managing locks on NFS files. Since Internet hosts should not be so openly trusted, these protocols should be blocked.

Firewall Rule:

Refer to rule #15 in the rulebase. The service group rpc_nfs_svcs contains: Portmap/rpcbind (111/tcp and 111/udp), NFS (2049/tcp and 2049/udp), lockd (4045/tcp and 4045/udp)

Testing:

Follow the same nmap, Firewall log checking, and actual application/protocol testing methodology to ensure that all of these kinds of traffic are blocked correctly. If it looks like data got through the firewall to a target host, that host's logs can also be checked or a "sniffer" type program (tcpdump/windump) can be used for packet capture verification.

4. NetBIOS Services

Original Recommendation:

NetBIOS in Windows NT -- 135 (tcp and udp), 137 (udp), 138 (udp), 139 (tcp). Windows 2000 – earlier ports plus 445(tcp and udp)

Vulnerability:

There are a large number of tools and exploits directed against various Windows platforms. Some of the tools are used to get system information from the registry or obtain user login/password data, while others will cause Denial of Service (DoS). Other security problems include file system shares that are unsecured and available to the Internet. Most of these vulnerabilities also have their associated tools that are quick, efficient, and simple to use. The good news is that blocking port 135-139 (tcp and udp) will prevent Internet users from successfully launching the vast majority of these attacks. In addition to the previous information the Microsoft-DS ports 445 (tcp/udp) for Windows 2000 should also be blocked for similar reasons as LDAP. Internet hosts should not be trying to connect to internal Directory Services.

Note: Chapter 5 of the text "Hacking Exposed by McClure, Scambray, and Kurtz" provides a great deal of information on the vulnerabilities, exploits, and tools used against the Windows NT operating system. Appendix B in the same text provides some information on Windows 2000 vulnerabilities. (http://www.hackingexposed.com/)

Firewall Rule:

Refer to rule #16 in the rulebase. The service group netbios_svcs contains: 135 (tcp and udp), 137 (udp), 138 (udp), 139 (tcp). Windows 2000 – earlier ports plus 445(tcp and udp). Note: Do NOT assume that the NBT group contains all of these ports or that all of these services even exist in the default software! Some service objects have to be created.

Testing:

Follow the same nmap, Firewall log checking, and actual application/protocol testing methodology to ensure that all of these kinds of traffic are blocked correctly. If it looks like data got through the firewall to a target host, that host's logs can also be checked or a "sniffer" type program (tcpdump/windump) can be used for packet capture verification.

5. X-Windows Services

Original Recommendation:

X Windows -- 6000/tcp through 6255/tcp

Vulnerability:

X-Windows systems provides a UNIX user with an attractive GUI interface to a system. If a site has chosen to restrict terminal based access originating from the Internet, then X-Windows should definitely be eliminated. A very clear, succinct description of what an attacker could do with X comes directly from " Building Internet Firewalls by Chapman and Zwicky (O'Reilly publications) p. 314." To paraphrase some information from this page: "There are a number of things an attacker can do with access to an X11 server, including: Getting screen dumps, Reading keystrokes, and Injecting Keystrokes as if they were typed by the user. " The possible ramifications of this could be as damaging as allowing an Internet user to have a root session on a UNIX server which is nothing short of full system compromise.

Firewall Rule:

Refer to rule #19 in the rulebase. The service group Xwindows: contains: port 6000-6250/tcp.
Note: The default X11 service does only spans 6000 – 6063, so create a new service.

Testing:

Follow the same nmap, Firewall log checking, and actual application/protocol testing methodology to ensure that X-Windows traffic is blocked correctly. If it looks like data got through the firewall to a target host, that host's logs can also be checked or a "sniffer" type program (tcpdump/windump) can be used for packet capture verification.

6. Naming Services

Original Recommendation:

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:

The main point in locking down DNS and LDAP is that these services can provide a hostile entity with quick, concise network, system, and user information about your environment. For sites not protecting TCP 53, a full zone-transfer could provide the IP addresses of all key systems and perhaps provide much more information if those systems have names like: "winntpdc", or "payroll." The recommendation here is to put the external DNS system on the screened subnet and the internal DNS behind the firewall on the trusted network and then protect the systems by adding rules to limit TCP/UDP 53.
Note: One could also use a split DNS system, where the external DNS has only a minimum of necessary entries for needed services and the internal DNS (the one inside the Firewall) provides the internal systems with name resolution. (Good DNS FAQ: http://www.acmebw.com/askmr.htm)

The Lightweight Directory Access Protocol is used with Microsoft's Windows 2000 Active Directory architecture and other systems. Since Active Directory represents a unified object

repository, significant user information can be quickly obtained by queries to this system. Denying any LDAP queries through the firewall limits the queries to being made by systems behind the firewall.

Firewall Rule:

Allowing DNS queries against non-DNS servers provides no practical use and so should be locked down to the proper DNS systems. Specifically queries from the Internet should only be allowed to hit the external DNS in the screened subnet. Internal queries should use the dns_int (internal dns) and so are restricted from the external dns (dns_ext).
Refer to rule #7 in the rulebase.

To prevent hostile users from obtaining the network roadmap of our internal systems, TCP 53, the port used for zone-xfer, is allowed only from the external DNS server to the internal DNS server.
Refer to rule #8 in the rulebase.

The internal DNS is permitted to do outbound 53/udp queries by rule #9.

LDAP traffic through the firewall is also denied.
Refer to rule #19 in the rulebase. The LDAP service (tcp and udp) is denied: group is ldap_group.

Testing:

To test the DNS rules, try using nslookup's functionality from various systems to ensure that the internal DNS works correctly and to ensure that the only valid zone transfer request crossing the firewall is from the Backup DNS to the Primary DNS. Similarly, use nmap to test TCP and UDP 389 to ensure that the firewall is dropping them. Verify all of this by checking the firewall's log.

7. Mail Services

Original Recommendation:

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

Vulnerability:

An over-riding theme when creating a firewall policy should be the concept of "allowing a system to do only as much as that system *should* be doing." In other words, there is no need to allow SMTP traffic to any device other than the site's appropriate mail server(s). Similarly, POP and IMAP should not be allowed to any system if these remote mail services are not permitted by the corporate security policy.

Blocking POP-2, POP-3, and IMAP from crossing the firewall (even from hitting the screened subnet) does, however, prevent remote users from using their Netscape mail and Outlook mail type clients, which could be a significant business issue. The risks of allowing

remote POP/IMAP access, which send their passwords in the clear as well as having exploits that can gain root access issue (http://www.cert.org/advisories/CA-97.09.imap_pop.html), versus using other methods is a policy issue involving a risk/benefit analysis. In this case, the recommendation is to block POP and IMAP, so other provisions must be made and the policy modified to support them. Some examples of alternatives are: using a dial in (RAS) server for any remote users, allowing users to set up an SSH port forwarded "tunnel" to the mail server, or implementing S/MIME (http://www.rsasecurity.com/standards/smime/faq.html).

Firewall Rule:

Refer to rules #5 and 6 in the rulebase. The SMTP service has been limited to the mail server smtp_mailbox.

Refer to rule #19 in the rulebase. Other mail protocols have been denied (POP (109/tcp and 110/tcp), IMAP (143/tcp)) using service group getmail_group

Testing:

Follow the same nmap, Firewall log checking, and actual application/protocol testing methodology to ensure that all of these mail services are blocked correctly. If it looks like data got through the firewall to a target system, that device's logs can also be checked or a "sniffer" type program (tcpdump/windump) can be used for packet capture verification.

8. Web Services

Original Recommendation:

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:

Using a similar design philosophy as that used to help secure email access, Web access on ports 80/tcp and 443/tcp, should be limited to the appropriate web servers. Limiting access to the correct Web servers prevents remote users from accessing un-intentional web servers (i.e.: those systems running IIS or Apache type servers without their knowledge). Further, eliminating the other common high ports and limiting web services to ports 80 or 443 would prevent having to open holes in the firewall on any tcp-high ports.
Note that limiting port and destination access to just the web servers does not prevent exploiting the web server applications. Web servers can still be exploited by various techniques, such as cgi script exploits. (http://infosec.navy.mil/tip11.html or search: http://www.cert.org/nav/alerts.html)

Firewall Rule:

Outbound web access from local_net is allowed by rule #3.

For inbound web access refer to rule #4 in the rulebase. The web services (http/https) have been limited to the web server on the screened subnet. The other high ports are denied by the generic "ANY, ANY, tcp-high-ports" rule near the bottom of the rule base.

Testing:

Follow the same nmap, Firewall log checking, and actual application/protocol testing methodology to ensure that the web traffic has been correctly restricted. If it looks like data erroneously got through the firewall to a target host, that host's logs can also be checked or a "sniffer" type program (tcpdump/windump) can be used for packet capture verification.

9. Small Services

Original Recommendation:

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

Vulnerability:

Many of the small services, those below 20, no longer play a practical part in the current network environment. Some of the protocols can consume significant resources (such as using a spoofed packet to start a port 7 echo and a port 19 chargen "dialogue") if abused.

Time protocol should not be accepted from an untrusted source, such as an Internet host. If time synching of devices is required, it should be provided as an internal service. Aside from the value of having all logs on time synched machines being consistent, accurate time services are also required for certain authentication services such as RSA's SecurID systems.

Firewall Rule:

Refer to rule #19 in the rulebase. In order to block the small services and Time, simply create a group (called the smallsvc_time_group) and include all the TCP and UDP services below 20 and then add 37/tcp and 37/udp.

Testing:

Follow the same nmap, Firewall log checking, and actual application/protocol testing methodology to ensure that all of these kinds of traffic are blocked correctly. If it looks like data got through the firewall to a target host, that host's logs can also be checked or a "sniffer" type program (tcpdump/windump) can be used for packet capture verification.

10. Miscellaneous Services

Original Recommendation:

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:

Many of the miscellaneous services listed above should be blocked for the same reasons mentioned in previous sections. Most of these services would provide remote, "untrusted" users with potentially significant information about the workings of the internal network, user accounts, and the ability to manipulate data. Unless there is a compelling business need these services should be blocked or locked down to specific hosts.

- TFTP is a less secure protocol, similar to FTP. TFTP should be blocked for much the same reasons as FTP, plus TFTP is even less secure as no authentication is needed to transfer data.
- Finger provides user account information (including whether the user is currently active or not) as well as possible contact information. Information from finger could be used as a starting point to guessing account login/password combinations and contact information would be useful for "social engineering" type attacks.
- NNTP (119/tcp), NTP (123/tcp), and LPD (515/tcp) should also not be crossing the firewall. If the site has their own internal Network Newsgroup (NNTP), outside Internet users should not be able to "snoop" on the internal messages. Outbound NNTP could be enabled if it was determined to serve a valid business need. NTP across the firewall should be blocked for the same reason as TIME. LPD, which provides remote printing, might be required for internal devices, but should not be allowed across the firewall.
- The syslog(514/udp) messages that devices send can provide very useful troubleshooting and event reconstruction information. The integrity of these log files is paramount if the information is to be trusted. The only devices that should be sending syslog messages are either local devices or possibly those on the screened subnet. The syslog service should be blocked and then opened up to specific devices on the screened subnet only if there is a business need to do so.
- SNMP can be a very effective network management tool because it can provide significant information on the configuration of systems and network components. Comparably, if this information were available to all users, it would prove to be a large potential threat. SNMP uses clear text community strings that are similar to passwords, as authentication. Some systems will allow SNMP set commands to reboot them or to change various settings, while SNMP traps provide useful messages. The potential for misuse is great. SNMP should not allowed across the firewall, unless there is a business need that requires very specific devices to talk to other specified devices. SNMP NOTE: To add useful network management functionality, SNMP access has been allowed from a single Network Management box (netmgmt) to the screened_net for queries (snmp_161 service group: 161/tcp and udp) and snmp-traps (snmp_traps group: 162 udp/tcp) are allowed from the screened_net back to only the netmgmt system.
- BGP is a routing protocol. Since the router will be handling connectivity to the internet, the firewall should not be allowing any BGP packets through it.

- SOCKS is used to Proxy different kinds of data (FTP, Telnet, etc). If the site is not running any SOCKS servers, the service should be denied as it serves no legitimate business need. If SOCKS proxies are being used, the firewall rules should be tightened down to specify the source and destination systems as explicitly as possible.

Firewall Rule:

Refer to rule #17, 18, and 20 in the rulebase. The SNMP services are allowed by rules 17 and 18 and then the rest are dropped by rule 20.

Testing:

Follow the same nmap, Firewall log checking, and actual application/protocol testing methodology to ensure that all of these kinds of traffic are blocked correctly. If it looks like data got through the firewall to a target host, that host's logs can also be checked or a "sniffer" type program (tcpdump/windump) can be used for packet capture verification.

11. ICMP Services

Original Recommendation:

ICMP-- block incoming echo request (ping and Windows traceroute), block outgoing echo replies, time exceeded, and unreachable messages

Vulnerability:

ICMP is often used to assist with network mapping. Echo replies can also be used as covert channels for some tools (i.e.: Loki). The less information interior devices provide back to the Internet the better. Thus, incoming pings should be blocked as well as responses to traceroute (time exceeded), echo replies (possible covert channel), and unreachable messages (provides port/service information).

Firewall Rule:

Refer to rules #10-13 in the rulebase for the ICMP deny rules. Note that the traceroute service (there by default in FW-1) was also added to the denied services list.
Policy note: the current configuration allows a subset of users (secureadmins) in the local_net access to the screened_net using ICMP tools (icmp_subset contains items like echo-request and traceroute) as well as responses from screened_net back to the admins (icmp_reply_subset contains items like echo-reply, dest-unreachable, time-exceeded, etc). Based upon business need, these rules should be modified to suite the current security policy.

Testing:

The ICMP blocking can be tested by using ping and traceroute (tracert) from an Internet device against protected devices. The Internet device can also try to connect to a service on a protected device that is known to not be supported and then verify that the response was blocked by the Firewall. Tools can be used to craft packets to generate and test icmp-reply packets.

Useful References:

Bibliographical Sources:

Chapman, D. Brent and Zwicky, Elizabeth D.; Building Internet Firewalls. O'Reilly publishing, September 1995.

McClure, Scambray, and Kurtz; Hacking Exposed. Osborne Publishing, 1999.

Additional Web Links:

Registered Port Numbers:
http://www.isi.edu/in-notes/iana/assignments/port-numbers

Trojan Port Numbers:
http://www.nohack.net/ports.html#trojan

DNS Reference:
http://www.acmebw.com/askmr.htm

Security Tools:
http://www.insecure.org/nmap/
http://www.securityfocus.com/
http://www.hackingexposed.com/

Misc. Security Sites:
http://www.cert.org/nav/alerts.html

Useful Checkpoint Information:
http://www.phoneboy.com/fw1/
http://msgs.securepoint.com/fw1/
http://www.geek-speak.net/fw1/fw1_properties.html

Windump:
http://netgroup-serv.polito.it/windump/

RFC Search:
http://sunsite.cnlab-switch.ch/cgi-bin/search/standard/nph-findstd?show_about=yes