Distributed Systems

Firewalls: Defending the Network

Paul Krzyzanowski
pxk@cs.rutgers.edu

inetd

Most UNIX systems ran a large number of tcp services as dæmons
- e.g., *rlogin, rsh, telnet, ftp, finger, talk, …*

Later, one process, ***inetd***, was created to listen to a set of ports and then spawn the service on demand
- pass sockets as standard in/standard out file descriptors
- servers don't run unless they are in use

TCP wrappers (*tcpd*)

- Plug-in replacement to *inetd*
- Restrict access to TCP services
  - Allow only specified machines to execute authorized services
  - Monitor and log requests
- Specify rules in two files:
  - `hosts.allow` and `hosts.deny`
  - access:
    - grant access if service:client in /etc/hosts.allow
    - deny access if service:client in /etc/hosts.deny
    - otherwise allow access
- support for booby traps (**honeypots**)

Firewalls

Isolate trusted domain of machines from the rest of the untrusted world
- move all machines into a private network
- disconnect all other systems
- untrusted users not allowed

not acceptable – we want to be connected

Solution:

protect the junction between a trusted internal network of computers from an external network with a **firewall**

Firewalls

Two major approaches to building firewalls:

packet filtering

proxies

Packet filtering

- Selective routing of packets
  - Between internal and external hosts
- By routers, kernel modules, or firewall software
- Allow or block certain types of packets

Screening router
- determine route *and* decide whether the packet should be routed

Packet filtering: screening router

IP packet data

Filter by

- IP source address, IP destination address
- TCP/UDP source port, TCP/UDP destination port
- Protocol (TCP, UDP, ICMP, …)
- ICMP message type
- interface packet arrives on
- destination interface

Allow or block packets based on any/all fields

- Block any connections from certain systems
- Disallow access to "dangerous services"

Packet filtering

Stateless inspection

- filter maintains no state
- each packet examined on its own

Packet filtering

Stateful inspection

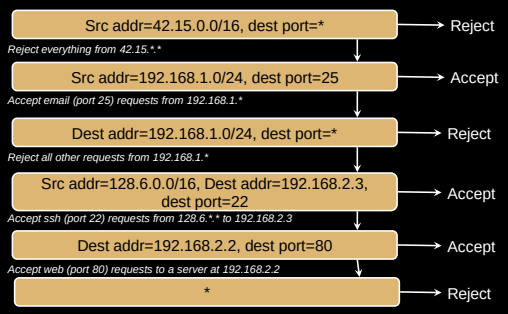
- keep track of TCP connections (SYN, SYN/ACK packets)
  - e.g. no rogue packets when connection has not been established
- "related" ports: allow data ports to be opened for FTP sessions
- Port triggering (outbound port triggers other port access to be redirected to the originating system)
  - Generally used with NAT (Network Address Translation)
- limit rates of SYN packets
  - avoid SYN flood attacks
- Other application-specific filtering
  - Drop connections based on pattern matching
  - Rewrite port numbers in data stream

Packet filtering

Screening router

- allows/denies **access to a service**
- cannot protect operations within a service

Packet filtering: rules



Proxy services

- Application or server programs that run on firewall host
  - dual-homed host
  - bastion host
- Take requests for services and forward them to actual services
- provide replacement connections and act as gateway services
- Application-level gateway

Stateful inspection and protocol validation

Proxy services

Proxies are effective in environments where direct communication is restricted between internal and external hosts

- dual-homed machines and packet filtering

Proxy example

Checkpoint Software Technologies' Firewall-1

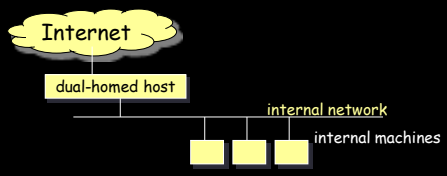
mail proxy:

- mail address translation: rewrite From:
- redirect To:
- drop mail from given address
- strip certain mime attachments
- strip Received info on outbound mail
- drop mail above given size
- perform anti-virus checks on attachments

does not allow outsiders direct connection to a local mailer

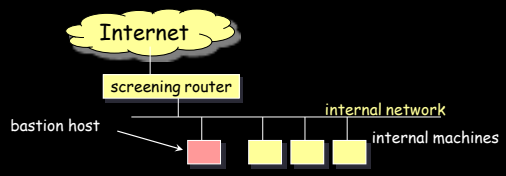
Dual-homed host architecture

- Built around dual-homed host computer
- Disable ability to route between networks
  - packets from Internet are not routed directly to the internal network
  - services provided by proxy
  - users log into dual-homed host to access Internet
  - user accounts present security problems



Screened host architecture

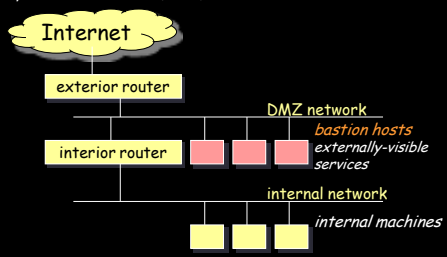
- Provides services from a host attached to internal network
- Security provided by packet filtering
  - only certain operations allowed (e.g. deliver email)
  - outside connections can only go to bastion host
- allow internal hosts to originate connections over Internet
- if bastion host is compromised…



Screened subnet architecture

Add extra level of isolation for internal network

- Place any externally visible machines on a separate *perimeter network* (DMZ)



Screened subnet architecture

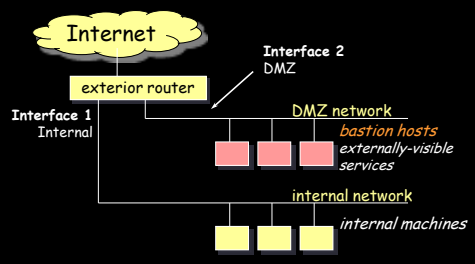
Exterior router (access router)

- protects DMZ and internal network from Internet
- generally… allow anything outbound … that you need
- block incoming packets from Internet that have forged source addresses
- allow incoming traffic only for bastion hosts/services.

Interior router (choke router)

- protects internal network from Internet and DMZ
- does most of packet filtering for firewall
- allows selected outbound services from internal network
- limit services between bastion host and internal network

Single router DMZ



Firewalling principles

- It is easier to secure one or a few machines than a huge number of machines on a LAN
- Focus effort on bastion host(s) since only they are accessible from the external network
- All traffic between outside and inside must pass through a firewall
- Deny overall
  - Turn everything off, then allow only what you need
- Private network should never see security attacks
- Be prepared for attacks from within
  - Infected machines

The end