

Inexpensive Firewalls



Simon Cooper <sc@sgi.com>

Lisa 1999

11 November 1999

<http://reality.sgi.com/sc/papers/lisa-1999.pdf>

- or -

<http://www.sfik.com/papers/lisa-1999.pdf>



A Firewall?



LISA 1999, Inexpensive Firewalls
Simon Cooper <sc@sgi.com>

What is an inexpensive Firewall?



- a specific use device
- an “all in one” firewall (filters + apps)
- uses readily available hardware
- uses an OS you are familiar with
- uses free or affordable tools



What an inexpensive firewall isn't...



- NOT a high performance firewall
- NOT a high reliability firewall
- NOT a maximum security firewall
- NOT a “no cost” firewall
- NOT a “plug and play” firewall



What are they good for?



- a departmental network
- a lab network
- a small business
- a home
- a personal domain



Agenda



- Ingredients
- Hardware
- OS
- Filtering and Services
- Administration
- Tips for building
- Experiences
- Q&A



Ingredients



- know what you want to run on or pass through your firewall
- old or cheap hardware
- a suitable and **familiar** operating system
- free or affordable tools
- your time



Know what you want to do



- who do you want to let in
- who do you want to try and keep out
- is it in alignment with your security policy
- what services will be offered



Hardware



- Use what you have
 - Suns, PCs¹, SGIs
- Laptops
 - Quiet, compact, built in UPS
- Last generation hardware
- Has two network interfaces

1. Don't re-use hardware your organization has rejected because of Y2K issues unless you can show it will continue to work.

LISA 1999, Inexpensive Firewalls
Simon Cooper <sc@sgi.com>



Hardware Issues



- Know which is the inside interface
 - choose the primary/first to be inside
- CDROM drive
 - check if it can read CD-R and CD-RW (this is worth a small investment)
- Power supplies, disks and fans wear out



Operating System



The operating system you use will need

- packet filtering
- free or affordable software for what you want to do
- to be **familiar** to you
- continued and active support
- an active security community
- Linux



Operating System Examples



- NT
- A BSD variant
- IRIX
- Solaris
- AIX



Hardening the OS



- Philosophy - disable/remove everything that is not needed
- Secure “distributions” exist
 - “freefire” (pointers)
 - Linux Router Project, picoBSD
- Can do it yourself
- Keep a written log. Write a script
- Don’t build your firewall on the network you are going to protect!



Hardening the OS



There are cheat sheets on the web for many OS. Search for keywords and combinations like

hardening, securing, bastion, <OS Name>

Sites with particular OS information seem to be on the increase - try searching there first.

Some “security news” sites carry articles on securing a specific OS.

Check the OS release with the information you find - don't completely rely on one information source.



The Kernel



Things to watch out for and protect against

- IP denial of service attacks
- IP forwarding off when system boots
- Packet filtering failure modes
- IP fragmentation - do re-assembly



Remote OS Logging



For unix

- syslog
 - some can be made send only
 - can send encrypted packets
 - use TCP rather than UDP

For NT

- A free syslog like tool, but simulates the behaviour. Not real time.



Checklist



Turn off all services you won't be using

Secure the file system

- update file permissions
- remove pieces you won't be using

Apply Kernel changes/patches

Run your initial integrity check now!



Filtering Topics



Desired features

What is available

ipfilterd

ipchains

Filtering issues

Example



Filtering: Desired Features



The wish list

- input & output rules for each interface
- interface forwarding rules
- ability to rewrite packets (masquerading)
- knowledge of ICMP, ability to rewrite
- logging of rejected or flagged packets
- hierarchical (user defined) rules

there is more...



Filtering: Desired Features (continued)



- handling of idle TCP sessions
- configurable handling of UDP
- detailed knowledge about some protocols (DNS, traceroute)
- configurable default policy



Filtering



No operating system has it all

- NT
- ipfilterd (IRIX, AIX), ipfilter (Solaris)
- ipchains (Linux, BSD Variants)



ipchains



What can it do

- in/out filters for each interface
- by protocol, port and addresses
- separate forwarding rules
- user defined rules
- support for packet rewriting (masquerading)
- understands ICMP packet types
- default policy



ipchains - weaknesses



- weak on logging
 - only logs a packet synopsis
- rules are built incrementally



Filtering Issues



- icmp path MTU discovery
- auth/identd - reject but allow a response
- REJECT or DROP
- protect yourself from mishaps
- don't assume inside is always inside



Example (input)



```
ipchains -F input
ipchains -P input reject
# Protect against IP address spoofing
ipchains -A input -i eth0 -s $inet -d $any -j ACCEPT
ipchains -A input -i eth1 -s $inet -d $any -l -j REJECT

# Allow incoming SMTP
ipchains -A input -i eth1 -p tcp -s $any -d $me 25 -j \
ACCEPT

# Catch all rule
ipchains -A input -s $any -d $any -l -j REJECT
```



Example (output)



```
# Protect against spoofing or routing errors
ipchains -F output
ipchains -P output REJECT
ipchains -A output -i eth0 -s $any -d $inet -j ACCEPT
ipchains -A output -i eth1 -s $any -d $inet -l -j REJECT

# Allow SMTP out
ipchains -A output -i eth1 -p tcp -s $me 25 -d $any -j \
ACCEPT

# Catch everything else
ipchains -A output -s $any -d $any -l -j REJECT
```



Resources for Creating Filters



Building Internet Firewalls,

2nd Edition, O'Reilly and Associates

By Elizabeth D. Zwicky, Simon Cooper, D. Brent Chapman

- Sometime in the 2nd quarter of 2000.
- Has handy tables of port numbers and details on the packets flow direction
- Bigger than before and includes information for NT

Linux HOWTOs for ipchains and masquerading



Services



Unix

- Mail - Postfix
- Web Proxy/Server - Apache
- Proxy - SOCKS
- Transparency/masquerading

NT

- Mail - Sendmail for NT (not free)
- Web Server - Apache for NT
- Proxy - Microsoft Proxy Server



Masquerading



How does it work?

- intercepts forwarded packets
- re-writes outgoing and return packets
- does it transparently
- can add dynamically loaded modules for “complicated” protocols



Masquerading Example



```
# Allow all non-blocked internal traffic to be
# masqueraded

ipchains -F forward
ipchains -P forward DENY
ipchains -A forward -i eth0 -s $inet -d $any -j MASQ
ipchains -A forward -s $any -d $any -l -j REJECT

# Allow direct SSH from external site to an internal
# system

ipmasqadm portfw -f
ipmasqadm portfw -a -P tcp -L $local 22 -R $internal 22
```



Administration



- ssh
- non-reusable passwords?
 - How about using a PDA



Administration



Be certain of the integrity of the system

- will save you time and worry
- use tripwire or equivalent
 - Can get tripwire for NT (commercial)
 - store the database on CD-R
 - under unix statically link the binary and store on the CD-R



Tips for Building



Use a CD-R or CD-RW. CD-RW can be used to get the process right

Recent hardware can boot directly from CD

Tools exist under unix to create bootable CDs

Use automated installation tools

- SGI RoboInst

Only connect your system to dangerous networks when you have finished building it



Experiences



Small company

- using a Sun IPX
- SOCKS + DNS
- Connects to the Internet via DSL

Personal Domain

- using a pre-built \$400 PC
- ipchains and masquerading
- Postfix, Web Server and DNS



Conclusion



You can build and run a firewall for those places that should have some protection but they have perhaps been overlooked because it was too expensive or time consuming to purchase and install a commercial firewall



Q/A



A copy of the slides are available at,

<http://reality.sgi.com/sc/papers/lisa-1999.pdf>

- or -

<http://www.sfik.com/papers/lisa-1999.pdf>



An Inexpensive Firewall



NOTE: Please do not test my firewall



LISA 1999, Inexpensive Firewalls
Simon Cooper <sc@sgi.com>