University of BRISTOL

Systems Security

COMSM1500

University of BRISTOL

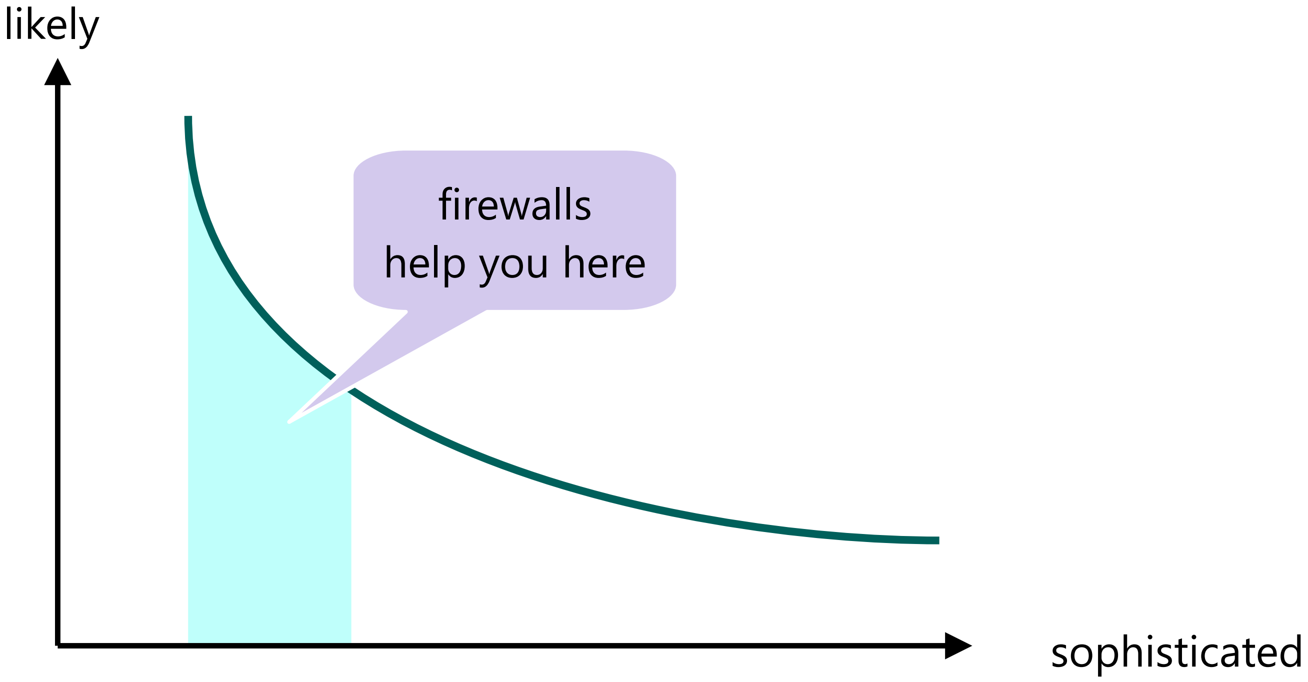
Firewalls

bristol.ac.uk

Plan

- Firewall
- Design goal
- Demilitarized zone
- Types of policies
  - white-list
  - black-list
- Types of firewall
  - Packet filter
  - Stateful filter
  - Application-level proxy
  - Circuit-level proxy
  - Personal firewall
- Attacks and firewall countermeasures
- The Great Firewall of China
- Linux iptables

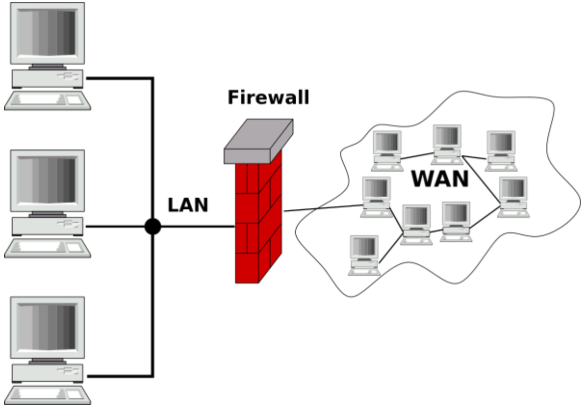
Threat Curve



firewalls
help you here

What is a firewall

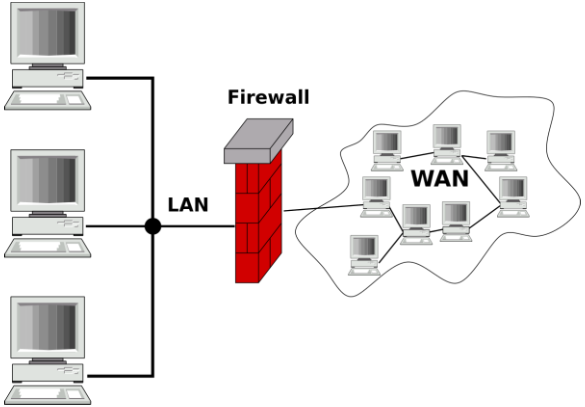
- Internet connectivity is essential
  - However, threat comes from outside
  - … remember the most secure computer is one that is turned off ;)
- Firewalls protect a LAN/machine from outside threats
- Interpose between "internet" and the local network/machine
- Used a "perimeter defense"
  - Single point of entry to impose security and auditing
  - Insulate local system from the outside world



What is a firewall

Homework/exam question:
Explain what is a firewall

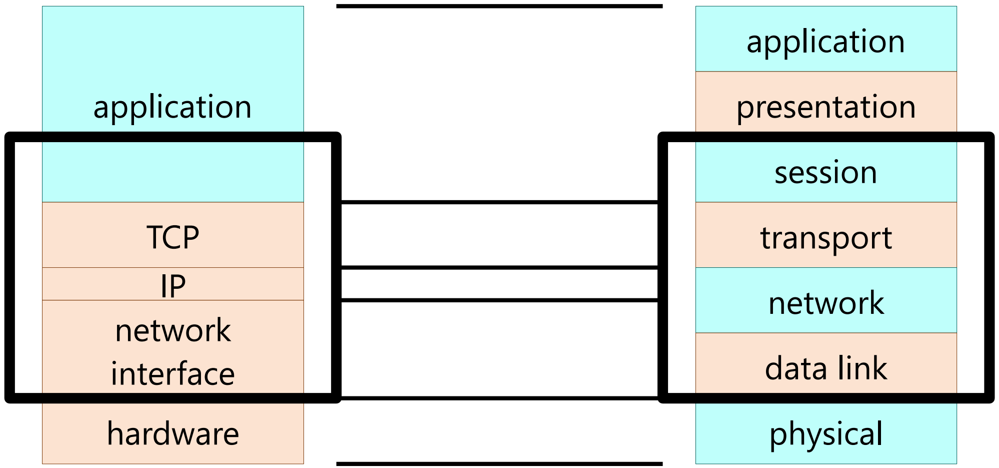
- Internet connectivity is essential
  - However, threat comes from outside
  - … remember the most secure computer is one that is turned off ;)
- Firewalls protect a LAN/machine from outside threats
- Interpose between "internet" and the local network/machine
- Used a "perimeter defense"
  - Single point of entry to impose security and auditing
  - Insulate local system from the outside world



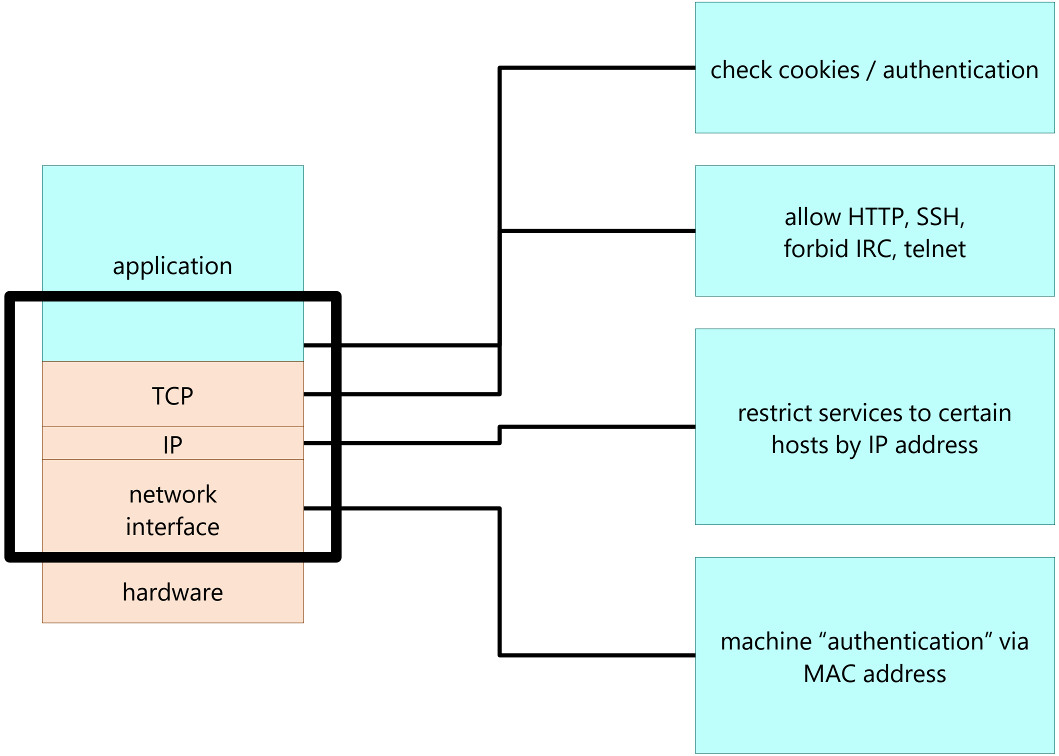
Design goals

- All traffic must pass through the firewall
  - Inside -> Outside
  - Outside -> Inside
- Only authorized traffic is allowed to pass through
  - This is defined by some security policy
- The firewall itself must be immune to penetration

Network Stack

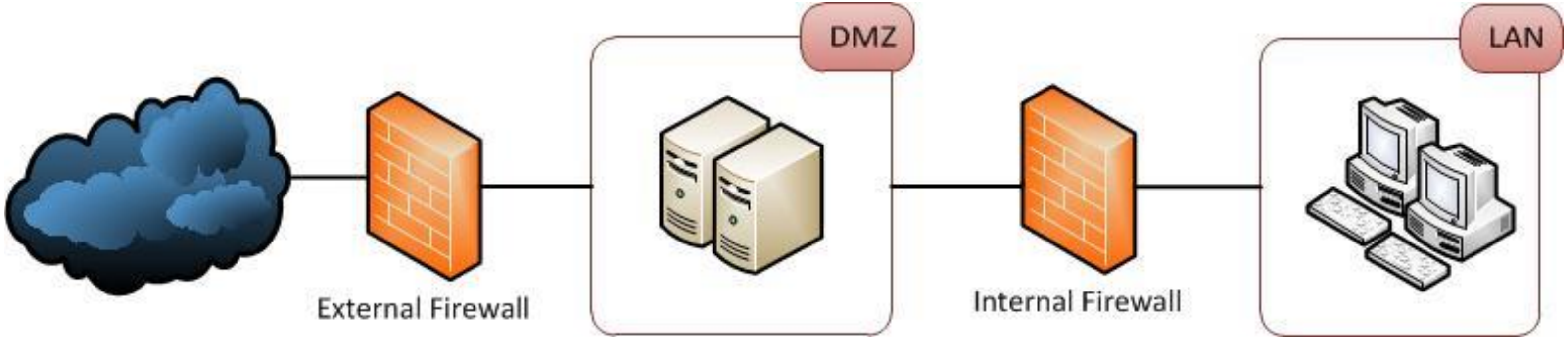


Network Stack


application
TCP
IP
network interface
hardware
check cookies / authentication
allow HTTP, SSH, forbid IRC, telnet
restrict services to certain hosts by IP address
machine “authentication” via MAC address

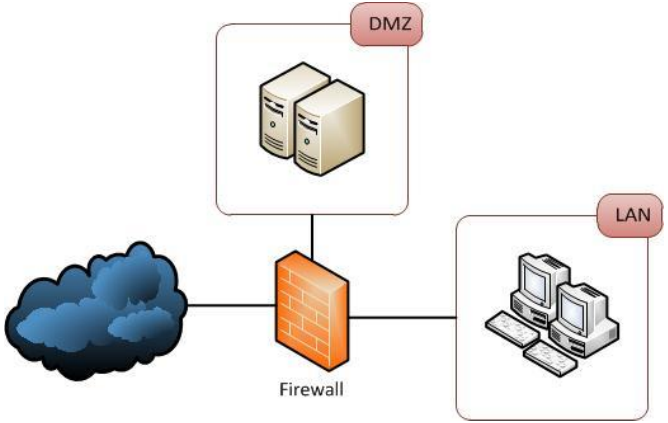
DMZ (demilitarized zone)

- Public facing resources
  - Accessed from inside network
  - … or from the outside
  - Policy may be different (e.g. no ssh from internet, but ok from LAN)

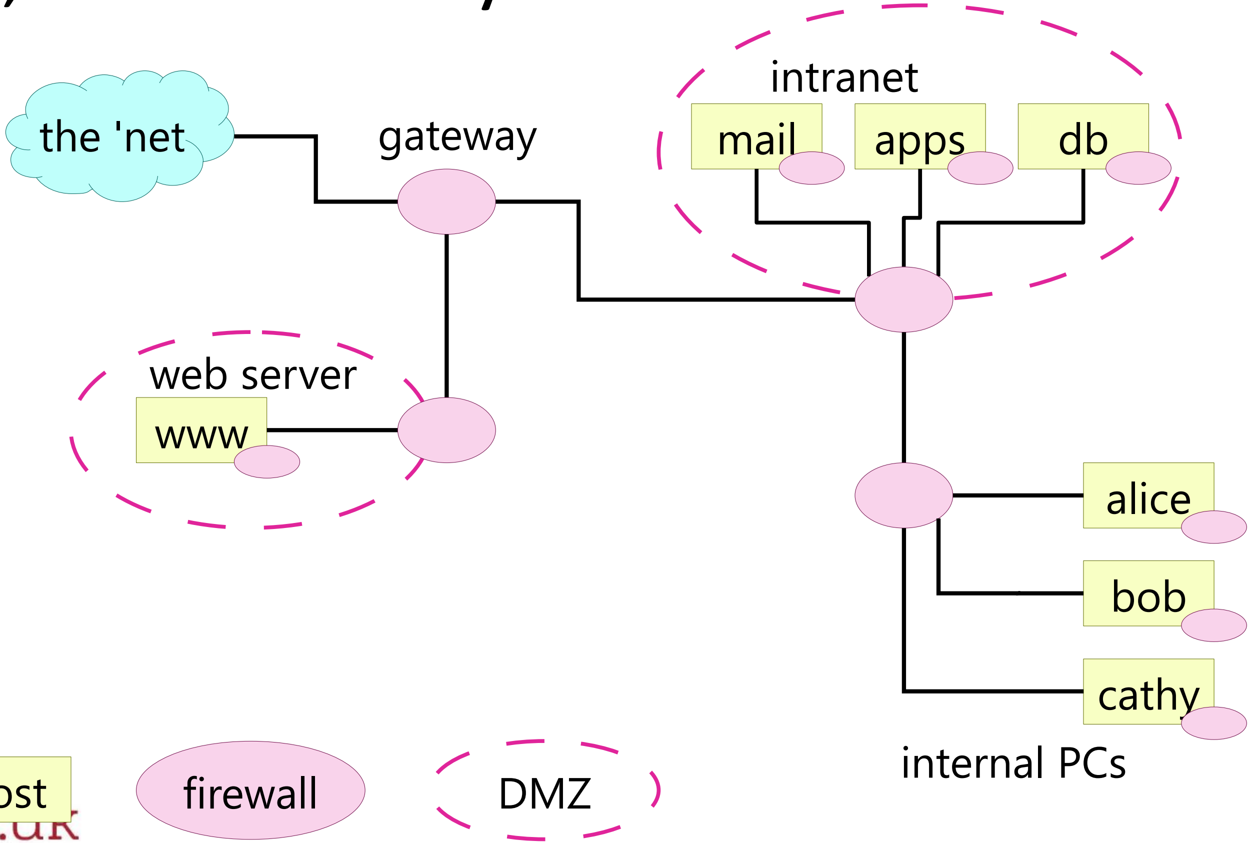


DMZ (demilitarized zone)

- Public facing resources
  - Accessed from inside network
  - … or from the outside
  - Policy may be different (e.g. no ssh from internet, but ok from LAN)

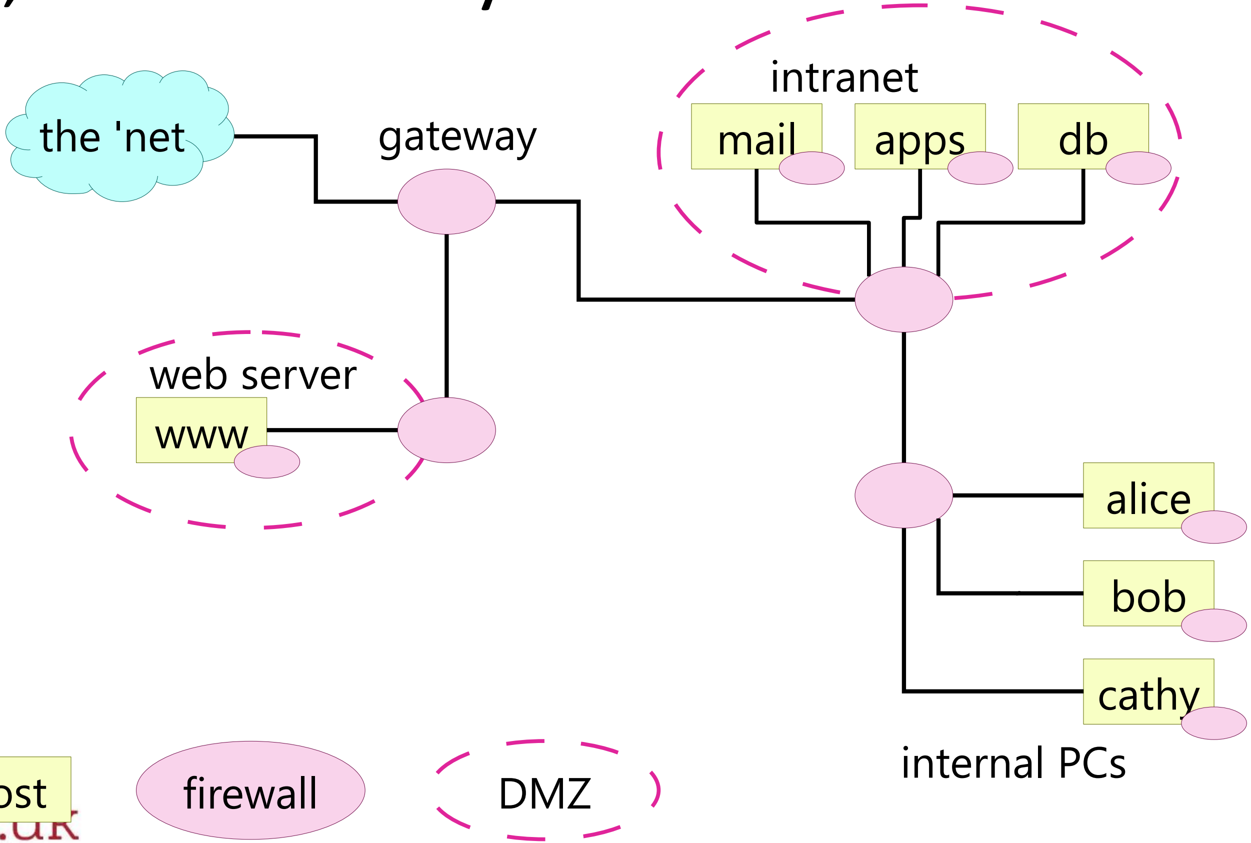


Firewall, firewall everywhere!



Homework/exam question:
Explain the role of DMZ

Firewall, firewall everywhere!



Type of policies

- Block all by default
  - a.k.a white-list
  - Only well-defined traffic
  - … justify why it should be allowed
- Allow all by default
  - a.k.a black-list
  - Only block traffic relating to known problem

Type of policies

- Block all by default
  - a.k.a white-list
  - Only well defined traffic
  - … justify why it should be allowed
- Allow all by default
  - a.k.a black-list
  - Only block traffic relating to known problem
- Where would you use black-list or white-list?

Type of policies

- Block all by default
  - a.k.a white-list
  - Only well defined traffic
  - … justify why it should be allowed
  - e.g. production web-server
- Allow all by default
  - a.k.a black-list
  - Only block traffic relating to known problem
  - E.g. individual computer

Homework/exam question: When to use white/black list, give examples

Type of policies

- Block all by default
  - a.k.a white-list
  - Only well defined traffic
  - … justify why it should be allowed
  - e.g. production web-server
- Allow all by default
  - a.k.a black-list
  - Only block traffic relating to known problem
  - E.g. individual computer

Types of firewall

packet filter

application-level proxy

stateful inspection

circuit-level proxy

personal firewall

Packet filtering

- Apply rules to each packet, generally based on TCP/IP headers.
  - e.g. allow any connection on port 80 (HTTP)
  - e.g. allow only connection from local network on port 22 (SSH)
- DPI (deep packet inspection) looks at higher layers too.
  - e.g. HTTP disallow certain hostname
- Can be done on a separate machine / router.

Stateful filtering

- Rules on processing packet, depends on previously seen packet
  - e.g. differentiate between old/new TCP connections
- Implement more complex constraints
  - HTTP server can only reply to request not establish connection
  - Verify that type of incoming/outgoing packet match

Application level proxy

- Proxy on a separate host
  - Can authenticate to the proxy separately
  - Separate connections
    - Client <-> Proxy
    - Proxy <-> Server
- Proxy must understand each protocol in use
- Much more in depth analysis
  - e.g. ftp proxy can scan content
- Downside: performance bottleneck
- Protocol need to be supported (TLS is not on purpose)

Circuit-level proxy

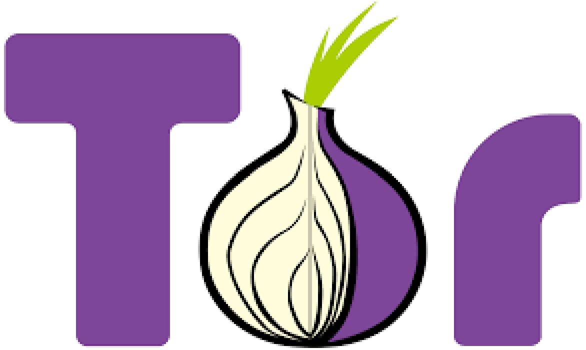
- Similar to application-level proxy but lower in the stack
  - i.e. relay TCP packets

Circuit-level proxy

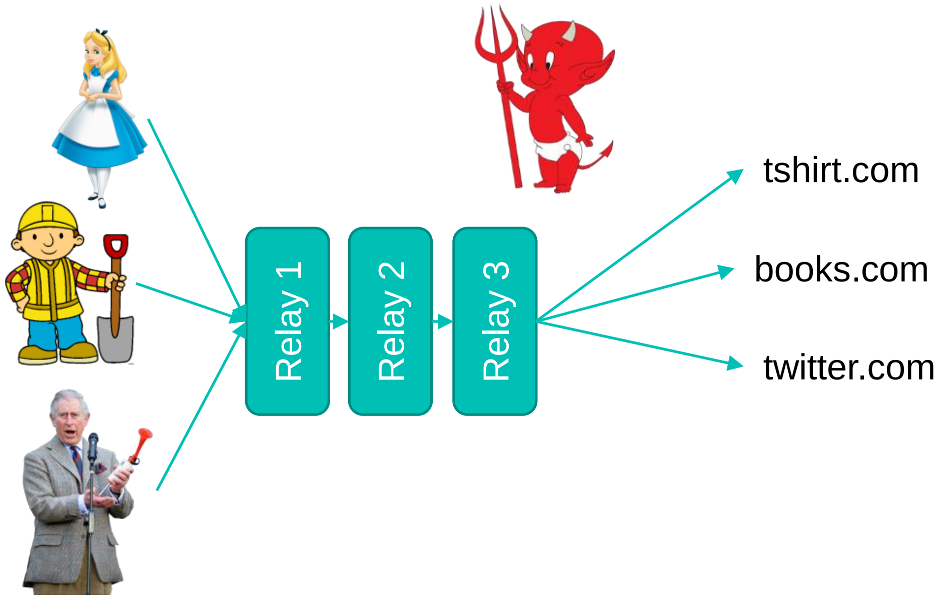
- Similar to application-level proxy but lower in the stack
  - i.e. relay TCP packets
- Can you give me an example?

Circuit-level proxy

- Similar to application-level proxy but lower in the stack
  - i.e. relay TCP packets



TOR Circuit



- Harder to know what Alice is doing
- Need to trust the relay
  - Relay 1 now Alice is doing something
  - Relay 3 now some is talking to t-shirt.com
  - Attacker need to control 1 and 3 to be really harmful
  - Hard/Costly to achieve
  - Discussed further later…

Personal firewall

- Built in or at supported by the OS
- Set access rules to individual program
  - e.g. Chrome can send packet, but not notepad
- Can ask the user to set settings
  - e.g. “allow this program to access internet”

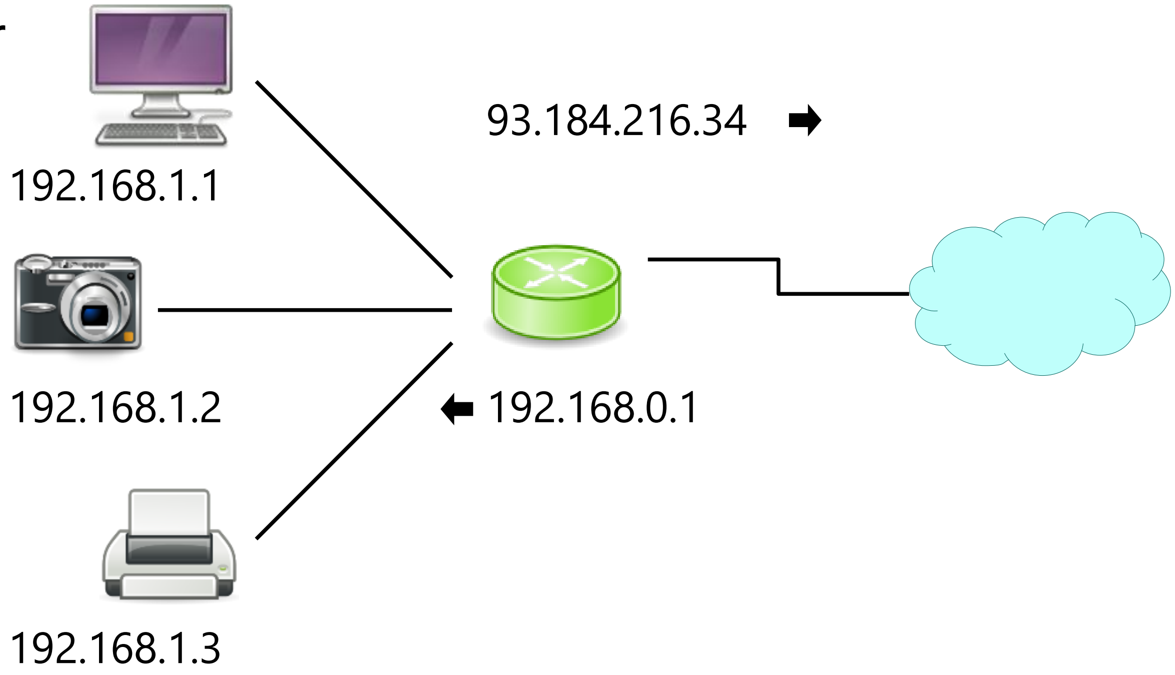
Homework/exam question:
Describe 5 types of firewall

Personal firewall

- Built in or at supported by the OS
- Set access rules to individual program
  - e.g. Chrome can send packet, but not notepad
- Can ask the user to set settings
  - e.g. “allow this program to access internet”

NAT (Network Address Translation)

- Designed to save IP addresses
- Works at IP layer
- Can be used to limit connection to only outbound
- Option for port forwarding
- e.g. your home router



Attacks and firewall countermeasures

- Slow Loris?

Attacks and firewall countermeasures

- Slow Loris?
  - Application level proxy
  - e.g. limit number of connections per IP address
  - … or impose minimum connection speed etc.

Attacks and firewall countermeasures

- Slow Loris?
- Network observation?

Attacks and firewall countermeasures

- Slow Loris?
- Network observation?
  - Circuit level proxy
  - e.g. TOR

Attacks and firewall countermeasures

- Slow Loris?
- Network observation?
- IP spoofing?
  - e.g. external IP packet pretending to come from within

Attacks and firewall countermeasures

- Slow Loris?
- Network observation?
- IP spoofing?
  - Packet filtering
  - Check IP match inside/outside logic

Attacks and firewall countermeasures

- Slow Loris?
- Network observation?
- IP spoofing?
- etc…

Attacks and firewall countermeasures

- Slow Loris?
- Network observation?
- IP spoofing?
- etc…

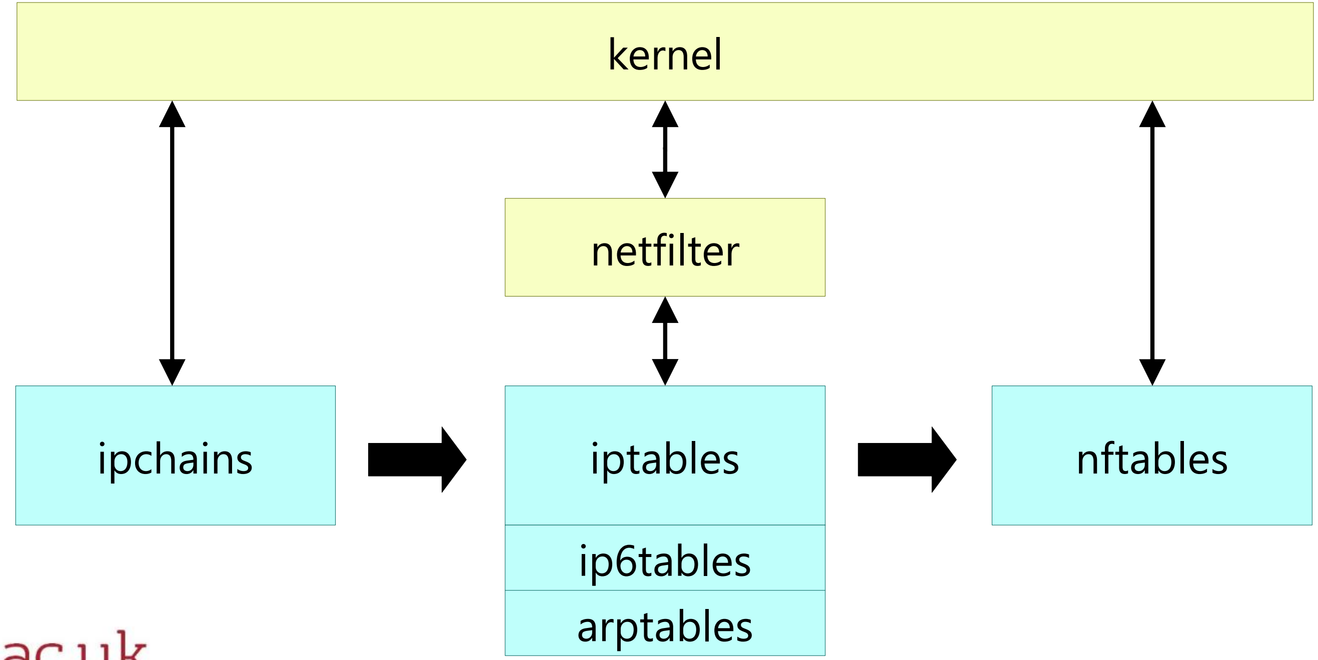
Homework/exam question: Given attack X how could you use a firewall as countermeasure.

The Great Firewall

- Block based on
  - IP address
  - URL
  - Keyword
  - Scan page content as well
  - Probably more
- Send TCP resets packet
  - we have seen in previous lecture how!
- … or drops connection
- Recently seems to be using machine learning
  - We discussed means to identify content even when connection is encrypted
- Also arm race with TOR
  - Previously discussed

iptable

iptables


kernel
netfilter
ipchains
iptables
ip6tables
arptables
nftables

Overview

- ipchains
  - Old, no more used
- Netfilter + iptables (~2000)
  - arptable, Xtable etc…
- Consolidated by nftable (~2014)
  - Single interface for all protocols
  - re-use most of the netfilter infrastructure
  - run simple virtual machine in kernel to implement firewall functions
- nft add rule ip filter output ip addr 1.2.3.4 drop
- iptables -A OUTPUT -d 1.2.3.4 -j DROP

Overview

- ipchains
  - Old, no more used
- Netfilter + iptables (~2000)
  - arptable, Xtable etc…
- Consolidated by nftable (~2014)
  - Single interface for all protocols
  - re-use most of the netfilter infrastructure
  - run simple virtual machine in kernel to implement firewall functions
- nft add rule ip filter output ip daddr 1.2.3.4 drop
- iptables -A OUTPUT -d 1.2.3.4 -j DROP
  - drop outgoing packet to 1.2.3.4

Tables and Chains

- iptables is implemented using different tables representing different stage of packet flow through netfilter/network stack
- In each table, a packet traverse a chain of function that determine if packet is dropped or transformed
- Convention:
  - Table lower case
  - Chain UPPER CASE

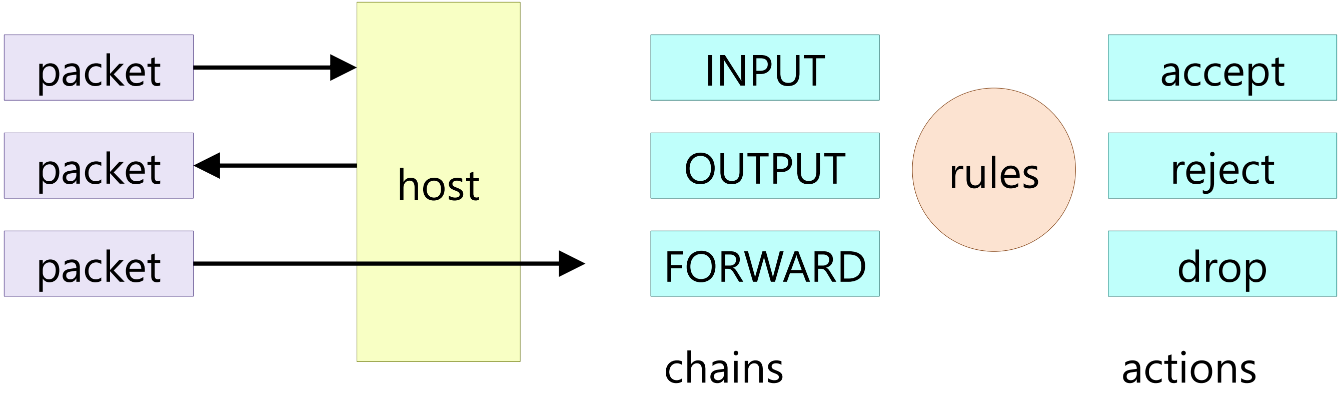
Tables

- filter
- nat
- mangle
- raw
- security

Chains

- PREROUTING
- INPUT
- FORWARD
- OUTPUT
- POSTROUTING
- Not all exist for all tables

filter table



nat table

- Network address translation
- PREROUTING
  - Modify incoming packets
- POSTROUTING
  - Modify outgoing packets

Tables

- filter
- nat
- mangle (to do more complex packet modification)
- raw (called first, should be used for low resource functionality, e.g. simple packet filtering)
- security (used to support MAC e.g. SELinux)

Examples

- `-A INPUT -s 255.0.0.0/8 -j DROP`
  - Drop any packet arriving from a local address (i.e. anti spoofing)
- `-A INPUT -p TCP --dport 80 -m state --state NEW -j ACCEPT`
  - Allow new connection on port 80 (i.e. HTTP server)
- `-A OUTPUT -p TCP -m state --state ESTABLISHED,RELATED -j ACCEPT`
  - Outgoing packet allowed on any port for established connection
- `-P INPUT DROP`
  `-P OUTPUT DROP`
  - Anything not previously allowed is dropped.

Examples

Homework/exam question:
Explain how IP table work.

- -A INPUT -s 255.0.0.0/8 -j DROP
  - Drop any packet arriving from a local address (i.e. anti spoofing)
- `-A INPUT -p TCP --dport 80 -m state --state NEW -j ACCEPT`
  - Allow new connection on port 80 (i.e. HTTP server)
- `-A OUTPUT -p TCP -m state --state ESTABLISHED,RELATED -j ACCEPT`
  - Outgoing packet allowed on any port for established connection
- `-P INPUT DROP`
  `-P OUTPUT DROP`
  - Anything not previously allowed is dropped.

Examples

Homework/exam question: Explain how IP table work. You don’t need to be able to write “policies”.

- -A INPUT -s 255.0.0.0/8 -j DROP
  - Drop any packet arriving from a local address (i.e. anti spoofing)
- `-A INPUT -p TCP --dport 80 -m state --state NEW -j ACCEPT`
  - Allow new connection on port 80 (i.e. HTTP server)
- `-A OUTPUT -p TCP -m state --state ESTABLISHED,RELATED -j ACCEPT`
  - Outgoing packet allowed on any port for established connection

-
  ```
  -P INPUT DROP
  -P OUTPUT DROP
  ```
  - Anything not previously allowed is dropped.

Plan

- Firewall
- Design goal
- Demilitarized zone
- Types of policies
  - white-list
  - black-list
- Types of firewall
  - Packet filter
  - Stateful filter
  - Application-level proxy
  - Circuit-level proxy
  - Personal firewall
- Attacks and firewall countermeasures
- The Great Firewall of China
- Linux iptables

Thank you, questions?

Office MVB 3.26