FireWall

Application Notes

Table of Contents

Part No. 950-1142B

Printed in U.S.A. 3/95

Section 1 — How to Build a Firewall

Intended Audience

These instructions are intended for use by experienced system administrators who are familiar with firewall concepts but don't have time to research all the details themselves. It is not intended as a tutorial, an entire book would be needed for that and as soon as one is written we'll mention it prominently. Nor is this intended to be an exhaustive discussion of the pros and cons of various kinds of firewalls, for that you should read Bellovin & Cheswick's *Firewalls and Internet Security.*

If you are unfamiliar with firewalls and network security we strongly suggest that you consider hiring a consultant to do the work for you, the payback in peace of mind alone is worth it.

The following instructions are only one example of how to configure a firewall; you can improve the level of security by adding yet more monitoring or reducing services, you can decrease the level of security by adding more services.

The latest version of this document is available via anonymous FTP in `ftp.livingston.com:/pub/firewall`, you may wish to check if it's been updated before following the instructions here.

Livingston Enterprises provides the following information for the convenience of our customers and anyone else who wishes to read it, and accepts no liability whatsoever for your use or misuse of the information provided in this document.

If you find any mistakes or errors, or have suggestions on how to improve this document, please email them to `firewall-doc@livingston.com`.

Intended Platform

These instructions assume the use of an FireWall IRX-211 router and a Sun Sparcstation running SunOS 4.1.3_U1 for the bastion host (sometimes called an applications gateway). BSD/OS and Linux are also popular choices for bastion host, with the advantage that their source code is readily available. Most of the following instructions can be applied to any host with appropriate adjustments. It's typically a good idea to use a type of host you're very familiar with as your bastion.

In our following examples, the IRX's WAN port (S1) is a 56k or T1 link to the Internet, the first ethernet port (ether0) is connected to your internal network that needs to be protected (this will be called the "internal network"), and the second ethernet port (ether1) is connected to a minimal ethernet with only the bastion host on it; this network is called the "buffer network" or "DMZ". The Internet only talks to the bastion host on the buffer network, the bastion host in turn talks to your internal network.

The advantage of placing the bastion host on an ethernet separate from the rest of your hosts is that if it *is* broken into, it can't be used to snoop your internal network for passwords or other useful information, and it cannot be used to spoof other addresses on your internal network. You can also use multiple bastion hosts to provide load balancing and redundant reliability; setup for that is outside the scope of this document.

The IRX and your bastion host should be in a physically secure location.

Addressing

If you have a class B network address with a 24-bit subnet mask, you should use one of your subnets for the buffer network, e.g. 172.16.1.0. If you have a class C network it can be painful using half your network address space for the buffer network with a 25-bit subnet mask, e.g. 192.168.1.0 and 192.168.1.128 as your two networks. Instead you might ask your service provider to provide you with a tiny subnet of his network (with either a 29 or 30 bit subnet mask) for use by your buffer network.

For the purposes of illustration below we are going to use the following hostnames and addresses; you should adjust them to fit your usage.

```
172.16.1.1  irx1     # ether1 on the IRX-211
172.16.1.2  bastion  # address for your bastion host
172.16.2.1  irx2     # ether0 on the IRX-211, netmask 255.255.255.0
172.16.2.2  mailhub  # your internal mailhub that the bastion host speaks to
172.16.2.3  loghost  # where you want logs to go; can be the same
                     # as mailhub but must not be bastion
```

Setting up the FireWall IRX-211

Ideally, port s0 should be connected to a terminal that can act as a console for configuration of the IRX. You may wish to disable telnet access to the IRX and perform all configuration from the console, as that would be most secure. Next most secure is to filter out telnet access from the internet or bastion host and perform all your configuration from some internal host or network. If you believe there is any chance at all that your internal networks can be snooped by unfriendly users you *must* avoid the use of telnet or pmconsole for configuration of the IRX, and configure it *only* from the console.

The IRX silently discards source-routed packets, but in our example we'll explicitly discard any packet that claims to be from an address that couldn't possibly be coming in through that interface.

Configure Interfaces

Configure ether0, ether1 and s1 on the IRX, and set up your filter rules. Here is an example for a case where s1 is speaking PPP over a leased line to a network service provider at 192.168.1.1.
Note that the IRX always wants external clock on synchronous ports.

```
set password XXXXXXXX      # fill in your own choice for password
set telnet 0               # disable telnet administration port
set loghost 172.16.2.3
set gateway 192.168.1.1

set ether0 address 172.16.2.1
set ether0 netmask 255.255.255.0
set ether0 routing off
set ether1 address 172.16.1.1
set ether1 netmask 255.255.255.0
set ether1 routing off
set s1 network hardwire
set s1 address 0.0.0.0
set s1 destination 192.168.1.1 255.255.255.0
set s1 protocol ppp
set s1 routing broadcast   # or off, depending on what provider wants
set s1 mtu 1500
save all
reset s1
```

Set a Filter on Incoming Internet Traffic on S1

You can set filters on incoming packets and/or outgoing packets on each interface (port or ethernet). Filtering incoming packets is safer than filtering outgoing packets, because 1) you know which interface that packet is coming in on, and 2) you can protect the router itself with the filter.

This example is for a hardwired network interface on port s1. If you use "dial on demand" you should instead add the packet filter to the appropriate location and netuser table entries. For example, if you had a switched 56k link to an Internet Service Provider using a location table entry of "isp" then you would use the command "`set location isp ifilter internet.in`" in place of the command "`set s1 ifilter internet.in`".

This example allows any kind of outgoing connection from the bastion host, blocks all incoming traffic to any host but the bastion, and allows the following incoming traffic to the bastion: SMTP, NNTP, DNS, and FTP. Note that unless you have the latest versions of ftpd and sendmail you may be vulnerable to attacks through those ports. Later in this document we will describe where to get and how to install the latest versions of DNS (BIND 4.9.2) and SMTP (sendmail 8.6.10).

The rules are applied in the order given, and you can either permit or deny. Anything not permitted is denied at the end.

The name "bastion" below must be replaced by the IP address of your actual bastion host. The log keywords are optional. Rules 9 and on are optional and can be omitted if you don't want to see who and what is trying to get in through the firewall. WARNING! Make sure you have enough log space on your loghost to avoid denial of service attacks caused by too many connection attempts. You may wish to change rule 8 to only permit your secondary nameservers to do a zone transfer from your DNS server, but it's usually easier to use the "xfrnets" feature in `/etc/named.boot` to restrict that.

```
add filter internet.in
set filter internet.in 1 deny 172.16.0.0/16 0.0.0.0/0 log
set filter internet.in 2 permit 0.0.0.0/0 bastion/32 tcp established
set filter internet.in 3 permit 0.0.0.0/0 bastion/32 tcp dst eq 21 log
set filter internet.in 4 permit 0.0.0.0/0 bastion/32 tcp src eq 20 dst gt 1023
         log
set filter internet.in 5 permit 0.0.0.0/0 bastion/32 tcp dst eq 119 log
set filter internet.in 6 permit 0.0.0.0/0 bastion/32 tcp dst eq 25 log
set filter internet.in 7 permit 0.0.0.0/0 bastion/32 udp dst eq 53
set filter internet.in 8 permit 0.0.0.0/0 bastion/32 tcp dst eq 53 log
set filter internet.in 9 deny 0.0.0.0/0 bastion/32 tcp dst eq 113
set filter internet.in 10 deny 0.0.0.0/0 0.0.0.0/0 log
set s1 ifilter internet.in
```

```
save all
reset s1
```

1. Deny any attempt to spoof your IP addresses from the outside.
2. Allow established TCP connections to the bastion in (you may want to limit this further by putting deny commands ahead of it).
3. Allow anyone to FTP to the bastion (only if its serving as an anonymous FTP server, otherwise omit this rule).
4. Allow bastion to FTP things from the Internet (this carries some risk).
5. Allow incoming news (NNTP) to the bastion (only if you get news).
6. Allow incoming mail (SMTP) to the bastion.
7. Allow Domain Name service queries to the bastion.
8. Allow Domain Name service zone transfers from the bastion to others.
9. Deny attempts to contact authd, but don't log them.
10. Deny everything else, and log the attempt.

Set a Filter on Incoming Traffic on ether1

```
add filter ether1.in
set filter ether1.in  1 permit bastion/32 0.0.0.0/0 tcp established
set filter ether1.in  2 deny   bastion/32 172.16.0.0/16 tcp dst eq 21 log
set filter ether1.in  3 permit bastion/32 0.0.0.0/0 tcp dst eq 21 log
set filter ether1.in  4 permit bastion/32 0.0.0.0/0 tcp src eq 20 dst gt 1023
         log
set filter ether1.in  5 permit bastion/32 0.0.0.0/0 tcp dst eq 119 log
set filter ether1.in  6 permit bastion/32 172.16.2.2/32 tcp dst eq 25 log
set filter ether1.in  7 deny   bastion/32 172.16.0.0/16 tcp dst eq 25 log
set filter ether1.in  8 permit bastion/32 0.0.0.0/0 tcp dst eq 25 log
set filter ether1.in  9 permit bastion/32 0.0.0.0/0 udp dst eq 53
set filter ether1.in 10 permit bastion/32 0.0.0.0/0 tcp dst eq 53 log
set filter ether1.in 11 permit bastion/32 172.16.2.3/32 udp dst eq 514
set filter ether1.in 12 deny   bastion/32 172.16.0.0/16 log
set filter ether1.in 13 permit bastion/32 0.0.0.0/0 tcp log
set filter ether1.in 14 deny 0.0.0.0/0 0.0.0.0/0 log

set ether1 ifilter ether1.in
save all
```

1. Allow established TCP connections to the bastion in (you may want to limit this further by putting deny commands ahead of it).
2. Deny attempts by the bastion to FTP into your internal network.
3. Allow bastion to FTP to anywhere else.
4. Allow bastion to respond to FTP get/put commands (if anonymous ftp desired).
5. Allow outgoing news (NNTP) from the bastion (only if you get news).
6. Allow bastion to forward mail (SMTP) to mailhub.
7. Don't allow bastion to send mail to any other internal hosts.
8. Allow bastion to send mail to anywhere else.
9. Allow Domain Name service queries from the bastion to anywhere.
10. Allow Domain Name service zone transfers and log them.

NOTE that if you want to block the DNS from querying your. internal Domain Name servers that you would put deny rules. in front of rules 9 and 10 similar to what was done in rule 11.

11. Allow bastion host to syslog to loghost.
12. Don't allow bastion any other form of access to your internal. network; this is a safety measure in case the bastion is broken into.
13. Allow the bastion host to open up any other kind of TCP connection to the outside world; for greater security delete this rule.
14. There shouldn't be any traffic from ether1 that's not from. the bastion host; if you see any deny it and log it!

Set a Filter on Incoming Traffic From ether0

```
add filter ether0.in
set filter ether0.in 1 permit 172.16.0.0/16 bastion/32 tcp established
set filter ether0.in 2 permit 172.16.0.0/16 bastion/32 tcp log
set filter ether0.in 3 permit 172.16.0.0/16 bastion/32 udp dst eq 53
set filter ether0.in 4 deny 0.0.0.0/0 0.0.0.0/0 log

set ether0 ifilter ether0.in
save all
```

1. Allow any established TCP connections between internal net and bastion.
2. Allow any TCP connection from internal net to bastion, but log it.
3. Allow internal hosts to query Domain Name server on bastion (delete this rule if you're keeping internal and external DNS separate).
4. Deny anything else and log it.

Section 2 — Bastion Host

Setting up the Bastion Host

This example is for a Sun Sparcstation running SunOS 4.1.3_U1 but similar principles may be applied to any host.

If you have an internal host running the same operating system you may wish to build the restricted kernel and compile any programs such as sendmail there instead of on the bastion; in that case you can omit loading the sys, shlib and debugging load sets on the bastion host, but you must then patch your internal host the same as your bastion host, and assure its security. A compromise approach is to go ahead and load sys, shlib and debugging on the bastion host, configure everything, then tar `/usr/kvm/sys, /usr/lib/shlib.etc` and `/usr/lib/compile` to a tape and delete them from the system. That way if new patches come out from Sun you can update easily.

It is recommended that the bastion host use a dumb terminal for its console instead of a graphics head, and the following commands assume that. There's no sense in wasting a graphics head on a machine that's never going to need it.

To avoid denial of service attacks, `/tmp`, `/var/tmp` and `/var/spool/ftp` should not share a partition with any of `/, /usr, /var/spool/mqueue` or `/var/log` (if you log locally). SCSI disks are cheap, leave yourself plenty of room.

Here's one example of how to lay out a 400 MB disk, with `/var/spool/ftp` on sd1c.

Device	Mount	Size
sd0a	/	32 MB (not less than 8!)
sd0b	swap	32 MB (or 2x memory, whichever larger)
sd0d	/usr	128 (not less than 64!)
sd0e	/tmp	16
sd0f	/var	144
sd0g	/var/tmp	32

If sd0 is a 1GB disk you can place a 600MB `/var/spool/ftp` in sd0h.

ALL CONFIGURATION SHOULD BE DONE WITH THE BASTION HOST NOT ON THE NETWORK YET.
Make sure it's locked down tight before anyone gets a crack at it.

Acknowledgments

Much of this section is adapted from a more general article on "How to improve security on a newly installed SunOS 4.1.3 system" version 1.0, 7/94, by Thomas Kroeger, `tmk@uhunix.uhcc.hawaii.edu`, `ftp.hawaii.edu:/pub/security/docs`. This section also includes considerable information gleaned from `comp.unix.security` and the `firewalls` mailing list (see Section 9 — Other Resources).

Install SunOS

Install the following sets from 4.1.3_U1:
root
usr
Kvm
Install
Networking
System_V (only for m4, omit if you're using GNU m4)
Sys (omit if you have another host to configure a kernel on)
Debugging (omit if you have another host to compile on)
Shlib_Custom (omit if you have another host to rebuild libc.so on)
Security

The following should not be loaded:
SunView_Users
SunView_Demo
Text
Demo
OpenWindows_Users
OpenWindows_Demo
OpenWindows_Fonts
User_Diag
Manual
TLI
RFS
SunView_Programmers
Graphics
uucp
Games
Versatec
OpenWindows_Programmers

Install Patches

Obtain the following security patches and apply them. Check the README for each patch carefully to make sure it applies in your case. If you have a Sun support contract you can get them from your local Sun answer centers or SunSolve Online, otherwise they may be available via anonymous FTP from `ftp.uu.net:/systems/sun/sun-dist` and `ftp.eu.net:~ftp/sun/fixes`. You should have no need for crypt on your bastion host so it's OK to use the international libc jumbo patch instead of the domestic libc jumbo patch.

If you're unwisely trying to use 4.1.3 instead of 4.1.3_U1 there is a much longer list of patches to apply, see Kroeger's paper.

If you're using 4.1.4 instead of 4.1.3_U1 you may not need these patches, or may need different patches.

100103-11 Script to change file permissions to a more secure mode
101436-02 bin/mail jumbo patch
101558-01 international libc jumbo patch
101579-01 Security problem with expreserve for Solaris 1.1.1
101587-01 security patch for mfree and icmp redirect
101621 Jumbo tty patch
101665-02 sendmail jumbo patch
101679-01 Breach of security using modload

You may not need patches 101436 and 101579 since you won't be using `/bin/mail` and `/usr/lib/expreserve`, remember to chmod u-s both of them (better yet, chmod 0000 both). You won't need 101665 if you get sendmail 8.6.10 and use it instead (STRONGLY recommended).

100103-11 uses cut, which wasn't loaded, so remove the 3 lines in 4.1secure.sh that need cut.
101558-01 should be applied in single user mode.
101587-01 has a typo; the command

```
mv   `arch`/ip_icmp.o   /sys/`arch -k`/OBJ
```

should be

```
mv   `arch -k`/ip_icmp.o   /sys/`arch -k`/OBJ
```

You may also want to apply:
101441 syslog messages get confused
101508 sun4m kernel jumbo patch (if you're on a sun4m)
101775 cron consumes more memory as time goes on

It's a good idea to keep track of which patches you've applied, which you can do by:

```
grep Synopsis: 10*/README | sed -e 's,R.*:,,' >> /etc/install/patch
```

Tighten Security on Bastion Host

Ensure all users in `/etc/passwd` have a password, and have as few users on the bastion as possible. Ideally only the firewall administrator(s) should have accounts on the bastion and each one should have a unique root account of their own. The password for root itself should be disabled (by placing an * in the password field).

Limiting the number of users on the bastion host is the single most important step you can take to improve its security!

Keep the number of people with root access to the bastion host to a minimum (2-3 is a good number) and give them each their own uid 0 account with well-chosen individual passwords. Do NOT use the same password on the bastion host as you've used on any other system!

The bastion host should not run NIS (Yellow Pages) or NFS.

Run `passwd` to set the password for root. Better yet, create a uid 0 account under another name and change root's password entry to "*" by editing `/etc/passwd` with `vipw`.

Edit `/.cshrc` and `/.profile` and remove "." from their path. Optionally, add `/usr/local/bin` at the end of their paths. Add "`umask 22`" to `/.profile`. Make sure any directories or binaries in root's path are writeable ONLY by root.

```
chmod 711  /etc /usr/etc  /var/spool
rm -f /.rhosts
rm -rf /usr/old /usr/openwin
chmod g-s /tmp
chmod 1777 /var/tmp
cd /var/spool; rm -rf lpd.lock lpd rwho uucp uucppublic
rm -rf /etc/uucp /usr/etc/yp /bin/sunview /bin/sunview1
mkdir /usr/local/bin
chown root /usr/bin /usr/ucb /etc /usr/etc /usr/local/bin
```

Clean up /etc as follows:

```
echo 172.16.1.1 > /etc/defaultrouter
```

Edit `/etc/inetd.conf` and comment out all services except telnet and possibly FTP.

Edit `/etc/rc` and make sure in.rwhod and /usr/lib/lpd are commented out and that umask is set to 022 near the top.

Edit `/etc/rc.local` and comment out portmap, the UNCONFIGURED block, domainname, the entire NIS section, keyserv, gtconfig, the diskless client time synchronization, routed, biod. Change the chmod 666 /etc/motd to chmod 644 /etc/motd and make SURE that the "rm -f /tmp/t1" is done before the dmesg in that set of commands. Comment out the unused devices for sky, gpconfig and dbconfig. Comment out ndbootd, exportfs, nfsd, rpc.mountd, rarpd, rpc.bootparamd, rpc.statd, rpc.lockd, rpc.pwdauthd, automount. After the ifconfig statement add the following line if you're broadcasting 1's on your buffer network:

```
ifconfig le0 broadcast 172.16.1.255 >/dev/null
```

Edit `/etc/passwd` and remove the sysdiag, sundiag and sync users, especially sync. Remove the line starting with "+".

Edit `/etc/group` and remove the line at the end starting with "+".

Replace the contents of `/etc/syslog.conf` with either the first line or both lines:

```
*.info   @loghost
*.warn   /var/log/syslog
```

Optionally, edit `/etc/ttytab` to remove the secure option from all entries, so that you cannot log in over the network directly as root.

Comment out everything in `/etc/aliases` and run `newaliases`; later on we'll be modifying sendmail to disallow local delivery.

```
cd /etc; chmod go-w aliases.* motd mtab remote state syslog.pid utmp
cd /etc; rm -f defaultdomain hosts.equiv exports hosts.lpd tmp
rmdir /etc/sm /etc/sm.bak
```

Edit `/etc/netmasks` if needed.

Clean up /dev as follows:

(adjust according to your mix of devices)

```
chmod 600 /dev/eeprom
cd /dev; rm -f nit audio audioctl
cd /dev; rm cg* gp* vme* win*
cd /dev; rm {r,}sd[4-9][a-h]  {r,}sd[12][0-9][a-h] rmt* mt*
```

Clean up /usr/5bin as follows:

```
mv /usr/5bin/m4 /usr/bin
rm -rf /usr/5bin /usr/5include
```

Clean up /usr/bin as follows:

```
cd /usr/bin; rm -f yp* chfn chsh cu tip uudecode
cd /usr/bin; chmod g-s iostat ipcs wall write
```

Clean up /usr/etc as follows:

```
cd /usr/etc; rm -f etherfind nfsd biod nfsstat
cd /usr/etc; chmod 750 chill arp
cd /usr/etc; chmod g-s arp chill devinfo dkinfo dmesg dumpfs kgmon trpt
cd /usr/etc; chgrp operator dump; chmod 510 dump; chmod g+s dump
```

Clean up /usr/kvm as follows:

```
cd /usr/kvm; chmod g-s crash getcons pstat; chmod 750 pstat
chmod 700 /usr/kvm/crash
```

Clean up /usr/lib as follows:

```
cd /usr/lib; chmod u-s acct/accton expreserve exrecover
```

Clean up /usr/ucb as follows:

```
cd /usr/ucb; rm rcp rdist rlogin rsh rup ruptime rusers rwho talk
chmod u-s /usr/ucb/quota
cd /usr/ucb; chmod g-s netstat vmstat
```

Configure a restricted kernel

If you didn't load "sys" on this host you will need to build a kernel on another host and copy it over. If you do build sys on this host you may wish to archive `/usr/kvm/sys` and `/usr/include` to tape and delete them once you're through installing your firewall. At the least you should consider doing "`chmod 000 /usr/lib/compile`" to disable the compiler when you're not using it.

```
cd /sys
chmod -R go-w .
cd /sys/`arch -k`/conf
cp SDST60 BASTION
vi BASTION
```

delete options NFSCLIENT, NFSSERVER, PCFS
add the following 2 lines after the remaining options

```
options HSFS
options "IPFORWARDING=-1"
```

delete pseudo-device win128 dtop4 ms openeepr snit pf nbuf clone
delete device-driver bwtwo cgthree cgsix cgtwelve gt audioamd
append the following line to the very end:

```
disk sr0 at scsibus0 target 6 lun 0     # CD-ROM device
```

Save and exit vi.

```
config BASTION
cd ../BASTION
make
mv /vmunix /vmunix.old
mv vmunix /vmunix
chmod 440 /vmunix
chgrp kmem /vmunix
cd .. ; rm -rf BASTION
```

Reboot, and if the new kernel boots successfully, delete `/vmunix.old`.

Section 3 — DNS and BIND

Install BIND

Replace Sun's named with BIND 4.9.2, and rebuild the name resolver routines in Sun's shared library with BIND's.

BIND 4.9.2 is available via anonymous binary FTP from
`gatekeeper.dec.com:/pub/misc/vixie/4.9.2-940221.tar.gz`

You'll need GNU's gzip to uncompress it, so first build gzip.
You can FTP it from `ftp.uu.net:/archive/systems/gnu/gzip-1.2.4.tar`

```
tar xf gzip-1.2.4.tar
cd gzip-1.2.4
./configure
make
make install
cd ..
rm -rf gzip-1.2.4
```

Now you can extract BIND and build it.

```
mkdir bind
cd bind
/usr/local/bin/zcat ../4.9.2-940221.tar.gz | tar xf -
chmod u+w conf/options.h Makefile res/Makefile compat/lib/Makefile
chmod u+w compat/include/sys/Makefile
vi conf/options.h
```

Read README and OPTIONS; you can redefine LOGFAC to be LOCAL0-7 instead of LOG_DAEMON if you wish. If you plan to have your internal DNS query to the bastion host's DNS and you're forwarding NIS hostnames to DNS you may want to use YPKLUDGE.

In `conf/options.h` you should comment out NCACHE, VALIDATE and ADDAUTH, and turn on SECURE_ZONES.

In `compat/include/sys/Makefile` you should add bitypes.h to HFILES

```
cp bin/mkdep.append /usr/ucb/mkdep
vi Makefile
```

Edit the Makefile and uncomment sections after "(sunos4.x)" and "(ultrix, sunos, other 4.[23]bsd-alikes)". Remove "man" from the list of SUBDIRS and move “compat” ahead of “res”.

```
vi res/Makefile
```

add -pic to LOCDEFS and comment out the two ${LDS}
lines in the .c.o rule.

Before AROBJS add these lines:

```
ARGET= ../compat/lib/inet_addr.o  ../compat/lib/strerror.o  \
         ../compat/lib/strtoul.o
ARSUFF= inet_addr.o  strerror.o  strtoul.o
```

As first action for libresolv.a put "`cp ${ARGET} .`"
Save and exit vi.

Edit `compat/lib/Makefile` and add -pic to DEFS and comment out
the two ${LDS} lines.

```
vi compat/lib/Makefile
make depend
make all
make install
```

Copy the root nameserver cache to your nameserver data directory.

```
mkdir /usr/local/lib/named
cp conf/root.cache /usr/local/lib/named/cache.db
```

or FTP to `ftp.internic.net:<netinfo>root-servers.txt`
or to `rs.internic.net`

Configure DNS

Now configure your `/etc/named.boot` and `/etc/resolv.conf` files, kill `in.named` if it's running and start your new `in.named`. "xfrnets" can be used in `/etc/named.boot` to restrict which networks can perform zone transfers. O'Reilly & Associates publishes an excellent guide to *DNS and BIND*.

Sample /etc/named.boot:

```
directory       /usr/local/lib/named
;
; type          domain                  source host/file
;
primary         0.0.127.in-addr.arpa    127.0.0.zone
;
primary         example.com             example.zone
primary         16.172.in-addr.arpa     172.16.zone
;
; load the cache data last
cache           .                       cache.db
```

Sample /etc/resolv.conf:

```
domain example.com
localhost
```

Sample files in /usr/local/lib/named:

```
:::::::::::::
127.0.0.zone
:::::::::::::
; 0.0.127.in-addr.arpa
;
@       IN      SOA     bastion.example.com. root.bastion.example.com. (
                        1       ; Serial
                        7200    ; Refresh
                        1200    ; Retry
                        259200  ; Expire
                        7200 )  ; Minimum
            IN      NS      bastion.example.com.
; add your other nameservers here with IN NS lines
;
1               IN      PTR     localhost.example.com.
:::::::::::::
172.16.zone
:::::::::::::
;
; 16.172.in-addr.arpa
;
@               IN      SOA     bastion.example.com. root.bastion.example.com. (
                        4       ; Serial
                        21600   ; Refresh
                        900     ; Retry
                        172800  ; Expire
                        3600 )  ; Minimum
            IN      NS      bastion.example.com.
; alternate servers go here with IN NS records
;
1.1             IN      PTR     fw1.example.com.
2.1             IN      PTR     bastion.example.com.
1.2             IN      PTR     fw2.example.com.
```

```
:::::::::::::
example.zone
:::::::::::::
;
; .example.com
;
@               IN      SOA     bastion.example.com. root.example.com. (
                                8       ; Serial
                                21600   ; Refresh 6hours
                                900     ; Retry 15 minutes
                                172800; Expire 48 hours
                                3600 )  ; Minimum 1 hour
                                IN      NS              bastion.example.com.
; alternate nameservers go here with IN NS records
;
;
; anonymous ftp alias
ftp                             IN      CNAME           bastion.example.com.
;
; news alias
news                            IN      CNAME           bastion.example.com.
;
; name server alias
ns                              IN      CNAME           bastion.example.com.
;
; gateway server
gate                            IN      CNAME           bastion.example.com.
;
; for machines that can't hack MX records
example.com.                    IN      CNAME           bastion.example.com.
;
;
localhost   IN      A       127.0.0.1
;
;
; -- start real addresses --
;
bastion                 IN      A       172.16.1.2
                        IN      MX      10              bastion.example.com.
fw1                     IN      A       172.16.1.1
fw2                     IN      A       172.16.2.1
mailhub                 IN      A       172.16.2.2
loghost                 IN      A       172.16.2.3
; don't list your internal hosts
```

Section 4 — Shared Library (shlib)

Rebuild Shared Library

How to rebuild your shared library with the bind 4.9.2 libresolv so you don't need to run NIS on your bastion host.

PREREQUISITES: Before performing these tasks you should apply Patch 101558 in single user mode, and then build and install BIND 4.9.2 or later. Note that the BIND instructions specified the use of the -pic flag when compiling libresolv routines, this is why.

```
cd /usr/lib/shlib.etc
vi Makefile
```

and change the lines below to read as they do here:

```
 OBJSORT=./objsort
AWKFILE=./awkfile
```

and add the `-ldl` option at the end of both ld command lines.
Save and exit the editor

```
mkdir tmp
```

Change to the `tmp` directory just made, extract the pic .o from `libc_pic.a` and delete the file `__.SYMDEF`. The reason you need to do the 3 `mv` commands is because `ar` truncated filenames over 15 characters long.

```
cd tmp
ar x ../libc_pic.a
rm __.SYMDEF
mv rpc_dtablesize. rpc_dtablesize.o
mv rpc_commondata. rpc_commondata.o
mv xccs.multibyte. xccs.multibyte.o

rm gethostent.o
ar t /usr/lib/libresolv.a > /tmp/libresolv.toc
ar x /usr/lib/libresolv.a

cd ..
cp -p lorder-sparc lorder-sparc.FCS
```

```
vi /tmp/libresolv.toc
```

In `/tmp/libresolv.toc` delete the lines for __.SYMDEF, getnetent.o and inet_addr.o. Save and exit vi.

```
vi lorder-sparc
```

In `lorder-sparc` replace the line for gethostent.o with the contents of `/tmp/libresolv.toc`. Save and exit vi.

```
make libc.so
```

Now you should have `libc.so.1.9.1` built in the current directory. It is recommended that you test this library before installing it. You can do so by setting the environment variable LD_LIBRARY_PATH to the current directory. For example:

```
setenv LD_LIBRARY_PATH `pwd`
date
```

Once you are satisfied that the new library worked, you can proceed to install it with the following commands:

```
mv libc.so.1.9.1 /usr/lib
ldconfig
unsetenv LD_LIBRARY_PATH
```

You are now running with the new library. You can verify this by doing a trace command of let's say "`date`".

```
trace date |& grep libc
```

The output should informed you that the new library is being used.

Remove the temporary directory.

```
rm -rf tmp
```

Section 5 — sendmail

Overview

sendmail is both complicated and runs as root, a very bad combination. Cheswick & Bellovin's book talks about a better way of dealing with mail but haven't as far as I know provided source.

As of this writing, the latest version of sendmail is 8.6.10 available in source form via the Internet from `ftp.cs.berkeley.edu:/ucb/src/sendmail`
This version of sendmail is well-documented in the book *sendmail*, available from O'Reilly & Associates, Inc.
This is the *only* version of sendmail to have fixed all security holes known as of this date, so you should get it and run it instead of Sun's.

You should configure it to accept mail for your entire domain and pass it through to an internal mailhost. Likewise all your internal hosts should send mail to the internal mailhost, which will either deliver it locally or pass it to the bastion host if it needs to be sent out onto the Internet.

Further, you should disable delivery to the proc and local mailers on the bastion host and `chmod u-s` or `chmod 000 /bin/mail`.

Installing Sendmail 8.6.10

FTP to `ftp.cs.berkeley.edu` and
mget `/ucb/src/sendmail/sendmail.8.6.10.*.tar.Z`

```
mkdir sendmail-8.6.10
cd sendmail-8.6.10
zcat ../sendmail.8.6.10.base.tar.Z | tar xf -
cd src
cp Makefile.SunOS Makefile
chmod u+w Makefile conf.h
vi Makefile
```

In `Makefile`,
Change DBMDEF= to just -DNDBM,
Comment out INCDIRS and LIBDIRS,
Remove -ldb from LIBS,
Add -DIDENTPROTO=0 to ENVDEF=.
Save and exit vi.

```
vi conf.h
```

In `conf.h` comment out MATCHGECOS.
Save and exit vi.

```
make sendmail
rm /usr/lib/sendmail*
make install-sendmail
```

Configure Sendmail

```
zcat ../sendmail-8.6.10.cf.tar.Z | tar xf -
```

This step will vary the most; the following example assumes your domain is named example.com and that you want to pass mail for anything in example.com inwards to mailhub.example.com and pass all other mail out to the Internet, with no local delivery. It is especially important that you disable your prog mailer but it’s also useful to disable your local mailer so that no mail is ever actually delivered on your bastion host.

```
mkdir eg
cd eg
```

Create a file called eg.mc with the following contents, adjust to taste.

```
divert(-1)
include(`../m4/cf.m4')
VERSIONID(`@(#)eg.mc    1.1 (example.com) 7/13/94')
MASQUERADE_AS(example.com)
define(LOCAL_RELAY,mailhub.example.com)
define(MAIL_HUB,mailhub.example.com)
FEATURE(nouucp)
MAILER(smtp)
```

```
m4 < eg.mc >eg.cf
vi eg.cf
```

Edit eg.cf and replace both calls to "$#local $: $1" with "$#relay $@ $H $: $1" On the lines that say Mlocal and Mprog change /bin/mail and /bin/sh to /bin/false and on the lines after them change "mail -d $u" and "sh -c $u" to "false". Add any aliases you want to be known as to the line starting with "Cw". Delete the line "Tuucp". Save and exit vi.

```
mv eg.cf /etc/sendmail.cf
```

Kill and restart your sendmail daemon (or just have it take effect next time you reboot). You may wish to save a copy of your eg.mc file in `/usr/local/etc` or somewhere, or even tar up the entire cf setup and save it somewhere in case you need to tweak sendmail in the future.

Section 6 — FTP

Installing Anonymous FTP on a Sun Workstation

This section is adapted from the Sun Administration Frequently Asked Questions list, which is available via FTP from `ftp.uwtc.washington.edu` in `/pub/FAQs/Installing_Anonymous_FTP`

NOTE: warnings and special notes are set off by ***.
PLEASE READ ALL NOTES AND WARNINGS!!! In fact, it would be best to read the entire document through before beginning the installation, so you will be aware of all of the possible options.

1) Create the user ftp in `/etc/passwd`, and the group ftp in `/etc/group`. The user`s home directory will be `~ftp` where `~ftp` is the root you wish anonymous users to see. Ideally `~ftp` should have its own partition, especially if you permit uploads.
Use an invalid password and user shell for better security. The entry in the `/etc/passwd` file should look something like:

```
ftp:*:30:30:Anonymous FTP:/var/spool/ftp:/bin/true
```

The entry in the `/etc/group` file should look like:

```
ftp:*:30:
```

2) Create the home directory `~ftp`. Make the directory owned by root (NOT ftp) with the same group as ftp. Thus, owner permissions are for root and group permissions are for the anonymous users. Set the permissions for `~ftp` to 555 (read, no write, execute).

3) Create the directory `~ftp/bin`. This directory is owned by root (group wheel) with permissions 111.

4) Mount the Installation CD-ROM on `/mnt` and copy the statically-linked `ls` from `/mnt/sunupgrade/toolkit/ls` into `~ftp/bin/ls`. `ls` is owned by root with permissions 111 (no read, no write, execute).

5) Make the directory `~ftp/etc`. This directory is owned by root with permissions 111.

6) Create the files `passwd` and `group` in `~ftp/etc`. These files should be mode 444. The `passwd` file should only contain root and ftp. The `group` file hould contain ftp's group, ftp. Place a * in all the password fields, e.g.

```
ftp:*:30:30:Anonymous FTP::
```

For maximum security, do not use the `passwd` or `group` files at all. They are only required to provide the name of a file owner when users do "`ls -l`". Since all files and directories should be owned by ftp or root, this is useless.

7) Make the directory `~ftp/pub`. This directory is owned by root and has the same group as ftp with permissions 2555. Files are left here for public distribution. All folders inside `~ftp/pub` should have the same permissions 2555.

***Neither the home directory (`~ftp`) nor any directory below it should be owned by user ftp! Modern ftp daemons support all kinds of useful commands, such as chmod, that allow outsiders to undo your careful permission settings. (Thanks to Wietse Venema for that note!)

8) If you wish to have a place for anonymous users to leave files, create the directory `~ftp/pub/incoming`. This directory is owned by root with permissions 1720 (root has all permissions, group ftp can only write). Files can be left here, but users cannot see what is there, to prevent the spread of unauthorized files. Note that this is potentially hazardous, so you may want to skip step 8.

9) If you want to have the local time showing when people connect, create the directory `~ftp/usr/share/lib/zoneinfo` and copy `/usr/share/lib/zoneinfo/localtime` into it. All of these directories should be mode 111, owned by root. The file should be mode 444 owned by root.

Much of the information for the above is in the man page for ftpd. There are also wrappers and ftpd-replacements available which add the ability to control access, log accesses, and keep statistical records of file transfers. Two of these are available on `ftp.uwtc.washington.edu:/pub/Sun_Software/log_tcp_4.3.shar.Z` and `ftpd.wuarchive.tar.Z`

Section 7 — Proxy Telnet and FTP

Proxy Telnet and FTP

Having gotten this far, you're now safely tucked away behind your firewall, but your users will probably want to be able to telnet and FTP out to the Internet. The safest way of doing this is to set up a proxy gateway for telnet and FTP.

FTP to `ftp.tis.com:/pub/firewalls/toolkit` and get

```
DISCLAIMER
LICENSE
fwtk-doc-only.tar.Z
fwtk-v1.3.tar.Z
```

Then build fwtk.

```
zcat fwtk-v1.3.tar.Z | tar xf -
cd fwtk
make
make install
rm /usr/local/etc/rlogin-gw
```

Edit `/usr/local/etc/netperm-table` to establish the rules you prefer, and see `config/inetd.conf` for examples of how to modify your `/etc/inetd.conf` to use the proxies.

Section 8 — Tools

Other Tools

You may find the following tools useful as well.

```
ftp.uu.net:/archive/systems/gnu/gzip-1.2.4.tar (to unzip .gz files)
ftp.uu.net:/archive/systems/gnu/m4-1.4.tar.gz (for sendmail mc files)
ftp.uu.net:/archive/systems/gnu/perl-4.036.tar.gz (for cops)
ftp.uu.net:/usenet/comp.sources.misc/volume40/iss (Internet Security Scanner)

ftp.cert.org:/pub/tools/cops/1.04/cops_104.tar.Z
ftp.cert.org:/pub/tools/tcp_wrappers/tcp_wrapper_7.2.tar.Z
ftp.cert.org:/pub/tools/tripwire/tripwire-1.2.tar.Z

sierra.stanford.edu:/pub/sources/swatch-2.1.tar.gz
```

M4

If you want to use GNU m4 instead of `/usr/5bin/m4` then you don't need to load the sys5 compatibility package for SunOS. Instead you build and install GNU m4 in `/usr/local/bin` (or you can place it in `/usr/bin` if you don't want to have to remember to run `/usr/local/bin/m4` when processing sendmail .mc files).

```
/usr/local/bin/zcat m4-1.4.tar.gz | tar xf -
cd m4-1.4
./configure
vi Makefile
        edit Makefile and change CFLAGS from -g to -O
vi lib/Makefile
        edit lib/Makefile and change CFLAGS from -g to -O
        and add alloca.o to LIBOBJS
make
make check
make install
```

PERL

```
/usr/local/bin/zcat perl-4.036.tar.gz | tar xf -
cd perl-4.036
./Configure
make
make test
make install
cd ..
rm -rf perl-4.036
```

COPS

```
zcat cops_104.tar.Z | tar xf -
cd cops_104
```

If you've installed PERL, just do:

```
mkdir /usr/local/lib/cops
cp perl/* /usr/local/lib/cops
```

If you haven't installed PERL, instead edit `makefile` and change INSTALL_DIR to /usr/local/lib/cops.

```
vi makefile
make
make install
```

Section 9 — Other Resources

CERT Advisory

You can subscribe to the CERT (Computer Emergency Response Team) advisory mailing list by sending e-mail to `cert-advisory-request@cert.org` with "subscribe" in the header.

You may also wish to subscribe to `cert-tools-request@cert.org`.

Sun Security Bulletins

Sun Security Bulletins are available free of charge as part of their Customer Warning System. It is not necessary to have a Sun support contract in order to receive them.

To subscribe to this bulletin series, send mail to the address "`security-alert@Sun.COM`" with the subject "subscribe CWS [mail-address]" and a message body containing affiliation and contact information. To request that your name be removed from the mailing list, send mail to the same address with the subject "unsubscribe CWS [mail-address]". Do not include other requests or reports in a subscription message.

Due to the volume of subscription requests which they receive, they cannot guarantee to acknowledge or execute requests which are not in the format described above. Normally they will acknowledge your request within 24 hours of receipt.

TIS Firewall Mailing list

Trusted Information Systems maintains an electronic-mail users' group `fwall-users@tis.com` for discussion of their FWTK toolkit. To join, send electronic mail to `fwall-users-request@tis.com`. To find out about their commercial services based on the FWTK toolkit, send email to `netsec@tis.com` or call Fred Avolio at 301-854-6889, 3060 Washington Road Glenwood, MD 21738.

Books

DNS and BIND in a Nutshell, by Paul Albitz & Cricket Liu,
ISBN 1-56592-010-4, O'Reilly & Associates, Inc.
(800) 998-9938 and (707) 829-0515.

Firewalls and Internet Security – Repelling the Wily Hacker, by
William R. Cheswick and Steven M. Bellovin,
ISBN 0-201-63357-4, Addison-Wesley.
Errata are available on
`ftp.research.att.com:/dist/internet_security/firewall.book`

Internetworking with TCP/IP, by Douglas Comer,
ISBN 0-13-468505-9, Prentice-Hall, Inc.

P*ractical Unix Security*, Garfinkle and Eugene Spafford
ISBN 0-937175-72-2, O-Reilly & Associates, Inc.

sendmail, by Bryan Costales with Eric Allman & Neil Rickert,
ISBN 1-56592-056-2, O'Reilly & Associates, Inc.
Sendmail 8.6.10 is well-documented in this book.

FTP

Good places to look for firewalls information on the Internet are:

`ftp.greatcircle.com`: home of the firewalls mailing list.
To join, send email to `majordomo@greatcircle.com`
with “`subscribe firewalls`” in the body of the message.

```
research.att.com:/dist/internet_security
ftp.tis.com:/pub/firewalls
coast.cs.purdue.edu:/pub/aux
```

CERT Security Checklist

Using the checklist will help you identify security weaknesses or modifications to your systems. The CERT Security Checklist is based on information gained from computer security incidents reported to CERT. It is available via anonymous FTP from `info.cert.org:/pub/tech_tips/security_info`.

Security Tools

Use security tools such as COPS and Tripwire to check for security configuration weaknesses and for modifications made by intruders. We suggest storing these security tools, their configuration files, and databases offline. TCP daemon wrapper programs can provide additional logging and access control. These tools are available via anonymous FTP from CERT in the `info.cert.org:/pub/tools` directory.