Introduction to Internet Firewalls

Darren Reed
(darrenr@cyber.com.au)

Cybersource Pty Ltd.

Introduction

- What is a Firewall ?
- Do Firewalls Work ?
- Do I need a Firewall ?

Agenda

- Starting Off
- Network/Security Policies
- What can Firewalls do ?
- Building a Firewall

Vocabulary

- Glossary of terms
  - Bastion
  - DMZ - De-Militarized Zone
  - Packet Filter
  - Proxy

Starting Off

- Why do people have firewalls ?
- What does it do ?
- What doesn’t it do ?
- What are you protecting ?
- What is the risk involved ?
- Your network/security policy.

Why do people have Firewalls ?

- Provide access to other networks
  - focal point for access to the Internet
- Security fence for your network
  - Hurdle for intruders to get over
- Help enforce network/security policies
  - prevent users from inadvertantly creating security problems
    - e.g. remote access via insecure lines.

What does it do ?

- Acts as a control point for:
  - people accessing your network from other networks
  - attackers trying to gain unauthorised access
  - people accessing other networks from your network

What doesn’t it do ?

- Doesn’t stop users doing silly things:
  - e.g. won’t stop someone walking out the door with confidential information on a floppy disk.
  - can’t prevent people from chosing bad passwords;
  - can’t prevent insiders from helping unauthorised outsiders get in.

Firewalls and Internal Security

- Doesn’t increase internal security:
  - doesn’t prevent users from using their PC’s to snoop internal network traffic;
  - doesn’t make all your Unix systems safe from all sendmail bugs;
  - doesn’t make NFS secure;
  - doesn’t stop people trying to break in;
  - doesn’t tell you who tried to break in;
  - can’t protect you against Viruses.

What are you protecting ?

- Internal fileservers:
  - valuable intellectual property;
  - business databases:
    - e.g. personnel records, balance sheets, customer records, etc.
- Operational systems
- Your reputation

Risk Involved

- What do you lose if someone breaks in ?
  - time ?
  - money ?
  - reputation ?
- Denial of service attacks
  - mail bombs

Network Security Policy

- Make sure you have one.
- Things to consider for your policy:
  - modems connected to PCs
    - e.g. allowing dialup PPP to a desk
  - using software from the Internet on your network
    - beware of trojans, viruses, etc

No policy means what ?

- Hard to define your Firewall
- Very hard to take action against insiders who undertake risky behaviour
- Hard to justify any expense

Creating a Security Policy

- What are you protecting ?
- What level of risk is acceptable to you ?
- What happens when there is an incident ?
  - Call CERT, police, other officials ?
  - Take your network “offline” ?

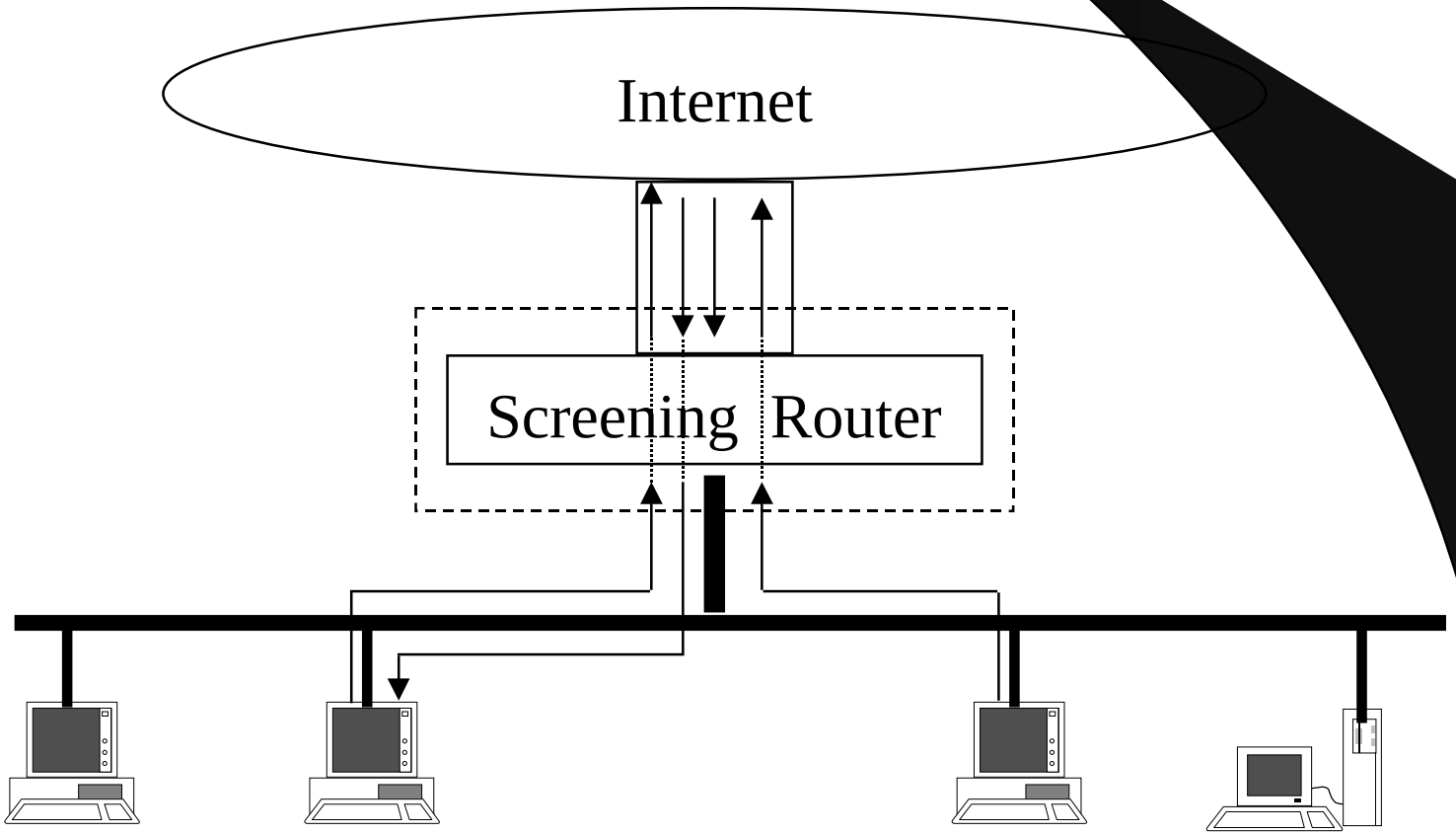
Firewalls and Policies

- Some issues
  - retrieving illegal software
  - accessing offensive material
    - e.g. pornographic images
- Role of Firewall
  - enforcing your policy

Building the Firewall

- Where does it go ?
  - between networks;
- What makes it up ?
  - routers, bastion hosts;
  - printers, other hardware.

Firewall Design #1 : Single Screening Router



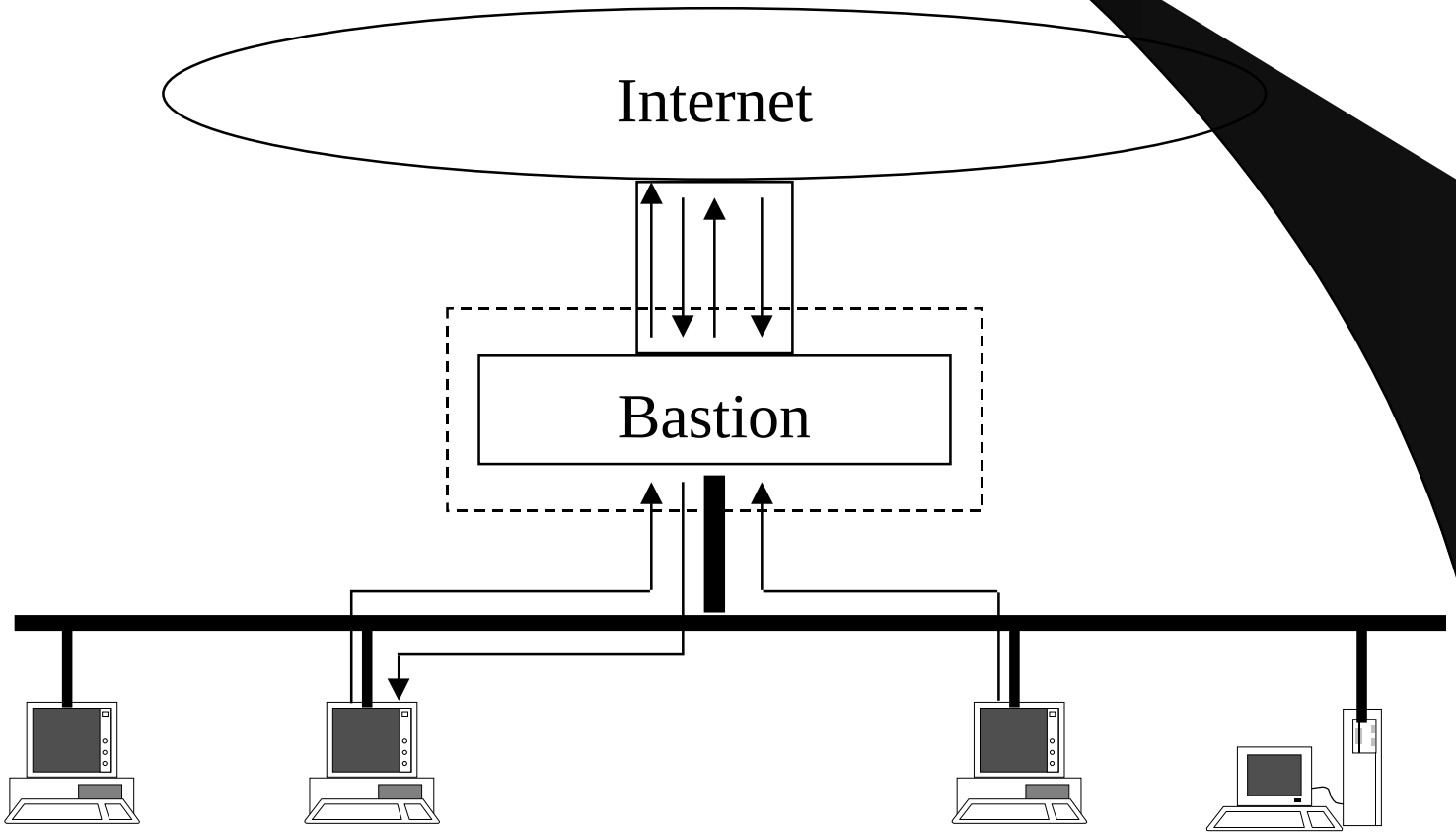
Single Screening Router

- Selectively allows IP packets through;
  - typically a router or dual-interface Unix
- IP packets must be able to travel from one endpoint to the other;
- A trivial mistake can render the Firewall ineffective;
- Interactions between rules for different services.

Screening Router #2

- Previous can be dangerous:
  - any host on the internal network can talk to any external host;
- Sometimes configured to only allow one internal host to talk to external networks;
- Other internal hosts use it as a proxy;
- Should be a dedicated router, not Unix.

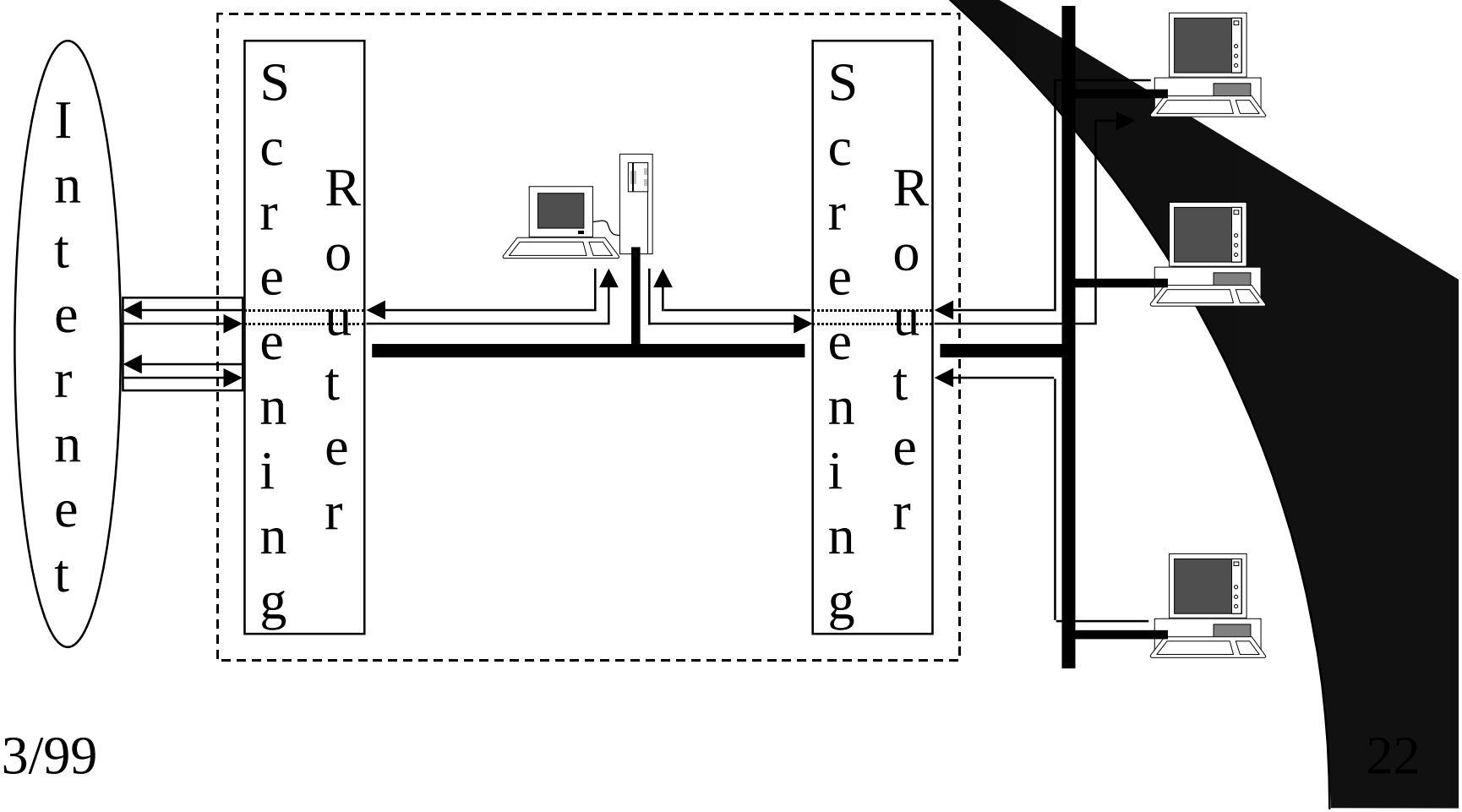
Firewall Design #2 : Dual-Homed Host



Dual Homed Host

- Bastion
  - Some sort of multi-user operating system
  - Performs proxying for all network operations
  - Configured to NOT pass or route any IP traffic
  - Can easily audit all connections
  - Can more easily filter on content than packet filtering

Firewall Design #3 : Screened Subnet



Screened Subnet

- A screening router for each exterior network
  - block access to everything but the bastion
- ‘DMZ’ between interior and exterior router
  - host(s) in DMZ are not trusted by anything
  - host(s) in DMZ do not trust anything

Adding Extra Components

- Dialin Server
  - PPP/SLIP
    - e.g. POP mailers
  - ARA (Appletalk Remote Access)
  - tty logins
  - using OTPs (One Time Passwords)
- Dedicated Internet Servers

Internet Services.

- DNS
- HTTP
- FTP
- Usenet
- E-mail
- IRC
- MBONE

Split DNS *Outside*

- External server:
  - provides information for external hosts ONLY;
  - points to internal server for all client queries (/etc/resolv.conf);
  - /etc/named.boot forwards queries to external nameservers
  - doesn’t provide entire internal DNS information.

Split DNS *Inside*

- DNS Server providing information for internal clients
  - provides information for internal hosts ONLY;
  - ALL internal client queries point to it (/etc/resolv.conf);
  - /etc/named.boot forwards unanswerable queries to external server.

WWW Servers

- Dedicated machine ?
- Run as an unpriviliedged user
  - maybe on a port other than 80 (relay port 80);
  - use chroot environment if possible;
- Beware of CGI scripts
- Can double as a cache/proxy
    - e.g. CERN httpd

Anonymous FTP

- FTP daemons have had their share of security bugs
- make sure “ftp” has an invalid password
- wu-archive ftpd
- checklist:
  - root owns ~ftp;
  - don’t allow group/world writeable directories;

E-Mail

- Configure firewall as relay host
- sendmail
  - security horror story unto itself
  - can be used to hide internal structure
    - e.g. rewriting mail headers (tricky!)
  - run with minimal priviliedge
  - use smrsh to limit shell activity

Usenet

- Dedicated internal machine ?
  - Requires a large amount of bandwidth;
  - Requires a large amount of diskspace for a reasonably complete feed;
- Restrict packets to/from feeds only;

IRC

- Easy to proxy client-server connections;
- Servers can act as proxies;
- Some Unix clients have *dangerous* scripting languages;
- Can therefore be dangerous to allow naive users to interact with malicious users.

MBONE

- Multicast-IP over UDP;
- Can be tunnelled over non-multicast supporting IP networks:
  - multicast packets aren't *routed* normally;
  - proxy'ing is too slow.

Variations

- Bastion between routers;
- Combining screening router and bastion;
- Merging a router in your DMZ with a host;
- Multiple access points to your network;
- Multiple access points to the Internet.

Bastion between routers

- Dual homed bastion;
- Two screening routers:
  - one connected externally and to the dual-homed bastion;
  - one connected internally and to the dual-homed bastion.

Combined Screening Router and Bastion

- Requires Unix with packet filtering ability;
- If the bastion is compromised, internal network is directly at risk.
- Which services to proxy, and which not to ?
  - FTP proxy
  - WWW proxy

Merging Exterior router with Bastion host

- Exposes the bastion to direct IP attacks
  - may require extra software/hardware or configuration changes;
- Still leaves internal network *separate*.

Merging Interior Router with Bastion Host

- Screened subnet is removed;
- If the bastion is compromised, internal network is directly at risk.

Multiple Internal Routers

- Adds more paths into your internal (valuable) network;
- Increases risk of a breakin;
- Routing errors:
  - may result in routing internal traffic via external network
  - more likely for a access list problem/inconsistancy.

Multiple External Routers

- Provides redundant Internet connections;
- Should only provide more ways to get to your Firewall and not through it;

Comparisons

- Cost
  - deploying a DMZ (at least 3 components)
- Complexity
  - managing your Firewall
- Risk
- Vulnerability

To Proxy or not to Proxy

- Proxying is slower than packet filtering;
- Proxying is less risky than packet filtering;
- Easier to debug proxy problems;
- Proxying provides access to session layer and higher data;
  - much easier to filter on data stream content;
- Proxying takes more effort to setup.

Building Your own Firewall

- Freely available packages
- Which Unix for your Firewall ?
- Reliability ?
- Configuring Internet Services for a Firewall

Freely Available Packages
(unbundled)

- SOCKS
- Freestone Toolkit
- TIS Firewall Toolkit (FWTK)
- ipfirewall
- IP Filter

SOCKS (Version 4)

- Requires support of client software to work
  - patches available for some client software (but must have source), else custom clients required;
  - programs like Netscape will come with built-in support;
- Logging is done by clients;
- Only works with TCP.

SOCKS Package

- Server (proxy agent);
  - does access control checks;
- Client library;
- SOCKSified clients
  - telnet, ftp, etc
- Mac, Windows versions of client software.

SOCKS (Version 5)

- RFC 1928 (Proposed Standard);
  - supports GSSAPI;
  - supports IPv4 & IPv6;
- Handles UDP;

Freestone Toolkit

- Curtesy of Sources of Supply (SOS) Corporation;
  - http://www.soscorp.com
- Recommends using TCP wrapper;
- Complex configuration files.

TIS Firewall Toolkit (FWTK)

- Curtesy of Trusted Information Systems (TIS)
  - http://www.tis.com/
- Individual proxies for common protocols
  - ftp-gw, http-gw, etc;
- Generic proxy to handle others (plug-gw);
- Alternate mail relay software provided
  - smapd/smap;

FWTK & Dual-homed Bastion

- Single configuration file;
- all clients have access control builtin;
- netacl;
  - access control
  - execute different proxies for different destinations/origins;
- support for extended authorization checks (authserv).

Freestone/FWTK Legalities

- Is freely available but limitations on:
  - using it as part of a “product”;
  - making changes to it;
- Officially unsupported
  - FWTK users are encouraged to buy Gauntlet if they like it.

ipfirewall

- Is available freely and can be registered;
- Shares common heritage with filtering code in Linux/FreeBSD (ipfw/ipfwadm);
- Only works on BSD Unixes based on 4.4BSD-Lite
  - NetBSD, FreeBSD, BSD/OS;
- Single command interface to kernel code.

IP Filter

- Freely available (for commercial use too);
- Works on:
  - SunOS 4.1.x, Solaris 2.x, FreeBSD, NetBSD, OpenBSD, BSD/OS;
- Web Page
  - http://coombs.anu.edu.au/~avalon/ip-filter.html

IP Filter details

- Provides filtering of IP packets;
- Can filter on addresses, ports, ICMP types, etc;
- Provides logging through a character device
  - log details can be stored in a file or sent to syslog or both;
- Can send *fake* ICMP/TCP RST replies

IP Filter as part of the system

- Can be built into the kernel
  - SunOS 4.1.x, NetBSD, FreeBSD, BSD/OS;
- Can be used as a loadable kernel module
  - Solaris 2, SunOS 4, NetBSD, FreeBSD;
- Set of Unix tools to monitor and control the filter;

Which Unix for your Firewall ?

- SunOS 4
- Solaris 2
- Linux
- BSDI’s BSD/OS
- Other BSDs:
  - FreeBSD
  - NetBSD

SunOS 4.1.4 or Solaris 2.5 ?

- SunOS 4.1.4 is Solaris 1.1.2
- Solaris 2.5 is SunOS 5.5
- No C compiler bundled with Solaris 2
- Many bugs fixes in SunOS 5.5 kernel that aren’t (and won’t be) in SunOS 4.1.4
- Solaris 2-x86 for PCs is available

Linux

- Continually evolving;
- “Young” compared to other Unix’s;
- No less bugfree;
- Works with a large variety of PC hardware;
- Source code is freely available;
- Comes with its own *Firewall* code.

BSD/OS

- Only remaining commercial BSD Unix;
- Can be used on PC hardware;
- Source code licence is relatively cheap;
- Commercial, thus support is available.

FreeBSD

- Dedicated to working on PCs
  - very friendly installation and configuration;
- Based on 4.4BSD-Lite;
- Robust networking and filesystem code;
- Works with a large variety of PC hardware;
- Source code is freely available;
- Comes with its own *Firewall* code.

NetBSD

- Works on a large variety of different hardware platforms
  - e.g. Suns, DECs, Alphas, Macs, Amigas, etc;
- Based on 4.4BSD-Lite;
- Robust networking and filesystem code;
- Source code is freely available.

Setting up your Firewall

- Install Operating System
- Install Firewall Software
- Test Configuration

Operating System Installation

- Disconnect your Firewall to-be from any network
- Use original media (tape/CD-ROM)
- Configure your system

Operating System Configuration

- Check permissions on files and directories
- Remove/disable set-uid and set-gid programs
- Delete all non-essential accounts
  - e.g. lp, ingres, games, etc.
- Install other software needed
  - e.g. tcp-wrappers, gcc, etc

Firewall Software Installation

- Copy software from CD-ROM/tape/floppy
- Build and install your Firewall software
- Configure your Firewall
- Cleanup
  - remove excess programs used for installation but not required for operation.

Test Firewall Setup

- Connect your Firewall to the Internal network and attempt to use it
- Disconnect from Internal network and connect to External network and test again
- Monitor your Firewall closely during the tests and look for unexpected behaviour
  - make changes as necessary and repeat until the Firewall performs as required.

Going Live

- Connect your Firewall to both Internal and External networks.
- Repeat tests
  - DO NOT be afraid to disconnect if a problem arises.
- Announce availability to users.

Logging

- Enable syslog'ing of everything
  - even unused facilities
- Enable process accounting
- Setup log rollover
  - on a daily/weekly/monthly basis ?
  - every x kilo-bytes ?
- Hard copies of logfiles

Logging Network Connections

- Setup and use TCP wrapper
  - "-t" flag for inetd under Solaris 2
- Logging access to portmapper
  - use portmap 3.0
  - doesn't log connections direct to RPC services
  - modify shared libraries

TCP Wrapper

- Used as a standin for the real program
  - executes real daemon if access is granted
- Checks *connecting* IP# against Access Control Lists
  - Use of IP#'s in ACLs is more secure than domainnames

Attacking TCP Wrapper

- Can be configured to do reverse DNS lookups to check IP# to domainname mapping
- Can be setup to disable source-routed connections
  - requires patch for SunOS 4.1.1-3
- Can be setup to disable IP spoofing attacks

IP Spoofing Attacks

- Source routed
- Using altered routes/ARP entries
- *Blind* attacks

Source Routed Attacks

- Route to be used is encoded in the IP options of the packet header by attacker;
- Route in packet header is used to send packets back;
- Packet filtering often allows this to be filtered on/disabled
  - e.g. Cisco's support "no ip source-route"

Fake ARP/Routing table entries

- ARP attacks only effective for LAN connections
  - can be effective against Intranet servers
- Routing tables can often be easily spoofed
  - e.g. sending fake RIP packets (no authentication)
- Forged ICMP redirects

Blind Attacks

- Packets are just send, no replies seen
- Replies are guessed/calculated by attacker
- Best done by making a trusted host *silent*
  - attack style used by Kevin Mitnick

Retarding IP Attacks

- ICMP
  - filter out (block) all ICMP redirects;
- Routing protocol
  - use static routes wherever possible;
  - leave the “real routing” to routers;
- Blocking IP spoofing attacks
  - block any traffic which comes from a network address you use, including 127.0.0.1

Patching the Kernel (SunOS4)

- adb -w -k /vmunix /dev/kmem
  - nfs_portmon/W1, nfs_portmon?W1
  - ip_forwarding/W-1, ip_forwarding?W-1
- Compile time options:
  - options “IPFORDWARDING=-1”
- Loadable Kernel Modules, NIT.

Patching the Kernel (Solaris2)

- /etc/init.d/inet, ndd -set
  - /dev/ip ip_path_mtu_discovery 0
  - /dev/ip ip_forwarding 0
  - /dev/ip ip_forward_src_routed 0
  - /dev/ip ip_ignore_redirect 0
  - /dev/ip ip_strict_dst_multihoming 1
  - /dev/tcp tcp_conn_req_max 32
  - /dev/tcp tcp_close_wait_interval 30000
  - /dev/tcp tcp_strong_iss 1 (2.5 only - default)

Making Sendmail Safe.

- Use mail.local as the Mlocal, rather than /bin/mail
- Turn on options:
  - Opauthwarnings needmailhelo noexpn restrictmailq

Reliability ?

- Keeping your Firewall up 24x7
  - Internet connectivity ?
  - Redundant systems ?
  - Backups ?
- Software problems
  - Product/user support ?
  - (Security) patches ?

Encryption

- Encrypting IP packets
  - SKIP/Photuris (IETF IPsec);
- VPN's (Virtual Private Networks)/ VNP's (Virtual Network Perimeters)
  - encrypted IP tunnels between private networks over public networks;
- Key management.

Authentication

- One time passwords (challenge-response):
  - cards: SecureID, etc;
  - algorithms: S/Key;
- Kerberos (ticket based):
  - server;
  - client.
- FWTK authserv.

Where to get more information

- Books
  - "Firewalls and Internet Security",
    - Bellovin & Cheswick, Addison Wesley
      - ISBN 0-201-63357-4
  - "Building Internet Firewalls"
    - Chapman & Zwick, O'Reilly & Associates
      - ISBN 1-56592-124-0

Mailling lists

- Firewalls
  - firewalls@greatcircle.com
    - (send e-mail to majordomo@greatcircle.com)
  - vendor mailling lists:
    - fwtk-users@tis.com
- Security bugs
  - bugtraq@crimelab.com
  - ntsecurity@iss.com
  - best-of-security@suburbia.net