COSC 734: Network Security

Chapter 11 – Firewall

Dr. Wei Yu

Dept. of Computer and Information Sciences

Towson University

Email: wyu@towson.edu

The function of a strong position is to make the forces holding it practically unassailable

—*On War*, Carl Von Clausewitz

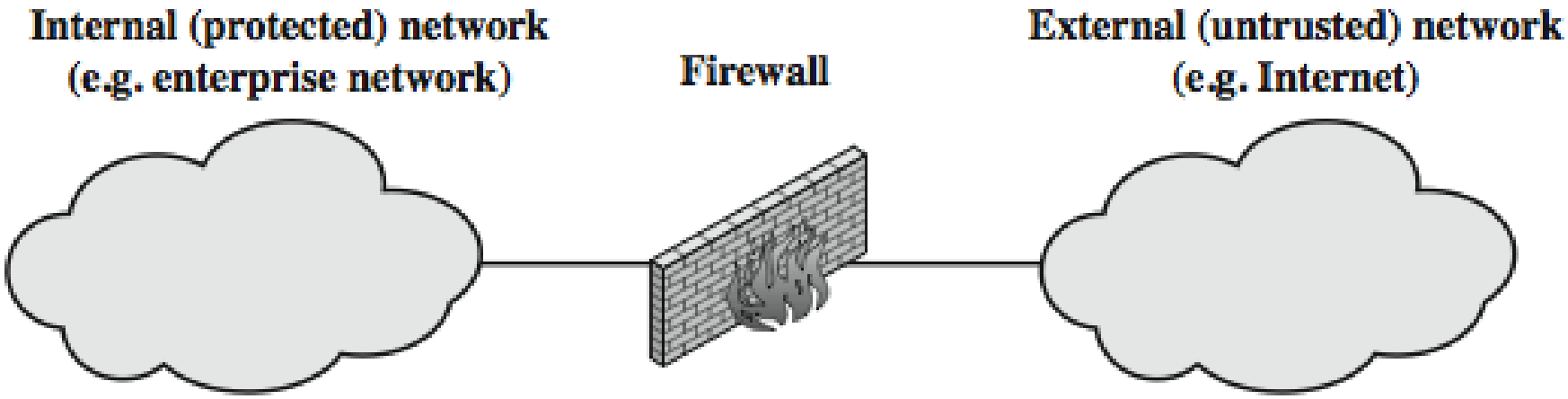
Introduction

- Seen evolution of information systems
- Internet connectivity is no longer optional, with information and services essential to the organization.
- While Internet access provides benefits, it enables the outside world to reach and interact with local network assets, creating a threat to the organization.
- Has persistent security concerns
  - can’t easily secure every system in org
- Typically use a **Firewall**
- To provide **perimeter defence**
- As part of comprehensive security strategy

What is a Firewall?

- A **choke point** of control and monitoring: defines a single choke point
  - Keeps unauthorized users out of the protected network
  - Prohibits potentially vulnerable services from entering or leaving the network,
  - Provides protection from various kinds of IP spoofing and routing attacks.
- Interconnects networks with differing trust
- Imposes restrictions on network services
  - only authorized traffic is allowed
- Auditing and controlling access
  - can implement alarms for abnormal behavior
- Provide NAT & usage monitoring
- Implement VPNs using IPSec
- Must be immune to penetration

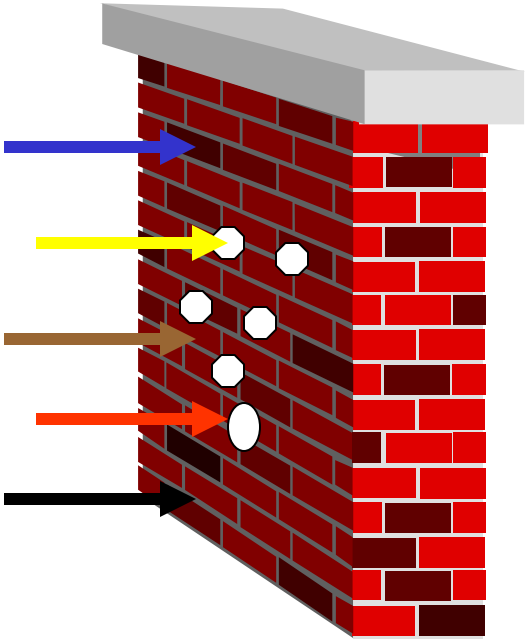
What is a Firewall?



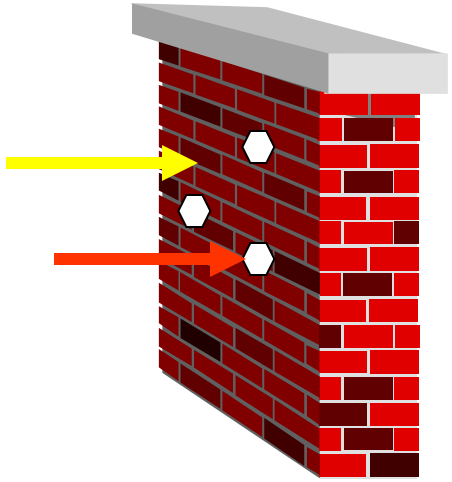
Firewall Limitations

- Cannot protect from attacks bypassing it
  - e.g., sneaker net, utility modems, trusted organisations, trusted services (e.g., SSL/SSH)
- Cannot protect against internal threats
  - e.g., disgruntled or colluding employees
- Cannot protect against access via WLAN
  - if improperly secured against external use
- Cannot protect against malware imported via laptop, PDA, storage infected outside

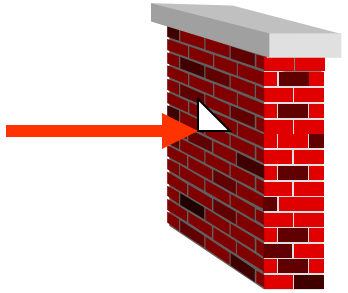
Internet Security Mechanisms: Big Picture



Prevent:
Firewall, IPsec, SSL



Detect:
Intrusion Detection



Survive/
Response:
Recovery, Forensics

- Goal: prevent if possible; detect quickly otherwise; and confine the damage

Firewall Capabilities

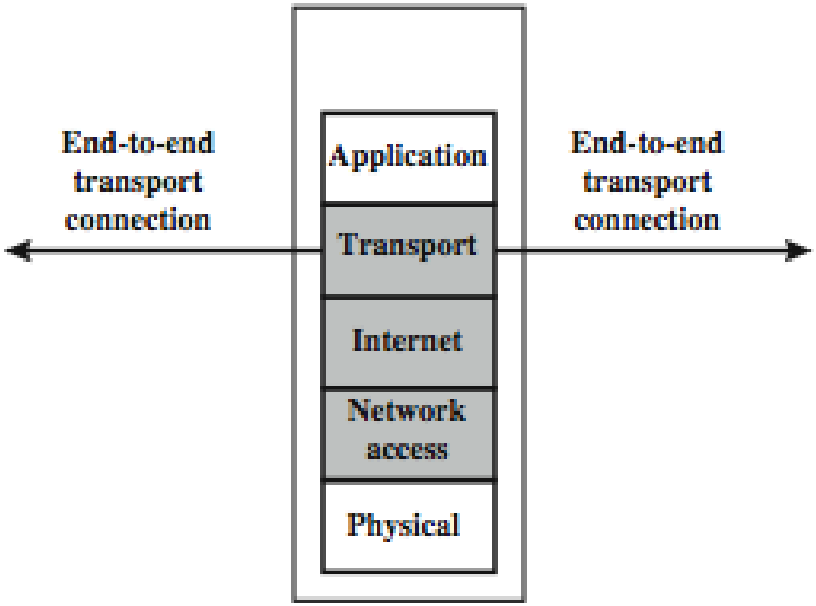
- Controlled access
  - restrict incoming and outgoing traffic according to security policy
- Other functions
  - log traffic, for later analysis
  - network address translation
  - encryption / decryption
  - application (payload) transformations

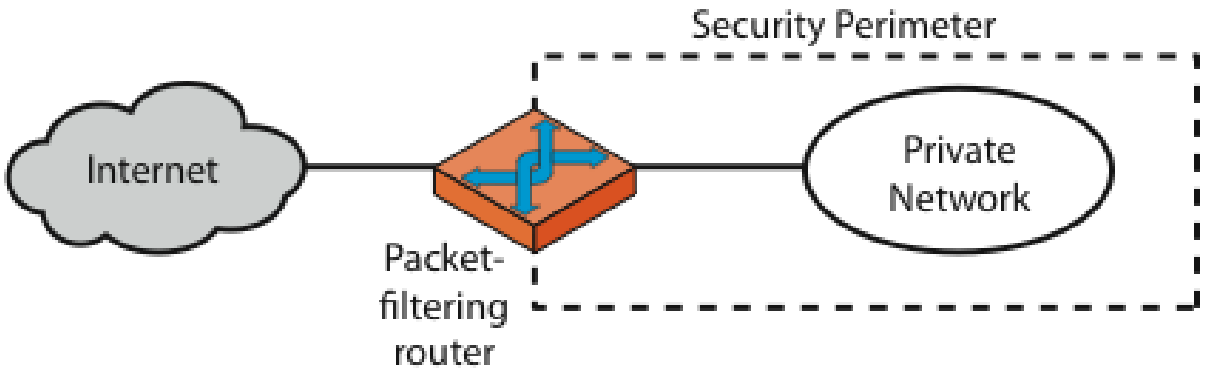
Firewall examples:
Checkpoint
Cisco PIX

Firewalls – Packet Filters

- Simplest, fastest firewall component
- Foundation of any firewall system
- Examine each IP packet (no context) and permit or deny according to rules
- Hence restrict access to services (ports)
- Some advantages are simplicity, transparency & speed
- Possible default policies
  - That not expressly permitted is prohibited: **conservative policy**
  - That not expressly prohibited is permitted: **permissive policy**

Firewalls – Packet Filters





(a) Packet-filtering router

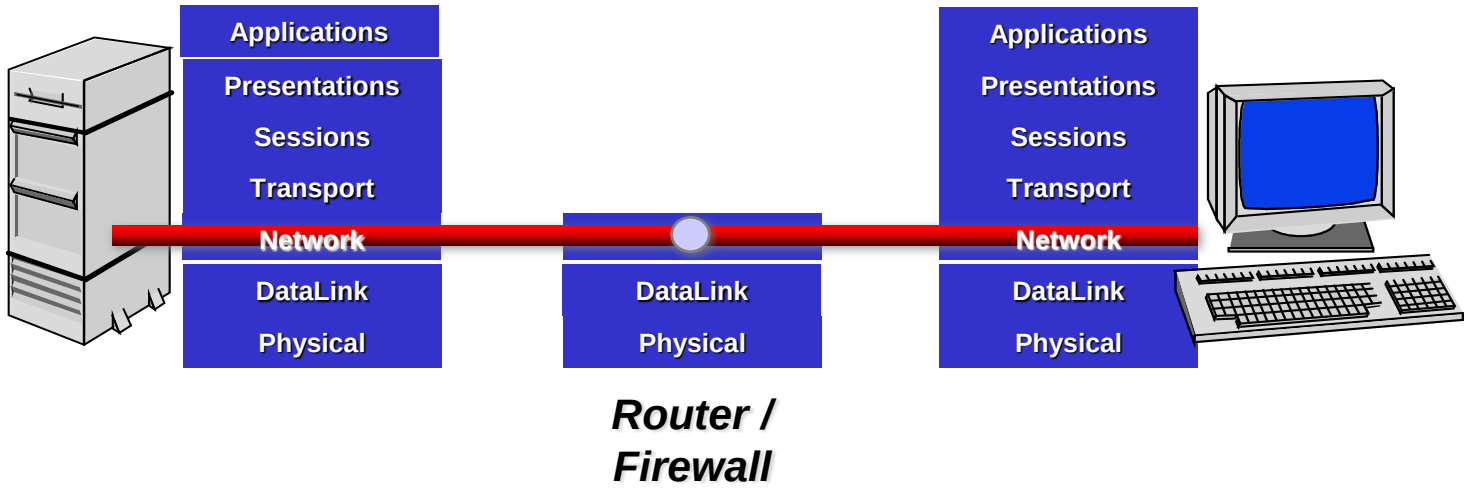
Packet filters

Analyzing packets going through the filter and determining "drop" or "allow" based on the following:

- Source IP address
- Destination IP address
- Source port
- Destination port
- TCP flag
  - SYN bit set: datagram for connection initiation
  - ACK bit set: part of established connection
- TCP or UDP or ICMP
  - Can be configured to block all UDP
- Direction
  - Leaving or entering the internal network?
- Router/firewall interface
  - decisions can be different for different interfaces

Packet Filtering

- Patterns specify values in the header of a single packet, e.g.,
  - Source IP address and port number
  - Destination IP address and port number
  - Transport protocol type



Firewalls – Packet Filters Examples

Table 20.1 Packet-Filtering Examples

A

action	ourhost	port	theirhost	port	comment
block	*	*	SPIGOT	*	we don't trust these people
allow	OUR-GW	25	*	*	connection to our SMTP port

B

action	ourhost	port	theirhost	port	comment
block	*	*	*	*	default

C

action	ourhost	port	theirhost	port	comment
allow	*	*	*	25	connection to their SMTP port

D

action	src	port	dest	port	flags	comment
allow	{our hosts}	*	*	25		our packets to their SMTP port
allow	*	25	*	*	ACK	their replies

E

action	src	port	dest	port	flags	comment
allow	{our hosts}	*	*	*		our outgoing calls
allow	*	*	*	*	ACK	replies to our calls
allow	*	*	*	>1024		traffic to nonservers

Firewalls – Packet Filter Examples

A. Inbound mail is allowed to a gateway host only (port 25 is for SMTP incoming)

B. Explicit statement of the default policy

C. Tries to specify that any inside host can send mail to the outside, but has problem that an outside machine could be configured to have some other application linked to port 25

D. Properly implements mail sending rule, by checking ACK flag of a TCP segment is set

E. This rule set is one approach to handling connections for applications

Limitation on Packet Filters

- IP address spoofing
  - Fake source address to be trusted
  - Add filters on router to block
- Source routing attacks
  - Attacker sets a route other than default
  - Block source routed packets

Firewalls – Stateful Packet Filters

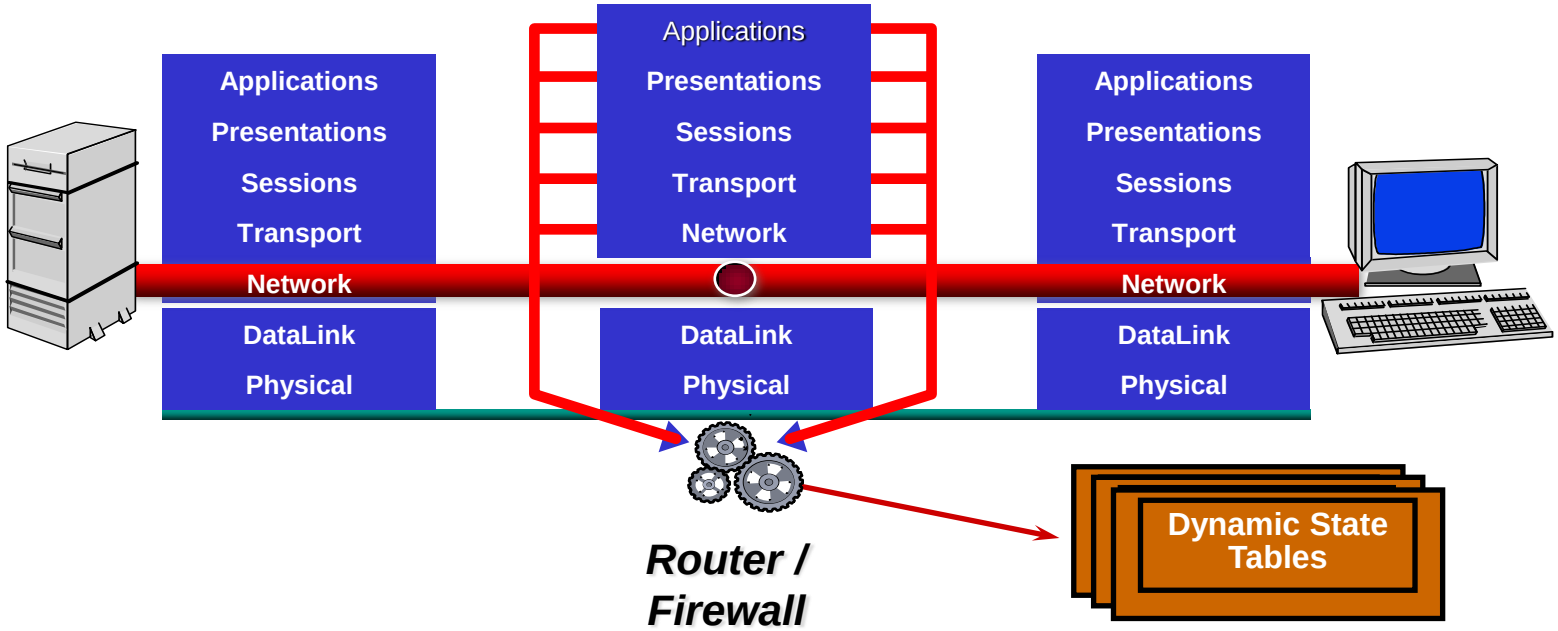
- Traditional packet filters do not examine higher layer context
  - i.e., matching return packets with outgoing flow
- Stateful packet filters address this need
- They examine each IP packet in context
  - Keep track of client-server sessions
  - Check each packet validly belongs to one
  - Some stateful firewalls keep track of TCP sequence numbers to prevent attacks that depend on the sequence number, such as session hijacking
- Hence are better able to detect bogus packets out of context
- May even inspect limited application data for some well-known protocols like FTP, IM and SIPS commands, in order to identify and track related connections

Stateful Packet Filter

- Packet decisions are made in the context of a *connection* or *flow* of packets
- If packet is the start of a new connection…
  - check against rules for new connections
- If packet is part of an existing connection…
  - check against state-based rules for existing connections
  - update state of this connection

Stateful Packet Filtering (cont'd)

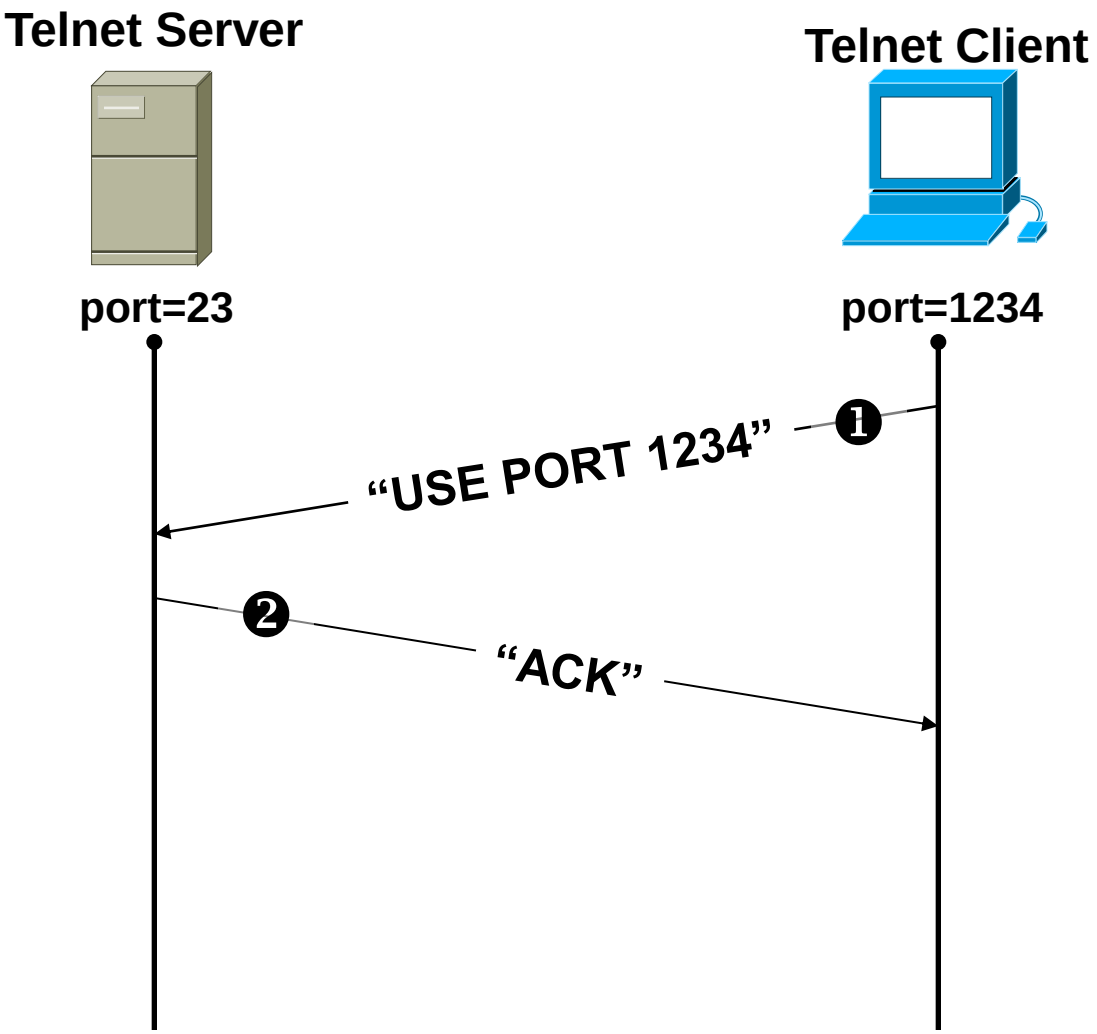
- Assessment
  - more powerful than packet filtering, can recognize more sophisticated threats or implement more complex policies
  - also more expensive to implement



Example: Telnet

❶ Client opens channel to server; tells server its port number. The ACK bit is not set when initiating the connection but will be set on the remaining packets.

❷ Server acknowleges.



Example: Firewall Access for Telnet

Format:

access-list *<rule number>*
<permit|deny>
<protocol>
<SOURCE host with IP address| any|IP address and mask>
[<gt|eq port number>]
<DEST host with IP address| any|IP address and mask>
[<gt|eq port number>]

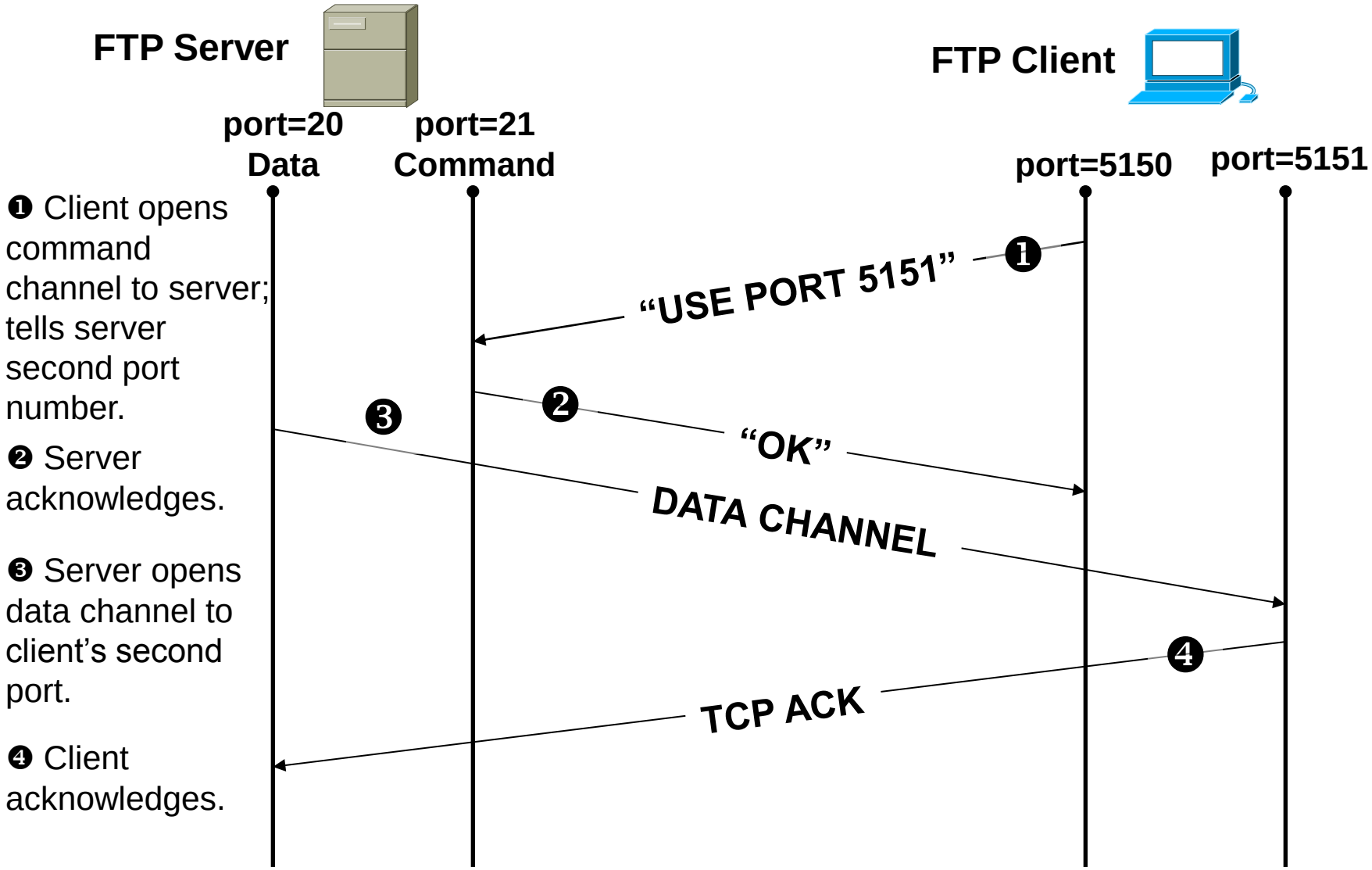
***Note*: any packets not explicitly permitted in an access list assumed to be denied or dropped.**

The following allows user to telnet from an IP address (172.168.10.11) to any destination, but not vice-versa:

```
access-list 100 permit tcp host 172.168.10.11 gt 1023 any eq 23
  ! Allows packets out to remote Telnet servers
access-list 101 permit tcp any eq 23 host 172.168.10.11 established
  ! Allows returning packets to come back in.  It verifies that the ACK bit is set

interface Ethernet 0
 access-list 100 out   ! Apply the first rule to outbound traffic
 access-list 101 in    ! Apply the second rule to inbound traffic
```

Example: FTP


FTP Server
FTP Client
port=20
Data
port=21
Command
port=5150
port=5151
❶ Client opens command channel to server; tells server second port number.
❷ Server acknowledges.
❸ Server opens data channel to client's second port.
❹ Client acknowledges.
"USE PORT 5151"
❶
❷
"OK"
❸
DATA CHANNEL
❹
TCP ACK

Example: Firewall Access for FTP

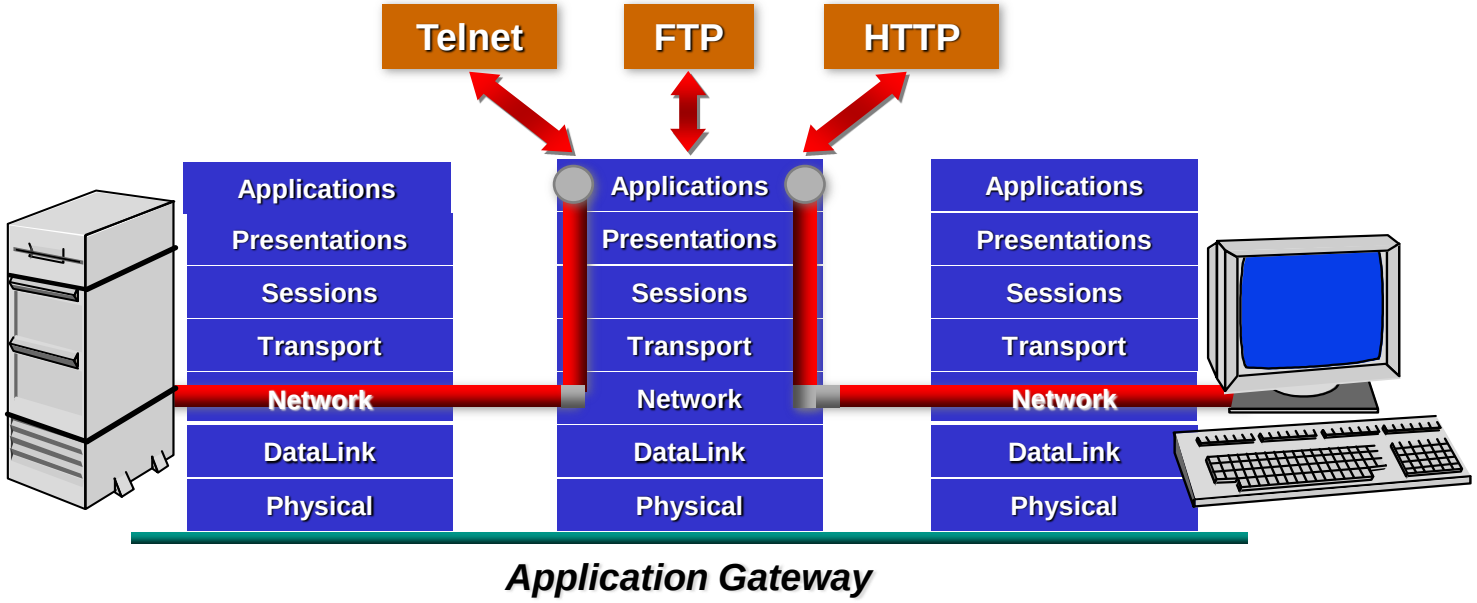
Allow a user to FTP (not passive FTP) from any IP address to the FTP server (172.168.10.12) :

```
access-list 100 permit tcp any gt 1023 host 172.168.10.12 eq 21
access-list 100 permit tcp any gt 1023 host 172.168.10.12 eq 20
  ! Allows packets from any client to the FTP control and data ports
access-list 101 permit tcp host 172.168.10.12 eq 21 any gt 1023
access-list 101 permit tcp host 172.168.10.12 eq 20 any gt 1023
  ! Allows FTP server to send packets back to any IP address with TCP ports > 1023

interface Ethernet 0
 access-list 100 in     ! Apply the first rule to inbound traffic
 access-list 101 out    ! Apply the second rule to outbound traffic
```

Proxy Firewalls

- Serve as *relays* for connections
- Two flavors
  1. application level
  2. circuit level



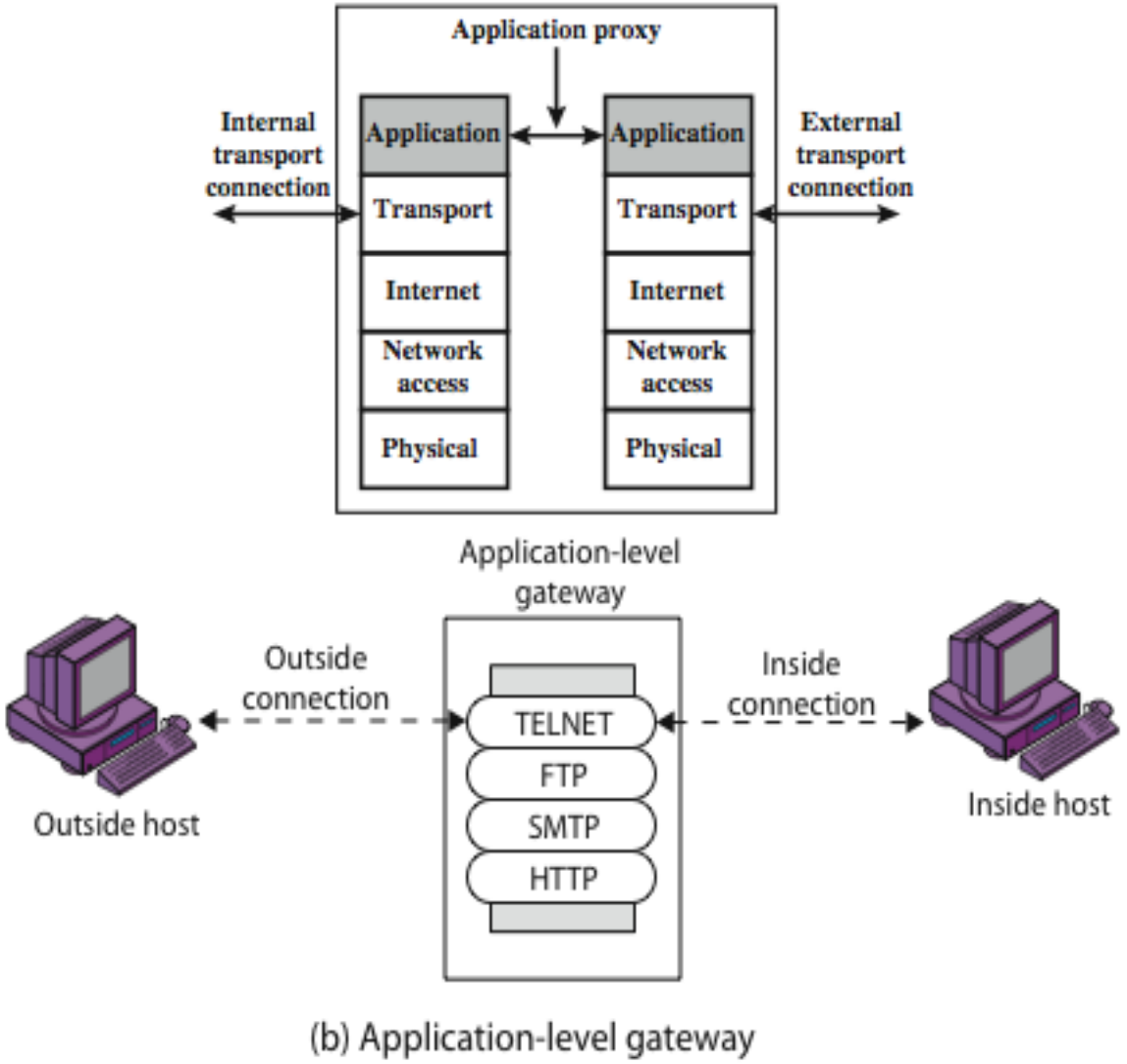
Firewalls - Application Level Gateway

- Have application specific gateway / proxy
- Has full access to protocol
  - User requests service from proxy
  - Proxy validates request as legal
  - Then actions request and returns result to user
  - Can log / audit traffic at application level
- Need separate proxies for each service
  - Some services naturally support proxying
  - Others are more problematic
- Application-level gateways tend to be more secure than packet filters, and can log and audit traffic at application level

Application Proxies

- Understand specific application protocols, e.g., HTTP, SMTP, Telnet
  - Proxy ‘impersonates’ both one side of connection to the other
- Can do arbitrary processing / inspection of application payloads
  - Example: check mail for viruses before forwarding
- Computationally expensive
- Must write a new proxy application to support new protocols
- May require hosts inside the organization to be configured to use the proxy

Firewalls - Application Level Gateway

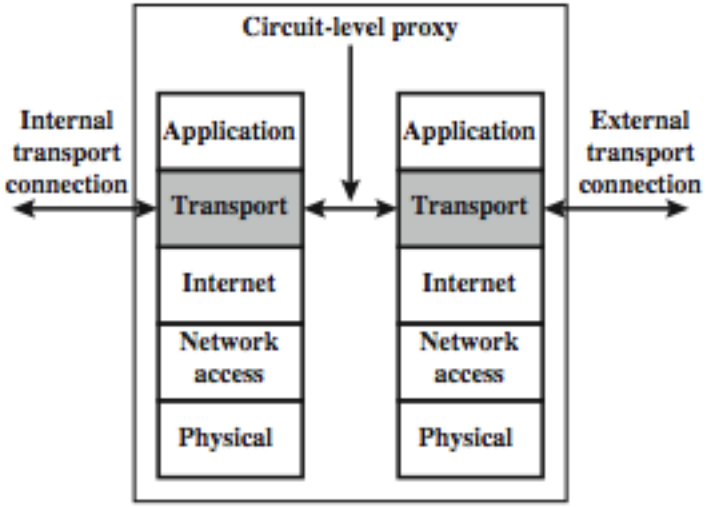


(b) Application-level gateway

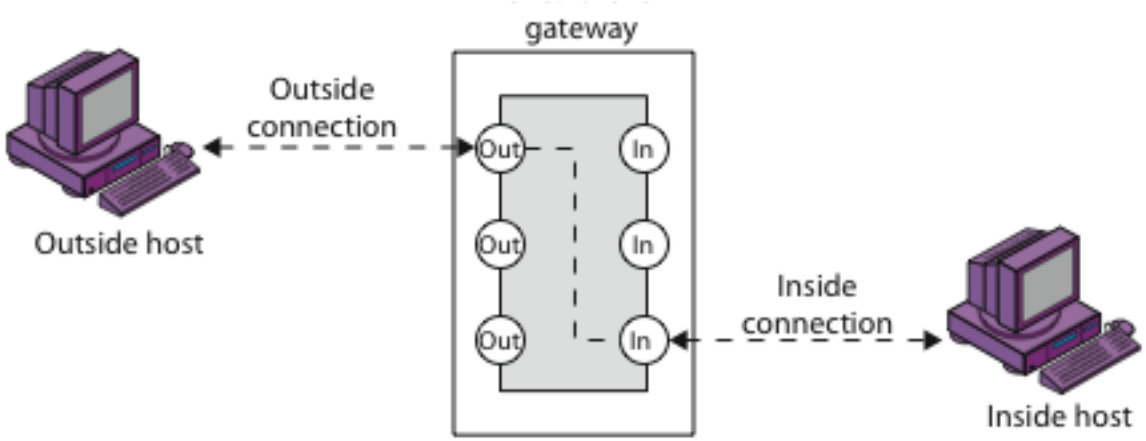
Firewalls - Circuit Level Gateway

- Relays two TCP connections (proxy at the TCP level, rather than the application level)
  - One between itself and an inside TCP user,
  - The other between itself and a TCP user on an outside host
  - Once the two connections are established, it relays TCP data from one connection to the other without examining its contents.
- Imposes security by limiting which such connections are allowed
- Once created usually relays traffic without examining contents
- Typically used when trust internal users by allowing general outbound connections
- SOCKS is commonly used
  - Client must open a TCP connection to the appropriate SOCKS port on the SOCKS server system.
  - If the connection request succeeds, the client enters a negotiation for the authentication method to be used, authenticates with the chosen method, and then sends a relay request.
  - The SOCKS server evaluates the request and either establishes the appropriate connection or denies it. UDP exchanges are handled in a similar fashion.

Firewalls - Circuit Level Gateway



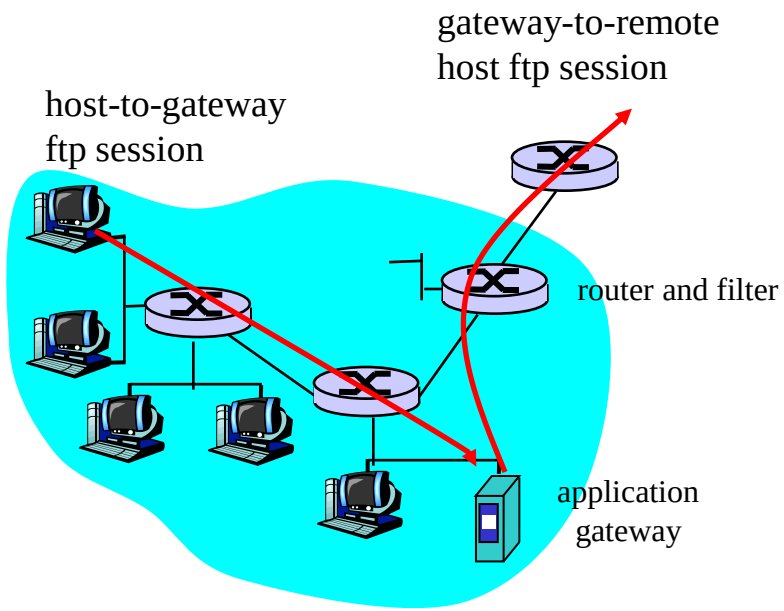
(e) Circuit-level proxy firewall



(c) Circuit-level gateway

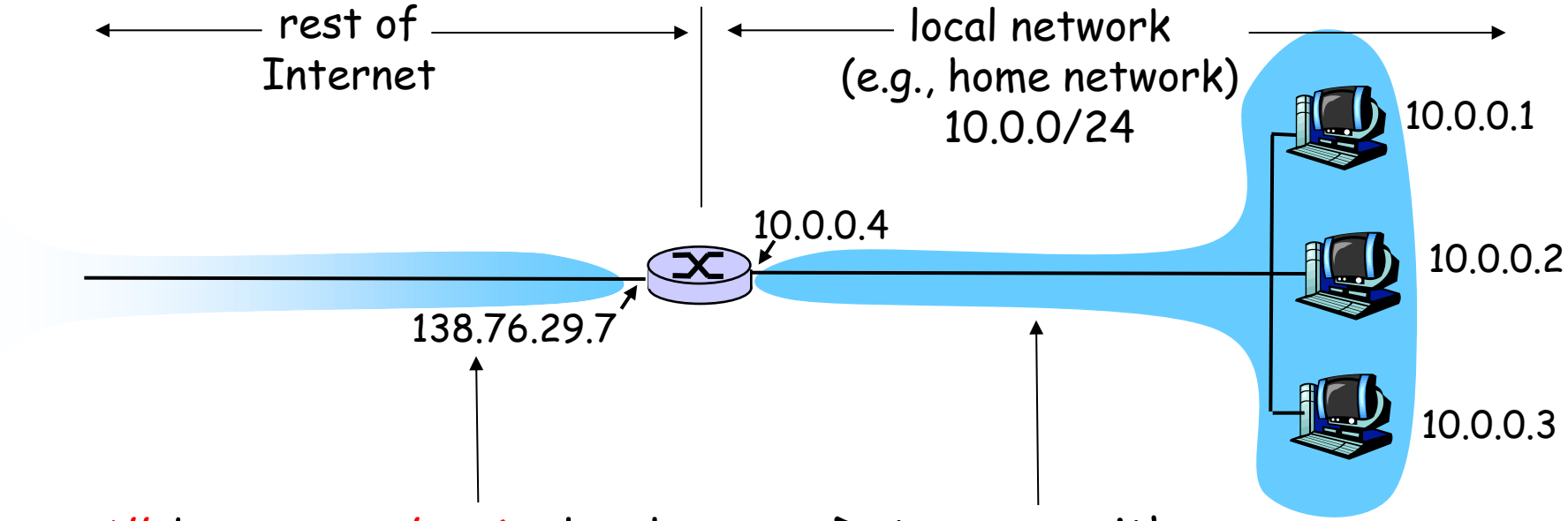
Application Gateways + Packet Filter

- Filters packets on application data as well as on IP/TCP/UDP fields.
- Example: allow select internal users to ftp outside.



1. Require all ftp users to ftp through gateway.
2. For authorized users, gateway sets up ftp connection to dest host. Gateway relays data between 2 connections
3. Router filter blocks all ftp connections not originating from gateway.

Example: Network Address Translation



All datagrams *leaving* local network have same single source NAT IP address: 138.76.29.7, different source port numbers

Datagrams with source or destination in this network have 10.0.0/24 address for source, destination (as usual)

10.0.0.0/8 has been reserved for private networks!

NAT: Network Address Translation

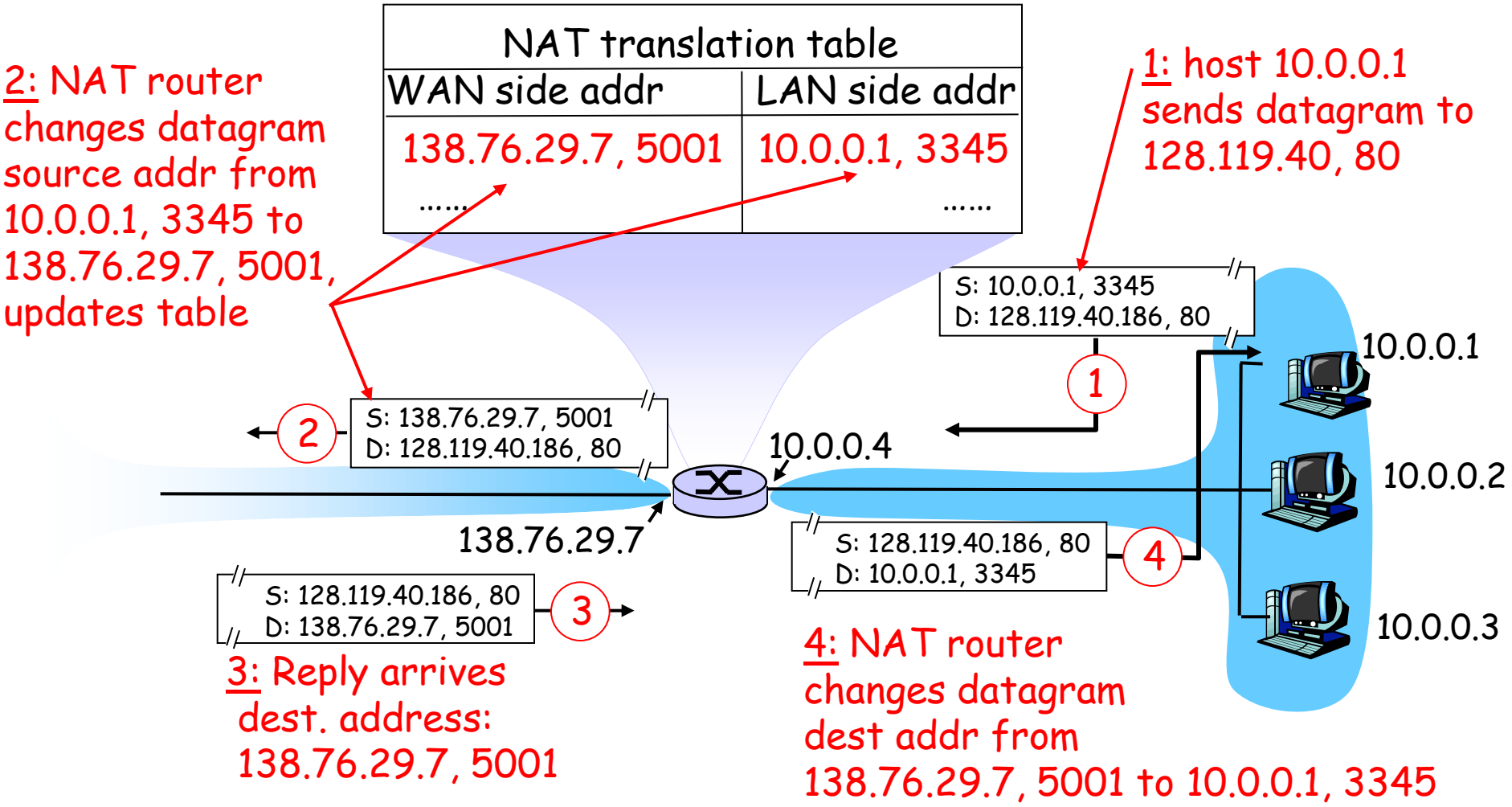
- Motivation: local network uses just one IP address as far as outside world is concerned:
  - no need to be allocated range of addresses from ISP: - just one IP address is used for all devices
  - can change addresses of devices in local network without notifying outside world
  - can change ISP without changing addresses of devices in local network
  - devices inside local net not explicitly addressable, visible by outside world (a security plus).

NAT: Network Address Translation

Implementation: NAT router must:

- *outgoing datagrams: replace* (source IP address, port #) of every outgoing datagram to (NAT IP address, new port #)
  - . . . remote clients/servers will respond using (NAT IP address, new port #) as destination addr.
- *remember (in NAT translation table)* every (source IP address, port #) to (NAT IP address, new port #) translation pair
- *incoming datagrams: replace* (NAT IP address, new port #) in dest fields of every incoming datagram with corresponding (source IP address, port #) stored in NAT table

NAT: Network Address Translation



Bastion Host

- Highly secure host system
  - Executes a secure version of its O/S, making it a trusted system
  - Has as only essential services installed on the bastion host
  - May require additional authentication before a user may access to proxy services
  - Configured to use only subset of standard commands, access only specific hosts
  - Maintains detailed audit information by logging all traffic
  - Each proxy module a very small software package designed for network security
  - Has each proxy independent of other proxies on the bastion host
  - Have a proxy performs no disk access other than read its initial configuration file
  - Have each proxy run as a non-privileged user in a private and secured directory
- Runs circuit / application level gateways or provides externally accessible services
- Potentially exposed to "hostile" elements , hence is secured to withstand this
- May support 2 or more net connections
- May be trusted to enforce policy of trusted separation between these net connections

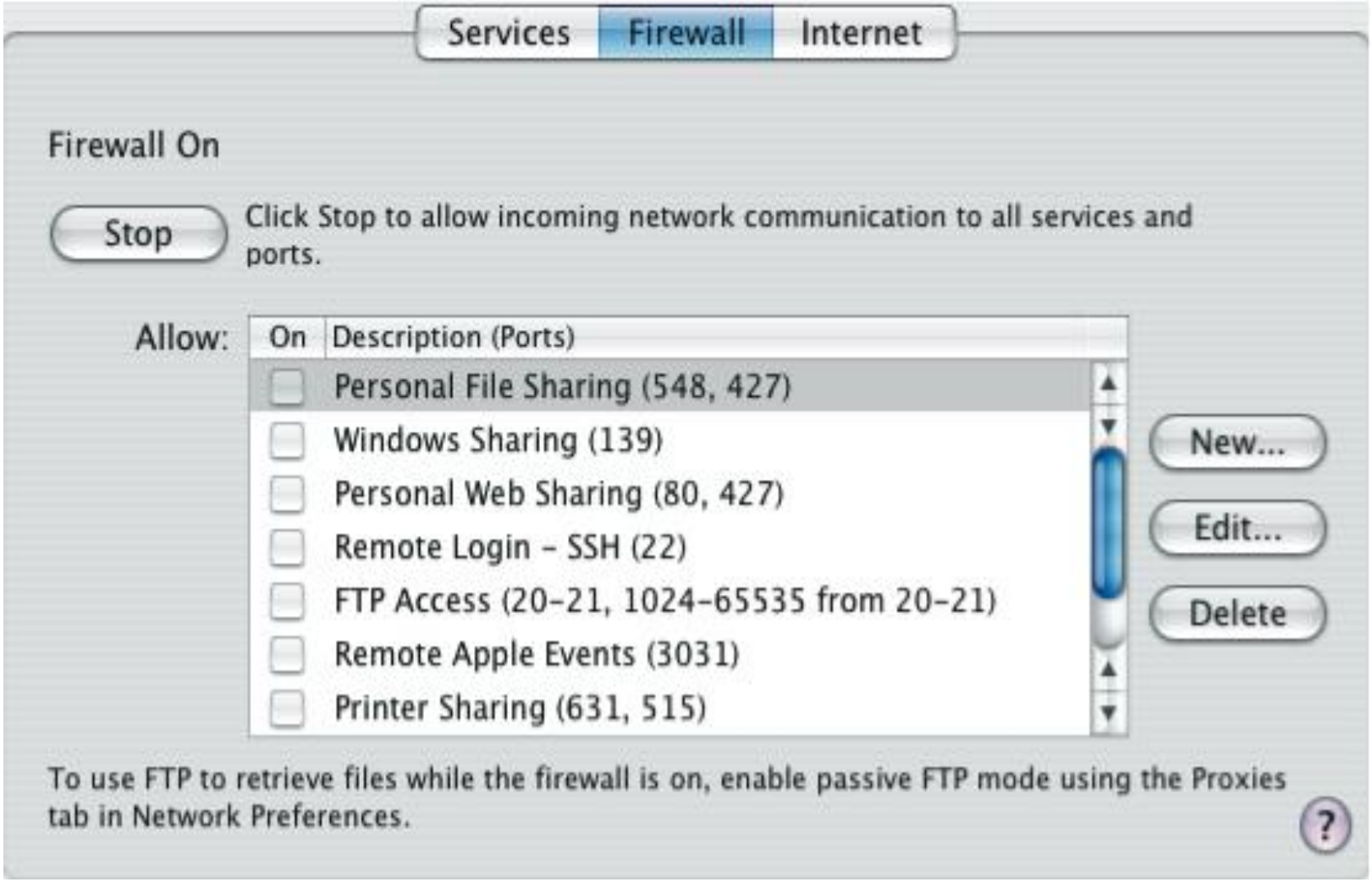
Host-Based Firewalls

- Software module used to secure individual host
  - Available in many operating systems
  - Or can be provided as an add-on package
- Often used on servers
- Advantages:
  - Can tailor filtering rules to host environment
  - Protection is provided independent of topology
  - Provides an additional layer of protection

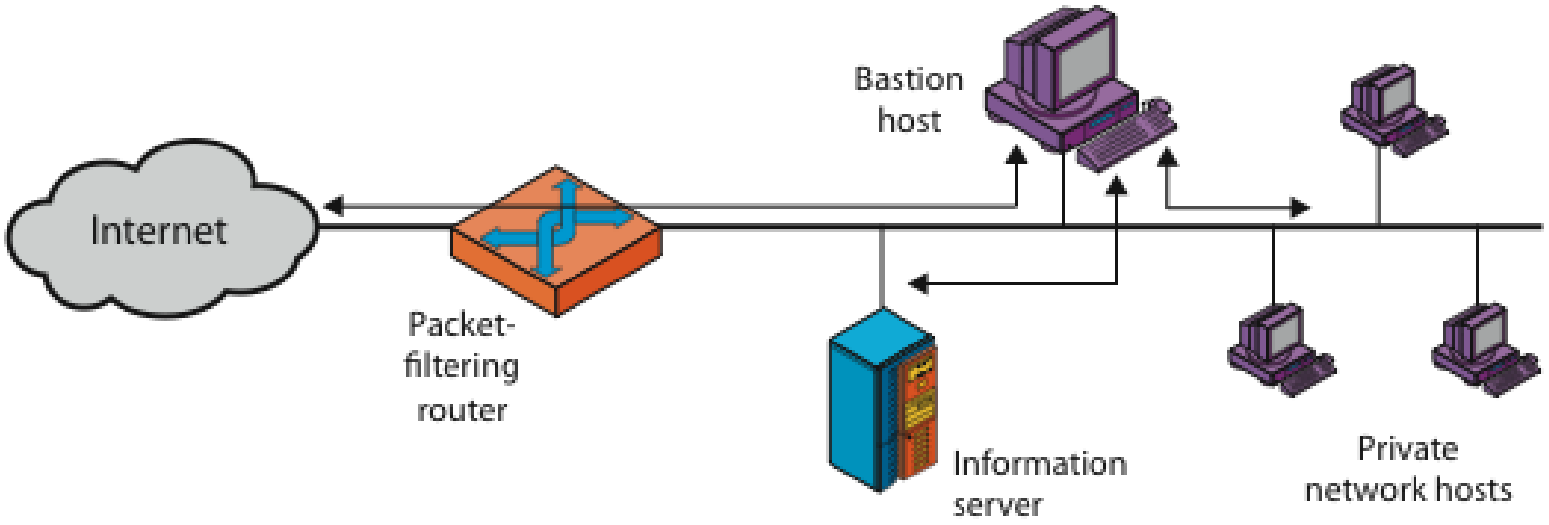
Personal Firewalls

- Controls traffic between PC/workstation and Internet or enterprise network
- In a home environment with multiple computers connected to the Internet, firewall functionality can also be housed in a router that connects all of the home computers to a DSL, cable modem, or other Internet interface
  - A software module on personal computer or in home/office DSL/cable/ISP router
- Typically much less complex than other firewall types
- Primary role to deny unauthorized remote access to the computer and monitor outgoing activity for malware

Personal Firewalls

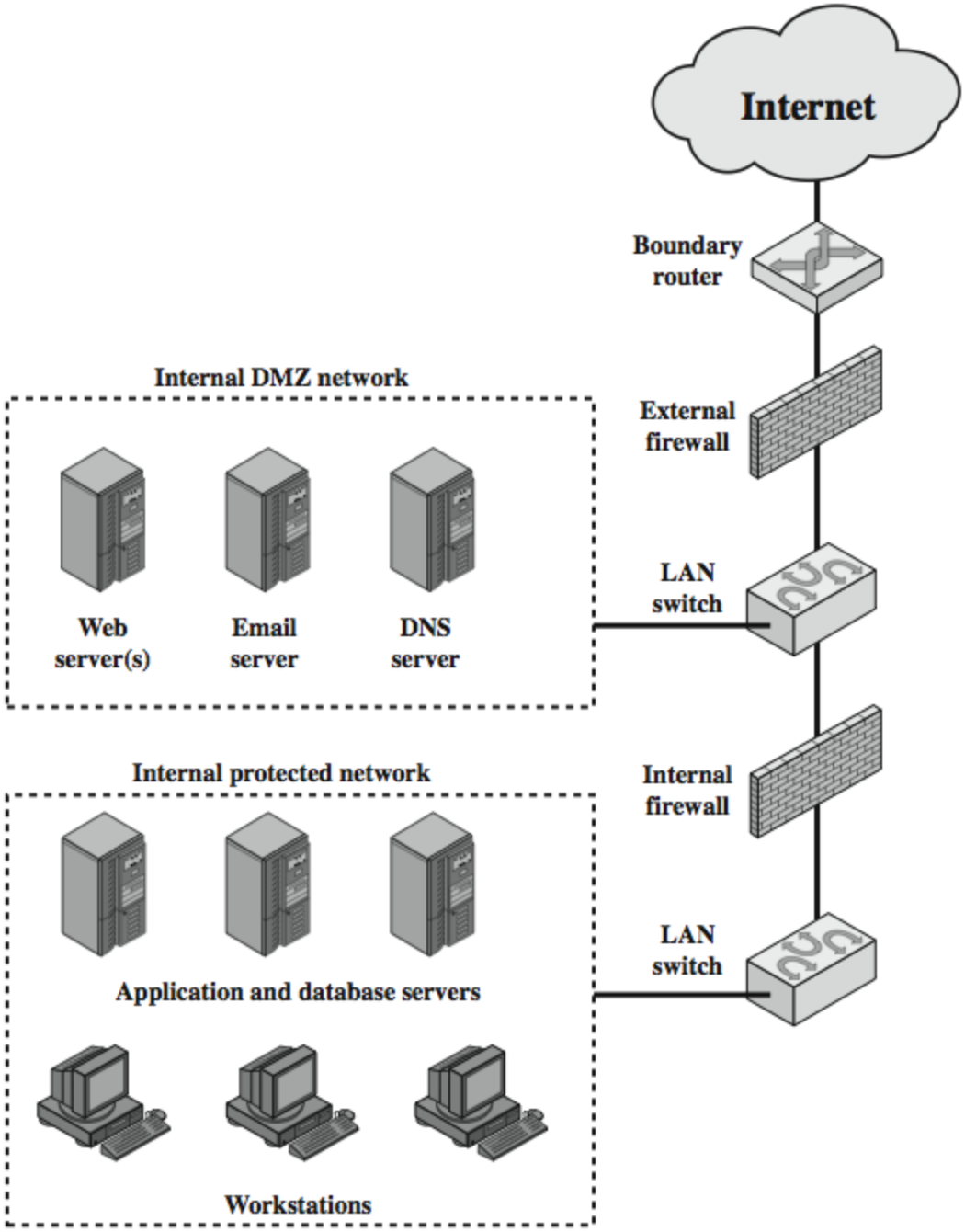


Firewall Configurations

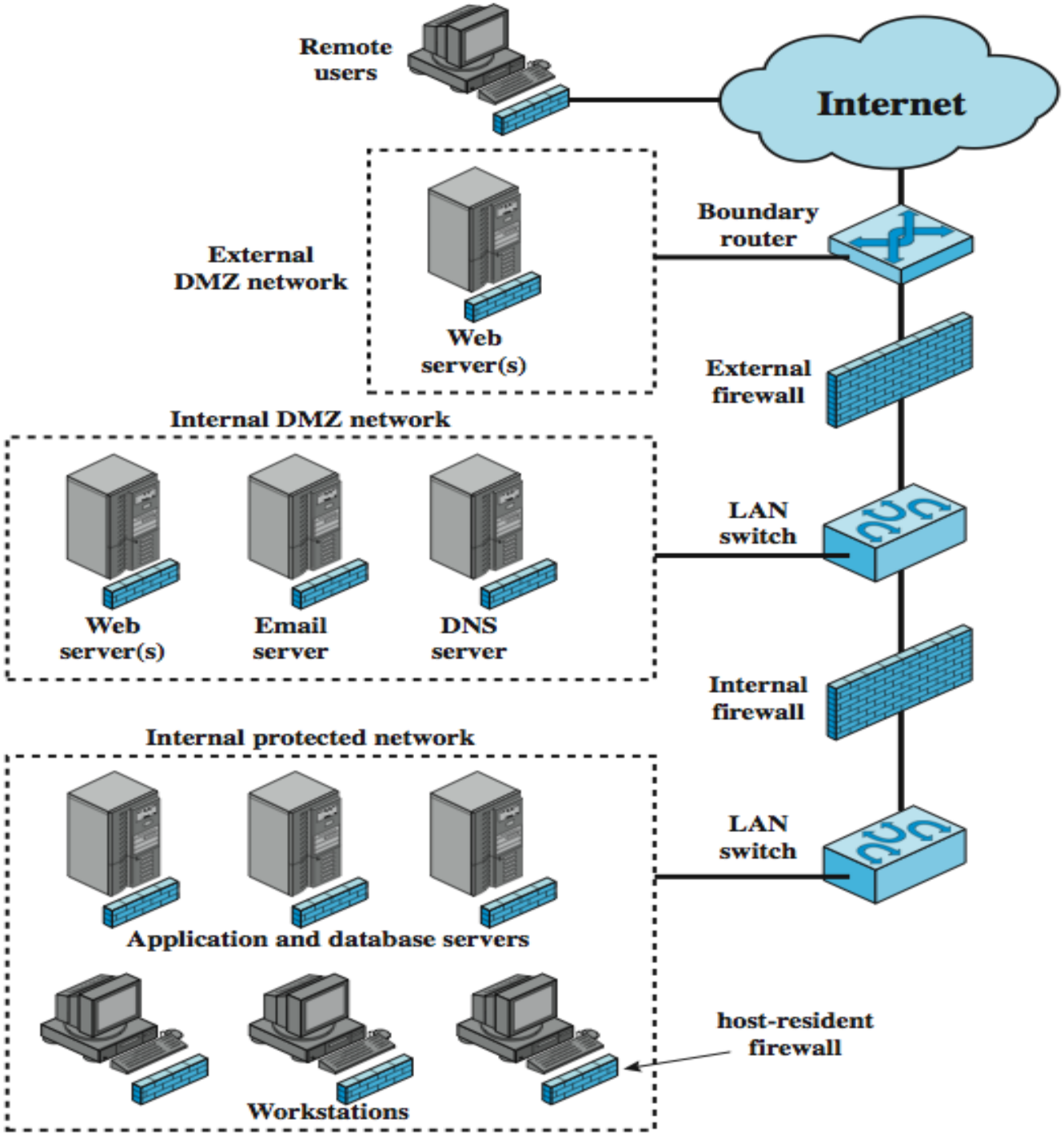


(a) Screened host firewall system (single-homed bastion host)

DMZ Networks



Distributed Firewalls



Examples of Firewall Locations

- **Host-resident firewall**: include personal firewall software and firewall software on servers, used alone or as part of an in-depth firewall deployment.
- **Screening router:** A single router between internal and external networks with stateless or full packet filtering. Typical for small office/home office (SOHO) use.
- **Single bastion inline:** A single firewall device between an internal and external router. The firewall may implement stateful filters and/or application proxies. This is the typical firewall appliance configuration for small to medium-sized organizations.
- **Single bastion T:** Similar to single bastion inline but has a third network interface on bastion to a DMZ where externally visible servers are placed. This is a common appliance configuration for medium to large organizations.
- **Distributed firewall configuration**: This configuration is used by some large businesses and government organizations.

Limitations of Firewalls

- Cannot protect against traffic that does not cross it
  - i.e., there may be other ingress points to the network, such as modems or wireless access points, that bypass the firewall
  - doesn't protect against "inside" attacks
- Configuration of firewalls to accomplish a desired high-level security policy is non-trivial
  - Open issues

Limitations of Firewalls

- Cannot protect against traffic that does not cross it
  - i.e., there may be other ingress points to the network, such as modems or wireless access points, that bypass the firewall
  - doesn't protect against "inside" attacks
- Configuration of firewalls to accomplish a desired high-level security policy is non-trivial
  - Open issue
    - Too many rules -> performance issues
    - How to validate firewall rules and consolidate them

  Examples:

  http://www.cse.msu.edu/~alexliu/publications/FirewallCompressor/FirewallCompressor_TPDS.pdf
  http://www.cse.msu.edu/~alexliu/publications/Redundancy/RedundancyTPDS.pdf
  http://www.cse.msu.edu/~alexliu/publications/FirewallVerification/verificationjournal.pdf

Firewall Compressing

- Access control lists (ACLs) represent a critical component of network security
  - Each rule in an ACL has a predicate over some packet header fields and a decision to be performed upon the packets that match the predicat
  - Real-life ACLs are typically four dimensional
    - Four dimensions: Source IP address, destination IP address, destination port number, and protocol type)
    - Five dimensional (source IP address, destination IP address, source port number, destination port number, and protocol type)

A. X. Liu, E. Torng, and C. R. Meiners, "Compressing Network Access Control Lists," IEEE TRANSACTIONS ON PARALLEL AND DISTRIBUTED SYSTEMS, VOL. 22, NO. 12, DECEMBER 2011

Firewall Compressing (cont.)

- When a packet comes to an ACL, the network device searches for the first (i.e., highest priority) rule that the packet matches, and executes the decision of that rule
- Two ACLs are equivalent if and only if they have the same decision for every possible packet

An Example ACL

Rule	SIP	DIP	SPort	DPort	Proto	Act
1	192.168.*.*	1.2.3.*	*	[4000, 5000]	TCP	discard
2	192.168.*.*	1.2.3.*	*	[0, 3999]	TCP	accept
3	192.168.*.*	1.2.3.*	*	[5001, 65535]	TCP	accept
4	*	*	*	*	*	discard

A. X. Liu, E. Torng, and C. R. Meiners, "Compressing Network Access Control Lists," IEEE TRANSACTIONS ON PARALLEL AND DISTRIBUTED SYSTEMS, VOL. 22, NO. 12, DECEMBER 2011

Firewall Compressing (cont.)

- ACL compression is useful for network system management and optimization because minimizing large ACL rule sets greatly reduces the complexity of managing and optimizing network configurations
- Some network products have hard constraints on the number of rules that they support
  - For example, NetScreen-100 only allows ACLs with at most 733 rules.
- ACL compression may allow users with larger ACLs to still use such devices.
- This may become an increasingly important issue for many users as ACL size has grown dramatically due to an increase in Internet applications and services as well as an increase in known vulnerabilities, threats, and attacks

A. X. Liu, E. Torng, and C. R. Meiners, "Compressing Network Access Control Lists," IEEE TRANSACTIONS ON PARALLEL AND DISTRIBUTED SYSTEMS, VOL. 22, NO. 12, DECEMBER 2011

Firewall Compressing (cont.)

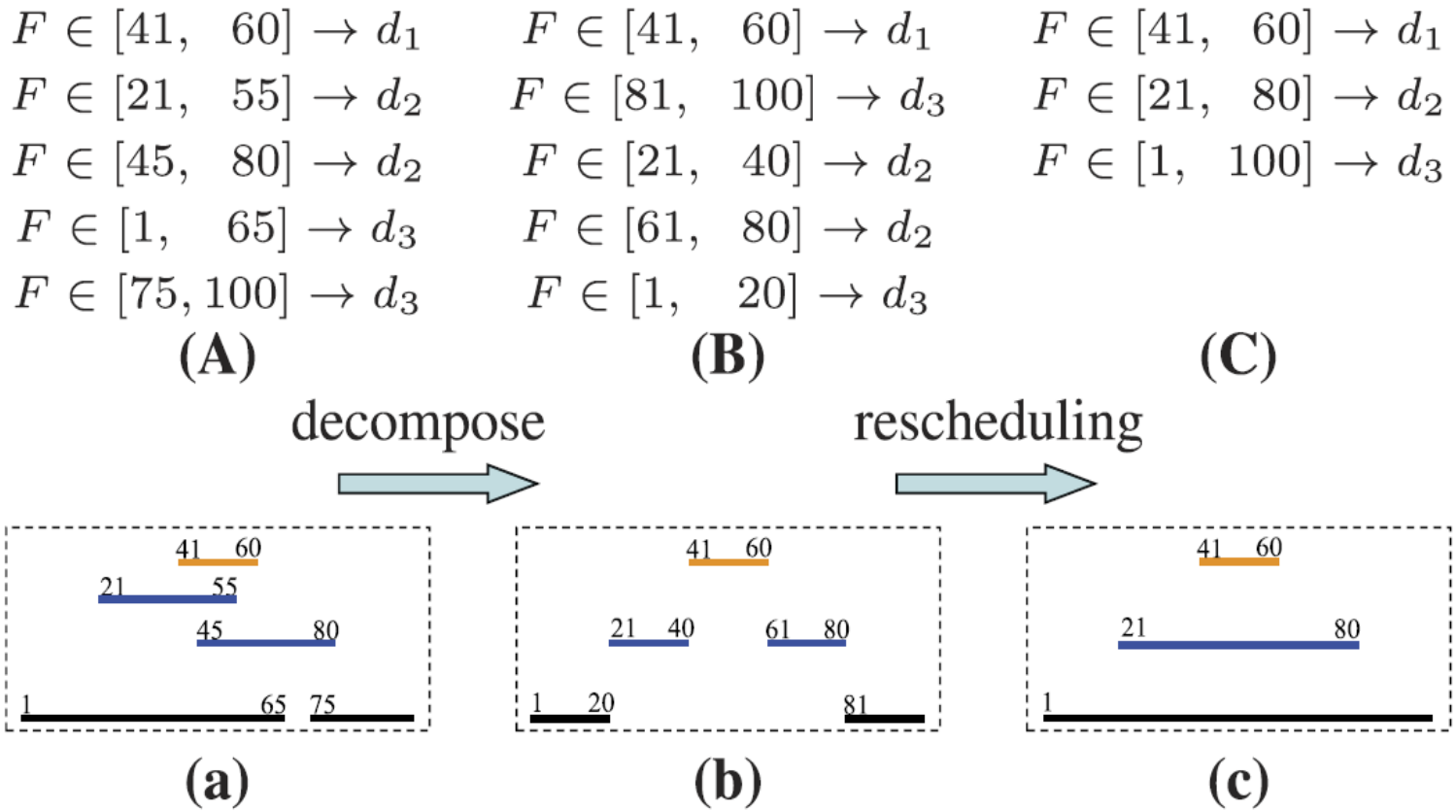


Fig. 1. Example minimization of an ACL.

A. X. Liu, E. Torng, and C. R. Meiners, "Compressing Network Access Control Lists," IEEE TRANSACTIONS ON PARALLEL AND DISTRIBUTED SYSTEMS, VOL. 22, NO. 12, DECEMBER 2011

Limitations of Firewalls

- Cannot protect against traffic that does not cross it
  - i.e., there may be other ingress points to the network, such as modems or wireless access points, that bypass the firewall
  - doesn't protect against "inside" attacks
- Configuration of firewalls to accomplish a desired high-level security policy is non-trivial
  - Open issue
    - Too many rules -> performance issues
    - How to validate firewall rules and consolidate them

  Examples:

  http://www.cse.msu.edu/~alexliu/publications/FirewallCompressor/FirewallCompressor_TPDS.pdf
  http://www.cse.msu.edu/~alexliu/publications/Redundancy/RedundancyTPDS.pdf
  http://www.cse.msu.edu/~alexliu/publications/FirewallVerification/verificationjournal.pdf

Summary

- have considered:
  - Firewalls
  - Types of firewalls
    - Packet-filter
    - Stateful inspection
    - Application proxy, circuit-level
  - Basing
    - bastion, host, personal
  - Location and configurations
    - DMZ, distributed, topologies