Network Security
(6CCS3NSE - 7CCSMNSE)

Diego Sempreboni
Department of Informatics
King's College London, UK

Second term 2019/20

Lecture 5

Objectives and learning outcomes

- To understand:
  - what is a firewall
  - what is a policy
  - how do firewalls operate
  - where do firewalls operate

Network bastioning

- Steady evolution of information systems in corporations government agencies and other organisation:
  - Now everyone wants to be on the Internet
  - And to interconnect networks

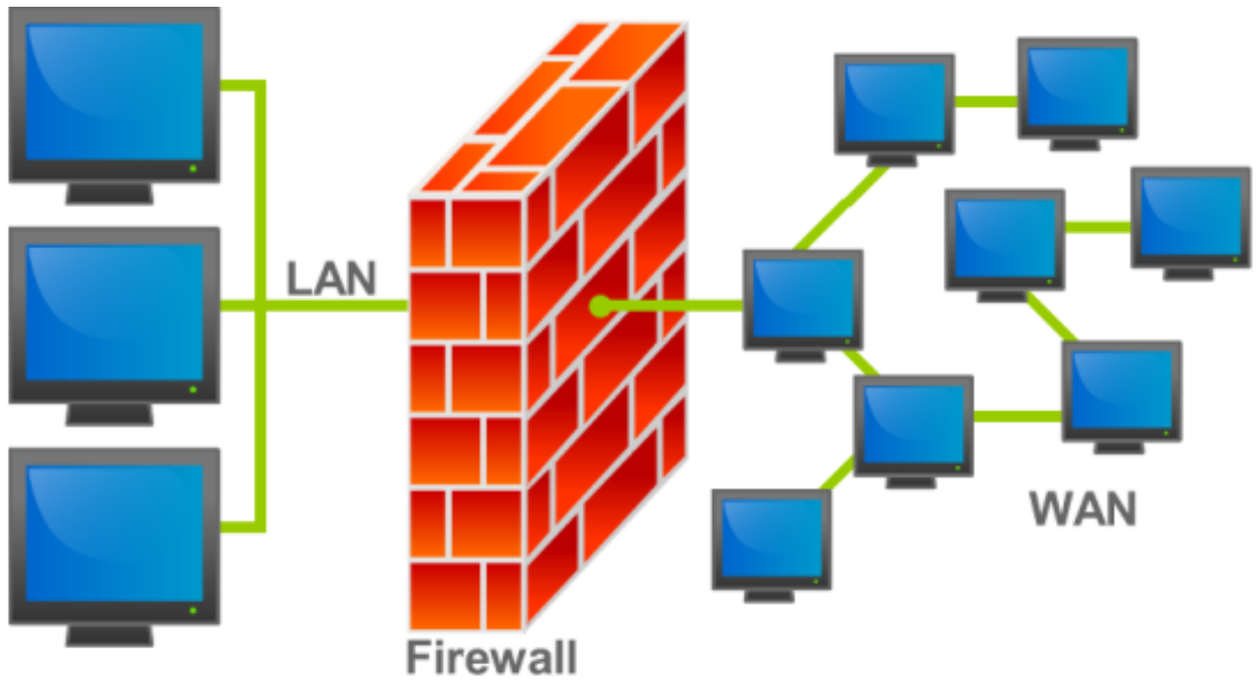
- Persistent security concerns:
  - Can’t easily secure every system in organisation

- Typically use a **firewall:**
  - To provide **perimeter defense** with single choke point
  - As part of comprehensive security strategy

Firewalls

- They can be used in bastion a network
- They can be used to feed alerts to advanced Intrusion Detection Systems (IDS)
- They can be used to implement Intrusion Prevention Systems (IPS)

What is a firewall?



- A firewall is a computer or a network security system:
  - That sits between your internal network and the rest of the network, and
  - Attempts to prevent bad things from happening (such as users sending company secrets outside, or outside people breaking into system inside)
  - Without preventing good things from happening (such as employees accessing information available externally)

What is a firewall?

- A **firewall** forms a barrier through which all the traffic going in each direction must pass.
- A **firewall security policy** dictates which traffic is authorised to pass in each direction
- A firewall is a **choke point** of control and monitoring
  - It may be designed to operate as a filter at the level of IP packets, or may operate at a higher protocol layer
- Interconnects network with differing trust
- Imposes restrictions on network services
  - Only authorised traffic is allowed
- Auditing and controlling access
  - Can implement alarms for abnormal behaviour
- Provides NAT and usage monitoring
  - Is a convenient platform for several Internet functions that are not security related, such as NAT and Internet usage audits or logs
- Implements VPNs using IPSec
- Must be **immune to penetration**

Techniques for firewalls to control access

- **Service control (what):** determines the types of Internet services that can be accessed, inbound or outbound

- **Direction control (where):** determines the direction in which particular service requests may be initiated and allowed to flow through the firewall

- **User control (who):** control access to a service according to which user is attempting to access it

- **Behaviour control (how):** controls how particular services are used

Firewall limitations

- Cannot protect:
  - from attacks bypassing it
  - against malware imported via laptop, PDA, storage infected outside
  - against access via WLAN if improperly secured against external use
  - against internal threats

Firewall limitations

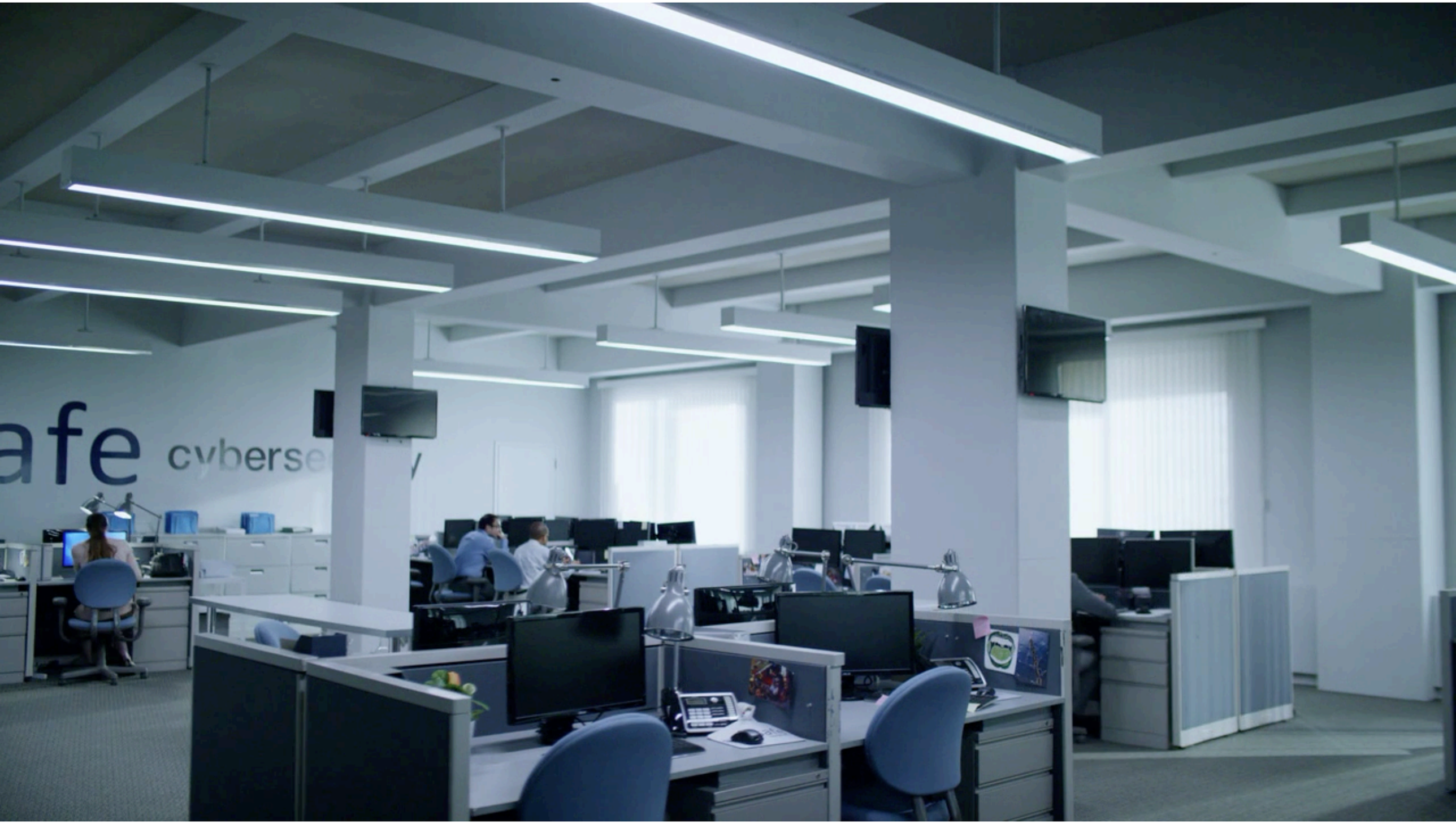
- A firewall cannot protect from attacks bypassing it or against malware imported via laptop, PDA, storage infected outside
- Example: sneakernet, utility modems, trusted organisations, trusted services (e.g., SSL/SSH,...)

Firewall limitations

- **Sneakernet** is an informal term describing the transfer of electronic information, especially computer files, by physically moving removable media such as magnetic tape, floppy disks, compact discs, USB flash drives (thumb drives, USB stick) or external hard drives from one computer to another, usually in lieu of transmitting the information over a computer network. The term, a tongue-in-cheek play on Ethernet, refers to the use of someone wearing sneakers as the transport mechanism for the data.

- There was a famous company that bragged about its security because none of its internal network was connected to the Internet…but curiosity killed the cat!
  - as he could not penetrate via the net, a social engineer loaded a virus on a usb key and threw the key over the perimeter wall of the company, so that it landed in the parking lot. Then he waited…
  - some employee saw the key in the parking lot (and thought “it is within the company’s perimeter, so it must have been lost by some colleague”), picked it up and plugged into his/her computer to see to whom it belonged, thereby infecting the whole company.

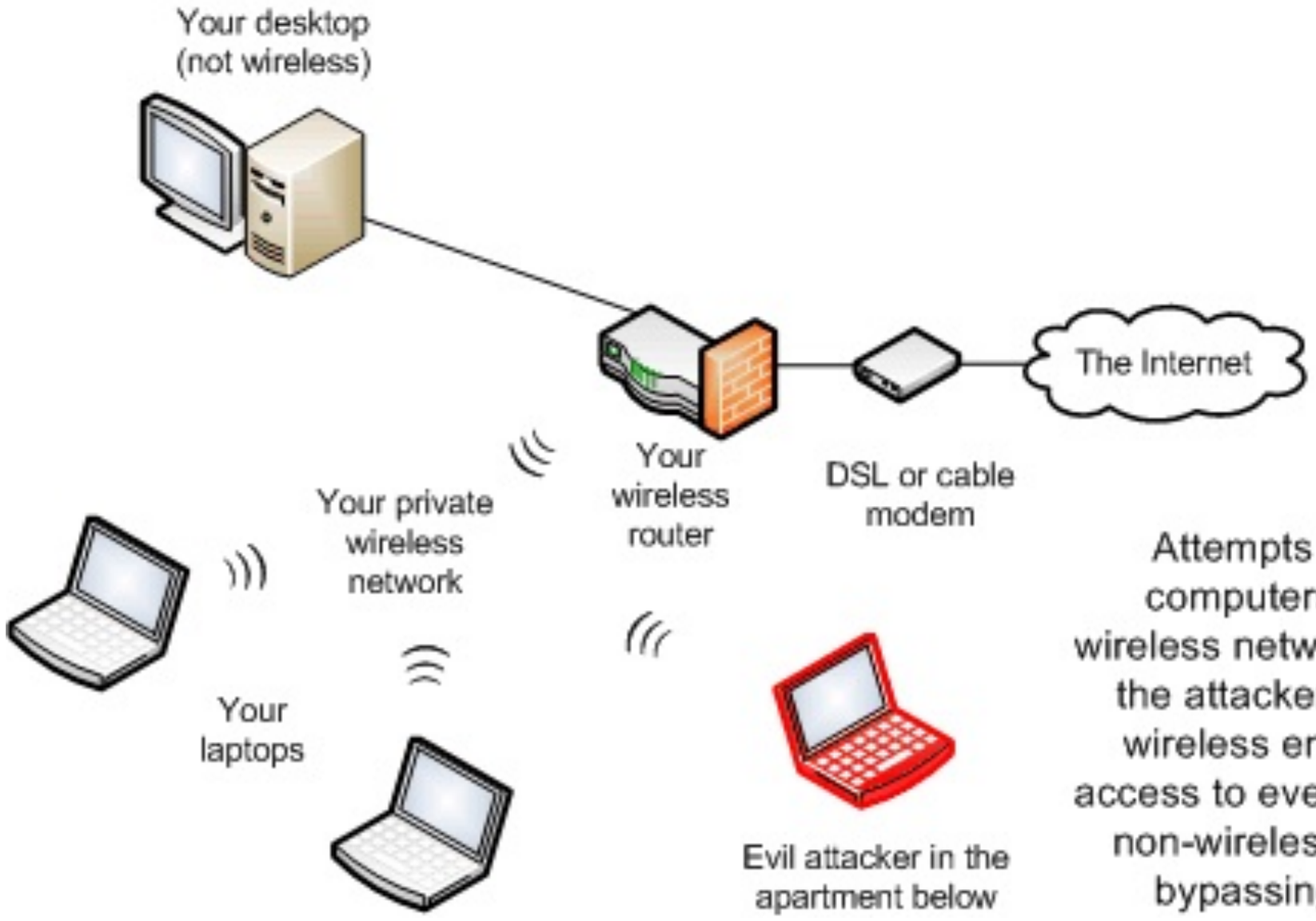
Mr. Robot: Sneakernet



Firewall limitations

Firewall limitations

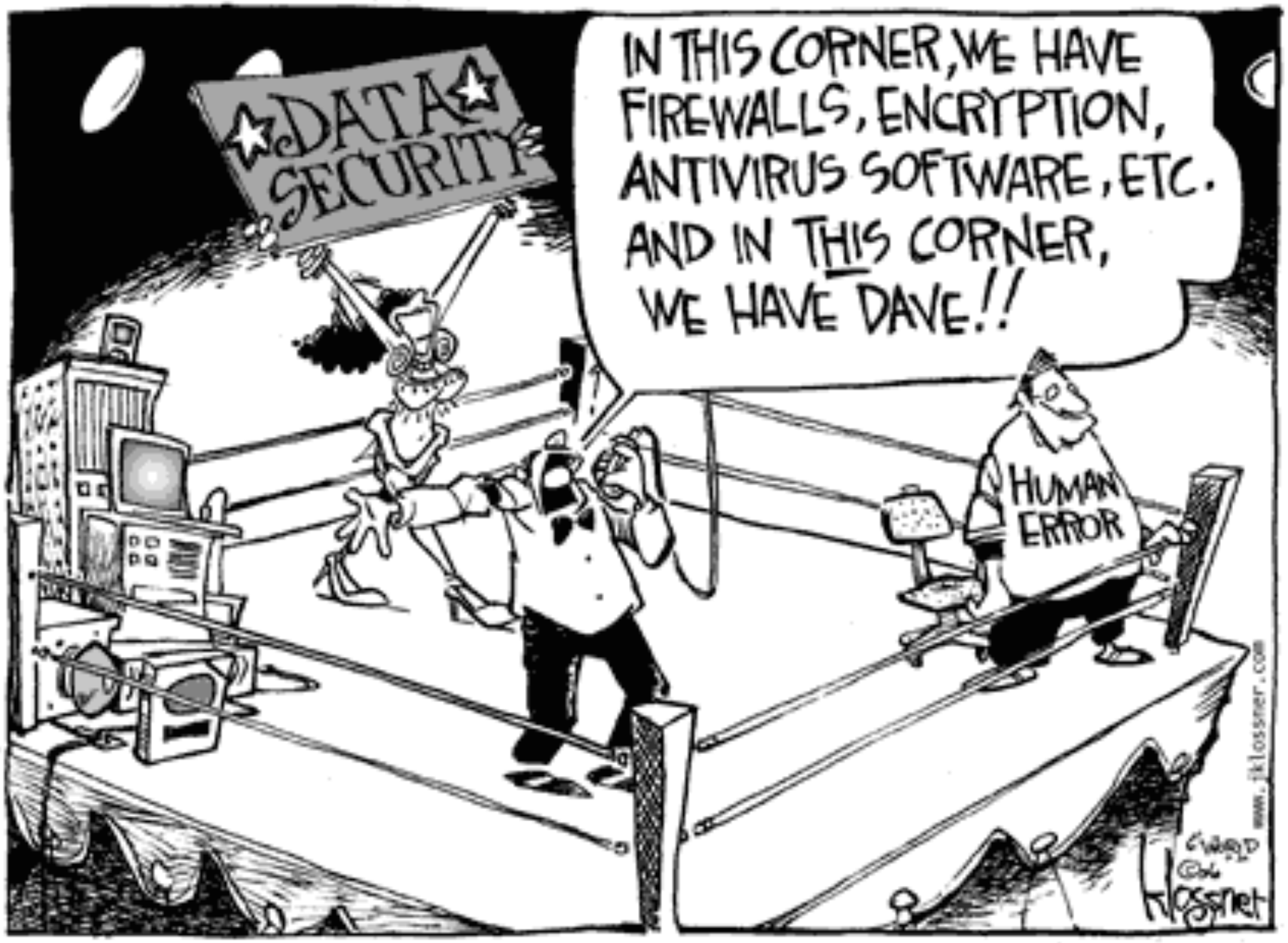
- A firewall cannot protect against access via WLAN if improperly secured against external use



Attempts to attack your computers through your wireless network can succeed if the attacker can crack your wireless encryption, giving access to everything, even your non-wireless computers, by bypassing your firewall

Firewall limitations

- A firewall cannot protect against internal threats, e.g., humans errors, disgruntled or colluding emplyees,...



Firewall limitations

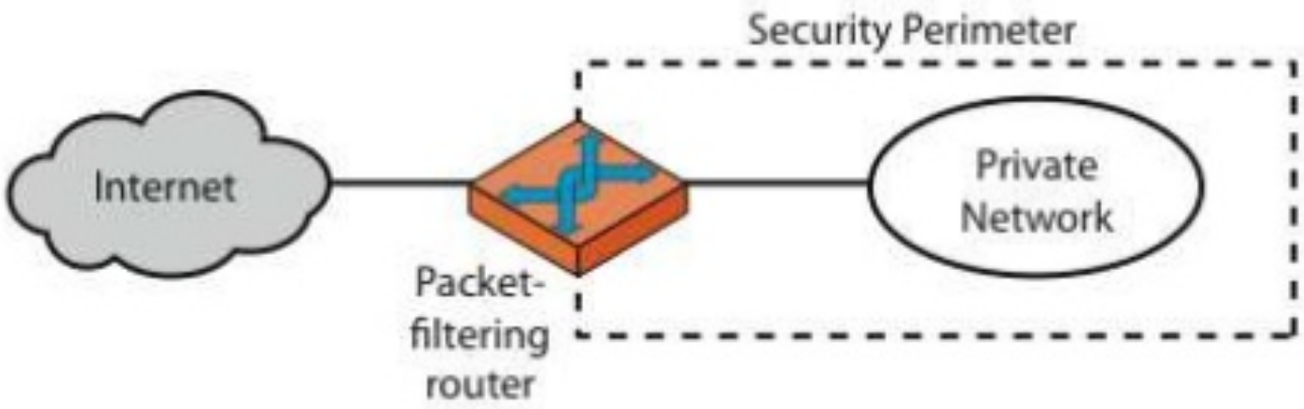
- A firewall cannot protect against internal threats

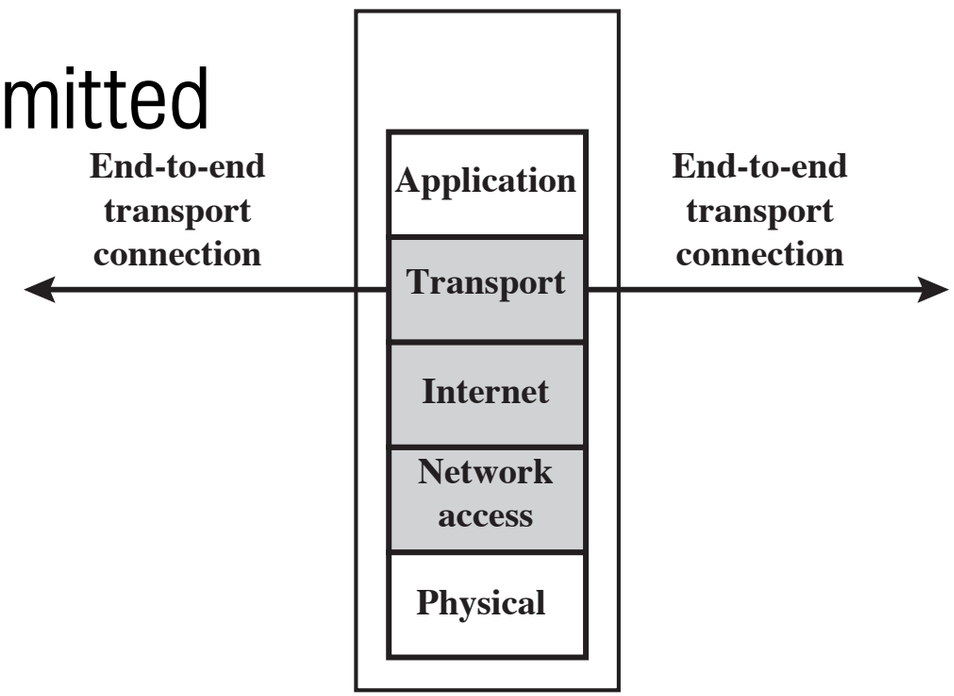
Firewall types

- A firewall can operate as
  - A positive filter, allowing to pass only packets that meet specific criteria, or
  - A negative filter, rejecting any packet that meets certain criteria
- Depending on the type of firewall, it may examine
  - One or more protocol headers in each packet,
  - The payload of each packet, or
  - The pattern generated by a sequence of packets

Packet-filter firewall

- Simplest, fastest firewall component
- Foundation of any firewall system (more complex/sophisticated firewall exist)
- Examine each IP packet (no context) and permit or deny according to rules
- Hence restrict access to services (ports)
- Possible default policies:
  - That not expressly permitted is prohibited
  - That not expressly prohibited is permitted





Packet-filter firewall

- A **packet-filter firewall** applies a set of rules to each incoming and outgoing IP packet and then forwards or discards the packet

- The firewall is typically configured to filter packets going in both directions (from and to the internal network)
- Filtering rules based on information contained in network packet:
  - **Source IP address:** the IP address of the system that originated the IP packet (e.g., 192.168.1.1)
  - **Destination IP address:** the IP address of the system the IP packet is trying to reach (e.g, 192.168.1.2)
  - **Source and destination transport-level address:** the transport-level (e.g., TCP or UDP) port number, which defines applications such as SNMP or TELNET
  - **IP protocol field:** defines the transport protocol
  - **Interface:** for the firewall with three or more ports, which interface of the firewall the packet came or which interface of the firewall the packet is destined for

Packet-filter firewall

- What if we want to block an entire range of its or a given sub-network? The answer is **network masks!**

Abbreviation	Address	Netmask
/27	32	255.255.255.224
/26	64	255.255.255.192
/25	128	255.255.255.128
192.168.1.0/24	256	255.255.255.0

Packet-filter firewall: example rules

- Packet-filter firewall is typically set up as a list of rules based on matches to fields in IP or TCP header.
- In each set, rules are applied top to bottom.
- If there is a match to one of the rules, that rule is invoked to determine whether to forward or discard the packet.
- If there is no match to any rule, then a default action is taken.
  Two default policies are possible.

Rule Set A

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

Rule Set B

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

Rule Set C

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

Rule Set D

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

Rule Set 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

Packet-filter firewall: default policies

- **Default = discard:** that which is not expressly permitted in prohibited.
  - More conservative policy: initially, everything is blocked, and services must be added on a case-by-case basis.
  - This policy is more visible to users, who are more likely to see the firewall as a hindrance.
  - However, this is the policy likely to be preferred by businesses and government organisations.
  - Further, visibility to users diminishes as rules are created.
- **Default = forward:** that which is not expressly prohibited is permitted
  - Default forward policy increases ease of use for end users but provides reduced security; the security administrator must, in essence, react to each new security threat as it becomes known.
  - This policy may be used by generally more open organisations, such as universities.

Packet-filter firewall: example rules

- **Granularity** depending on the type of router, filtering may be done:
  - At input time,
  - At output time, or
  - both; and
  - By looking at control fields.
- Rule Set A
  - No direction applies
  - Rule #A.1 blocks outbound packets to **theirhost** (SPIGOT)
  - Rule #A.1 blocks inbound packets from **theirhost** (SPIGOT)

Rule Set A

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

Rule Set B

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

Rule Set C

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

Rule Set D

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

Rule Set 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

Packet-filter firewall: example rules

- * = ANY; default = discard

A. Inbound mail is allowed (port 25 is for SMTP incoming), but only to a gateway host.
However, packets from a particular external host, SPIGOT, are blocked (e.g., because that host has a history of sending massive files in e-mail messages).

B. This is an explicit statement of the default = discard policy.
All rule sets include this rule implicitly as the last rule.

Rule Set A

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

Rule Set B

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

Rule Set C

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

Rule Set D

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

Rule Set 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

Packet-filter firewall: example rules

C. This specifies that any inside host can send mail to the outside.

- A TCP packet with a destination port of 25 is routed to SMTP server on destination machine.
- The problem with this rule is that the use of port 25 for SMTP receipt is only a default; an outside machine could be configured to have some other application linked to port 25.
- As rule C is written, an attacker could gain access to internal machines by sending packets with a TCP source port number of 25.

Rule Set A

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

Rule Set B

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

Rule Set C

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

Rule Set D

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

Rule Set 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

Packet-filter firewall: example rules

D. Archives intended result that was not achieved in C.

- D exploits a feature of TCP connections: once a connection is set up, ACK flag of a TCP segment is set to acknowledge segments sent from other side.
- Thus, rule set D allows IP packets where source IP address is one of a list of designated internal hosts and the destination TCP port number is 25.
- It also allows incoming packets with a source port number of 25 that include ACK flag in TCP segment.
- We explicitly designate source and destination systems to define these rules explicitly.

Rule Set A

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

Rule Set B

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

Rule Set C

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

Rule Set D

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

Rule Set 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

Packet-filter firewall: example rules

E. This rule set is one approach to handling FTP connections.

- With FTP, 2 TCP connections are used:
  - Control connection to set up file transfer,
  - Data connection for actual file transfer.
- Data connection uses a different port number assigned dynamically for transfer.
- Most servers, and hence most attack targets, use low-numbered ports; most outgoing calls tend to use a higher-numbered port, typically above 1023.

Rule Set A

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

Rule Set B

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

Rule Set C

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

Rule Set D

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

Rule Set 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

Packet-filter firewall: example rules

- Thus, rule set E allows
  - Packet that originate internally
  - Reply packets to a connection initiated by an internal machine
  - Packets destined for a high-numbered port on an internal machine
- This scheme requires that the systems be configured so that only the appropriate port numbers are in use.
- Rule set E points out difficulty in dealing with applications at packet filtering level.
  Another way to deal with FTP and similar applications is either stateful packet filters or an application-level gateway (see below).

Rule Set A

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

Rule Set B

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

Rule Set C

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

Rule Set D

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

Rule Set 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

Packet-filter firewall: advantages

- Very simple
- Typically are transparent to users and are very fast.

Packet-filter firewall: disadvantages

- No examination of upper-layer data, hence no prevention of attacks that employ application-specific vulnerabilities or functions.
  - For example, a packet filter firewall cannot block specific application commands. If a packet filter firewall allows a given application, all functions available within that application will be permitted.
- Because of the limited information available to the firewall, the logging functionality present in packet filter firewalls is limited.
  - Packet-filter logs normally contain same information used to make access control decisions (source/destination addresses, traffic type).
- Most packet filter firewalls do not support advanced user authentication schemes (also due to lack of upper-layer functionality).

Packet-filter firewall: disadvantages

- Packet-filter firewalls are generally vulnerable to attacks and exploits that take advantage of problems within the TCP/IP specification and protocol stack, such as network layer address spoofing.
  - Many packet filter firewalls cannot detect a network packet in which the OSI Layer 3 addressing information has been altered.
  - Spoofing attacks are generally employed by intruders to bypass the security controls implemented in a firewall platform.
- Due to the small number of variables used in decisions, packet filter firewalls are susceptible to security breaches caused by improper configurations.
  - It is easy to accidentally configure a packet filter firewall to allow traffic types, sources, and destinations that should be denied based on an organisation's information security policy.

Packet-filter firewall: attacks and countermeasures

- **IP address spoofing:**

  The intruder transmits packets from the outside with a source IP address field containing an address of an internal host.

  - Fake source address to be trusted (intruder transmits packets from the outside with internal host source IP address).
  - Attacker hopes that use of a spoofed address will allow penetration of systems that employ simple source address security, in which packets from specific trusted internal hosts are accepted.

- **Countermeasure:**

  - Add filters on router to block such packets (discard packets with an inside source address if the packet arrives on an external interface).
  - This countermeasure is often implemented at the router external to the firewall.

Packet-filter firewall: attacks and countermeasures

- **Tiny fragment attacks:**
  - Split header info over several tiny packets: attacker uses the IP fragmentation option to create extremely small fragments and force the TCP header information into a separate packet fragment.
  - Attack designed to circumvent filtering rules that depend on TCP header information.
    - Typically, a packet filter will make a filtering decision on the first fragment of a packet.
    - All subsequent fragments of that packet are filtered out solely on basis that they are part of packet whose first fragment was rejected.
      - Attacker hopes that the filtering firewall examines only the first fragment and that the remaining fragments are passed through.
- **Countermeasure:**
  - Either discard or reassemble before check.
  - A tiny fragment attack can be defeated by enforcing a rule that the first fragment of a packet must contain a predefined minimum amount of the transport header.
  - If the first fragment is rejected, then the filter can remember the packet and discard all subsequent fragments.

From packet filters to stateful packet filters

- Traditional packet filter makes filtering decisions on an individual packet basis, without considering any 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 (record information about TCP connections),
  - Check each packet validly belongs to one.
- Hence are better able to detect bogus packets out of context.
- May even inspect limited application data.

- Let's see what is meant with **higher layer context**.

From packet filters to stateful packet filters

- Most standardised applications that run on top of TCP follow a client/server model.
- For example, for the Simple Mail Transfer Protocol (SMTP), e-mail is transmitted from a client system to a server system.
  - Client system generates new e-mail messages.
  - Server system accepts incoming e-mail messages and places them in the appropriate user mailboxes.

- SMTP operates by setting up a TCP connection between client and server, in which the TCP server port number, which identifies the SMTP server application, is 25.
- TCP port number for the SMTP client is a number between 1024 and 65535 that is generated by the SMTP client.

From packet filters to stateful packet filters

- In general, when an application that uses TCP creates a session with a remote host, it creates a TCP connection in which
  - TCP port number for remote (server) application is a number less than 1024 and
  - TCP port number for local (client) application is a number between 1024 and 65535.
- Numbers < 1024 are “well-known” port numbers and are assigned permanently to particular applications (e.g., 25 for server SMTP).
- Numbers 1024–65535 are generated dynamically and have temporary significance only for lifetime of a TCP connection.

- A simple packet filtering firewall must permit inbound network traffic on all these high-numbered ports for TCP-based traffic to occur.
  - This creates a vulnerability that can be exploited by unauthorised users.

Stateful packet filters

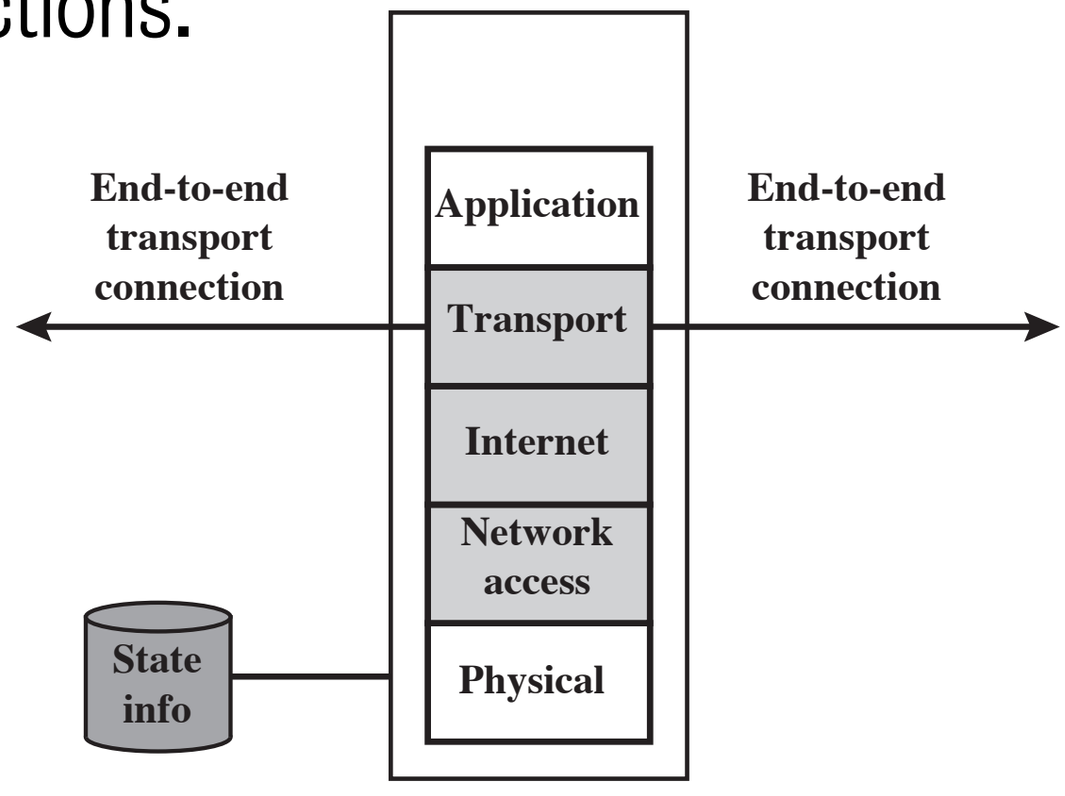
- A **stateful packet inspection firewall** tightens up the rules for TCP traffic by creating a directory of outbound TCP connections.
- There is an entry for each currently established connection.

Source Address	Source Port	Destination Address	Destination Port	Connection State
192.168.1.100	1030	210.22.88.29	80	Established
192.168.1.102	1031	216.32.42.123	80	Established
192.168.1.101	1033	173.66.32.122	25	Established
192.168.1.106	1035	177.231.32.12	79	Established
223.43.21.231	1990	192.168.1.6	80	Established
2122.22.123.32	2112	192.168.1.6	80	Established
210.922.212.18	3321	192.168.1.6	80	Established
24.102.32.23	1025	192.168.1.6	80	Established
223.21.22.12	1046	192.168.1.6	80	Established

- Packet filter will now allow incoming traffic to high-numbered ports only for those packets that fit the profile of one of the entries in this directory.

Stateful packet filters

- A stateful packet inspection firewall reviews the same packet information as a packet filtering firewall, but also records information about TCP connections.
- Some stateful firewalls also keep track of TCP sequence numbers to prevent attacks that depend on the sequence number, such as session hijacking.
- Some even inspect limited amounts of application data for some well-known protocols like FTP, IM and SIPS commands, in order to identify and track related connections.



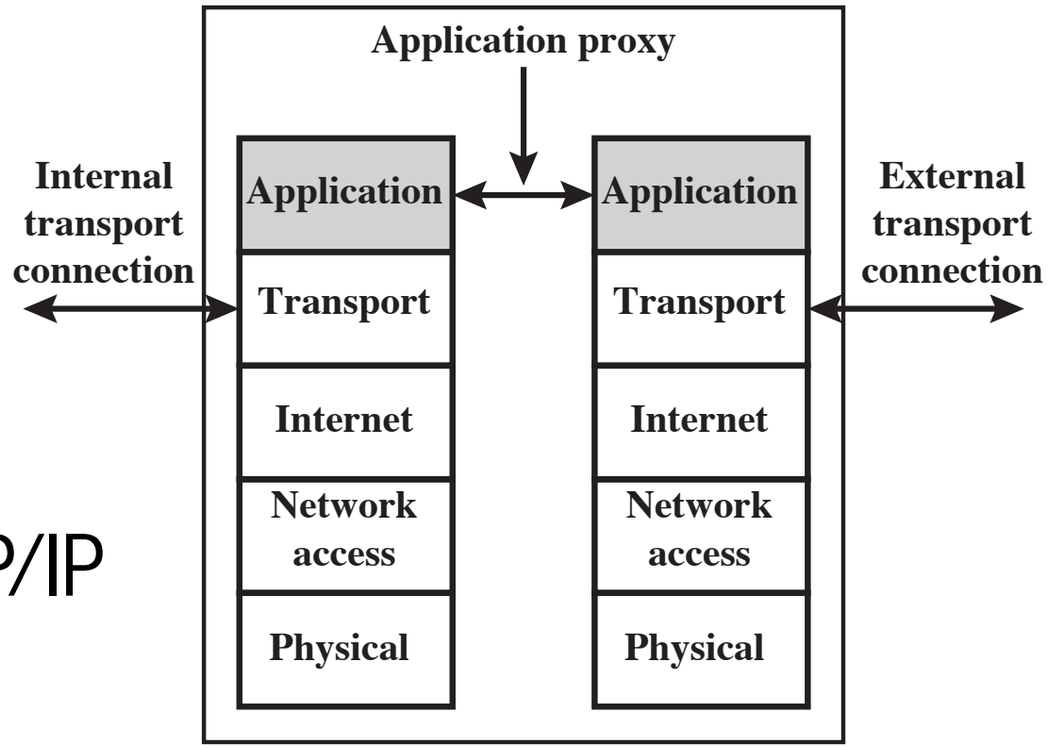
Stateful packet filters

- **IPTables** is a popular tool used to filter network packets in Linux systems. There are two key concepts to understand how IPTables work: tables and chains.
  - **Tables:** general abstraction of the different chains: raw, **filter**, nat, mangle, and security.
  - **Chains:** chains contain a list of defined rules to match against incoming packets ({INPUT,OUTPUT,FORWARD}). The action to perform is called **target** ({ACCEPT,REJECT}).
- Allow incoming SSH connections (22) and existing packets:

```
iptables -P DROP
iptables -A INPUT -p tcp --dport 22 -s $REMOTE_ADDR -j ACCEPT
iptables -A INPUT -m state --state ESTABLISHED,RELATED -j ACCEPT
iptables -A OUTPUT -m state --state ESTABLISHED,RELATED -j ACCEPT
```

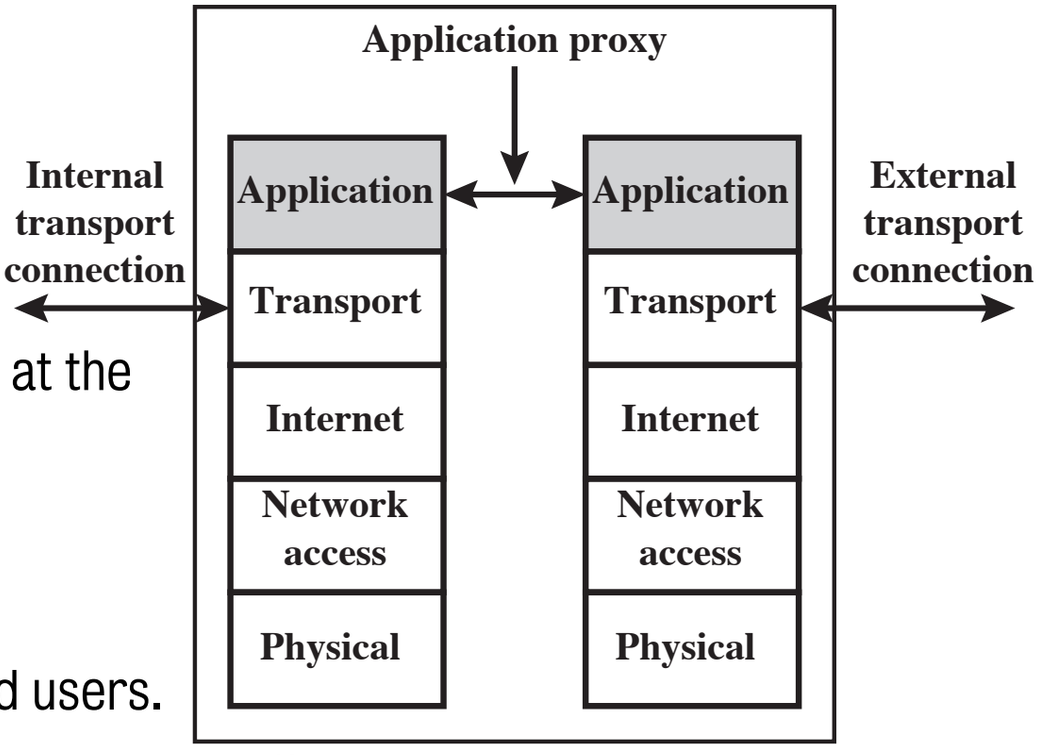
Application-level gateway (or proxy)

- An **application-level gateway**, a.k.a **application proxy**, acts as a relay of application-level traffic.
- Has full access to protocol:
  - User contacts gateway using a TCP/IP application, such as Telnet or FTP.
  - Gateway asks user for name of remote host to be accessed.
  - User responds and provides a valid user ID and authentication information (so that proxy can validate request as legal).
  - Gateway contacts application on remote host and relays TCP segments containing application data between the two endpoints.



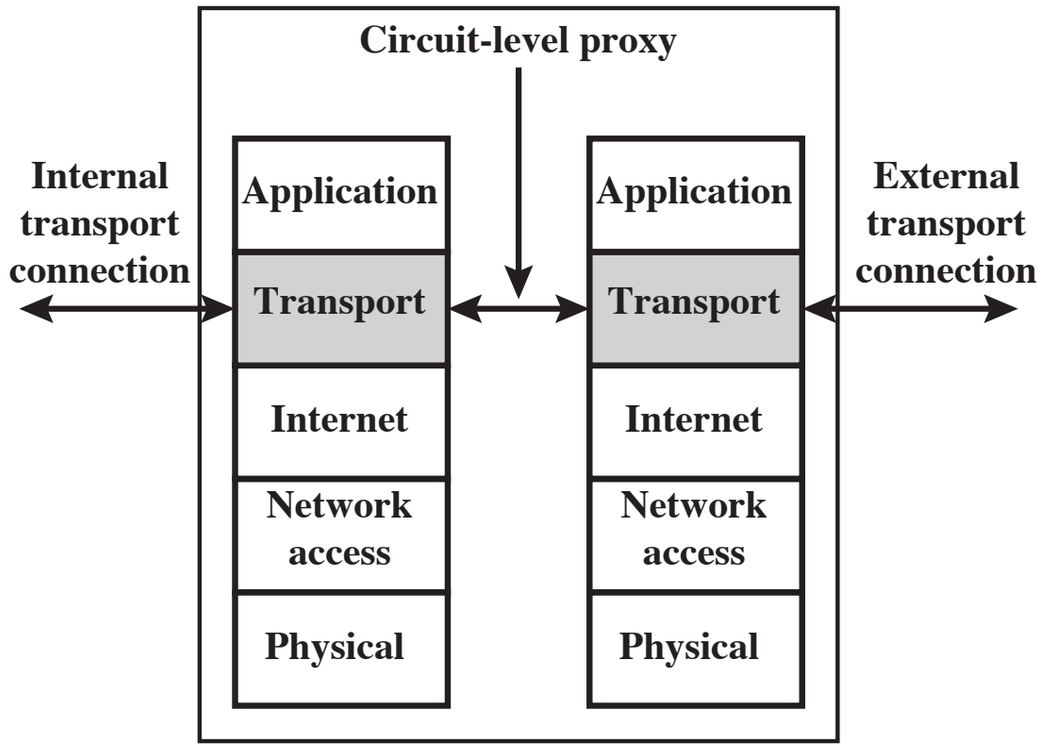
Application-level gateway (or proxy)

- If gateway does not implement the proxy code for a specific application, then the service is not supported and cannot be forwarded across the firewall.
- Gateway can be configured to support only specific features of an application that network administrator considers acceptable while denying all other features.
- Application-level gateways tend to be more secure than packet filters.
  - Rather than trying to deal with the numerous possible combinations that are to be allowed and forbidden at the TCP and IP level, application-level gateway need only scrutinise a few allowable applications.
  - In addition, it is easy to log and audit all incoming traffic at the application level.
- Two main disadvantages:
  - Additional processing overhead on each connection.
    - There are two spliced connections between the end users.
    - Gateway is at splice point and must examine and forward all traffic in both directions.
  - Need separate proxies for each service:
    - some services naturally support proxying,
    - others are more problematic.



Circuit-level gateway

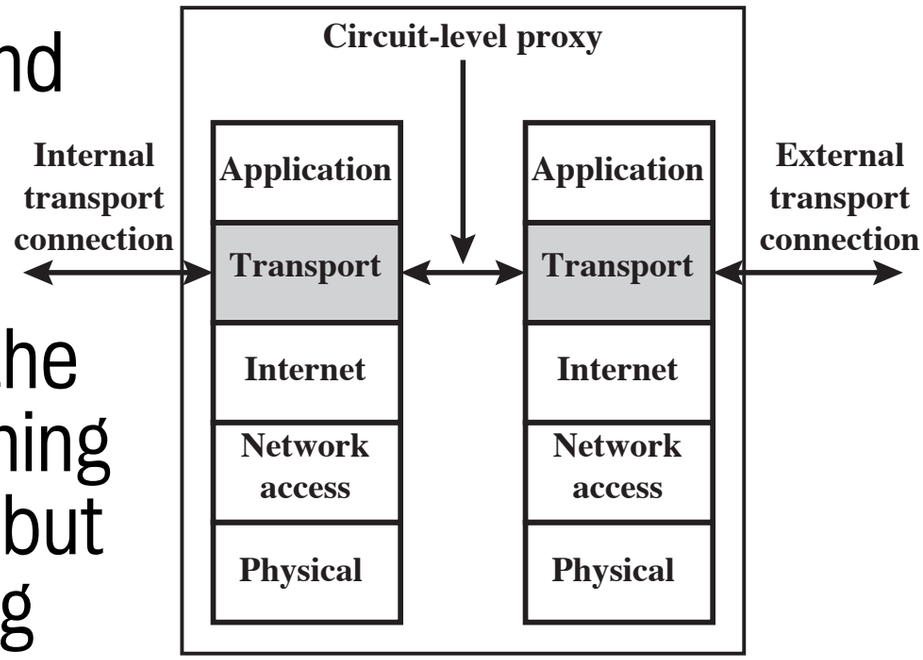
- **Circuit-level gateway** or **circuit-level proxy** can be
  - stand-alone system or
  - specialised function performed by an application-level gateway for certain applications.



  - As with an application gateway, a circuit-level gateway does not permit an end-to-end TCP connection.
  - Circuit-level gateway sets up (and relays) two TCP connections:
    - one between itself and a TCP user on an inner host,
    - one between itself and a TCP user on an outside host.
  - Once the two connections are established, gateway typically relays TCP segments from one connection to the other without examining the contents.
  - Security function consists of determining which connections will be allowed.

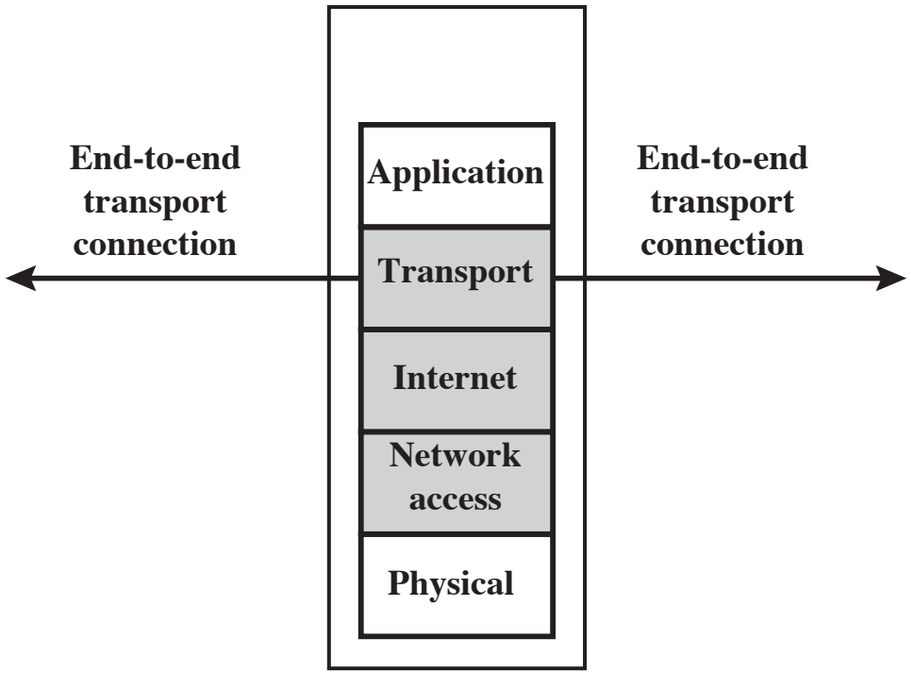
Circuit-level gateway

- Typically used when system admin trusts internal users by allowing general outbound connections.
- Gateway can be configured to support application-level or proxy service on inbound connections and circuit-level functions for outbound connections.
  - In this configuration, gateway can incur the processing overhead of examining incoming application data for forbidden functions but does not incur that overhead on outgoing data.

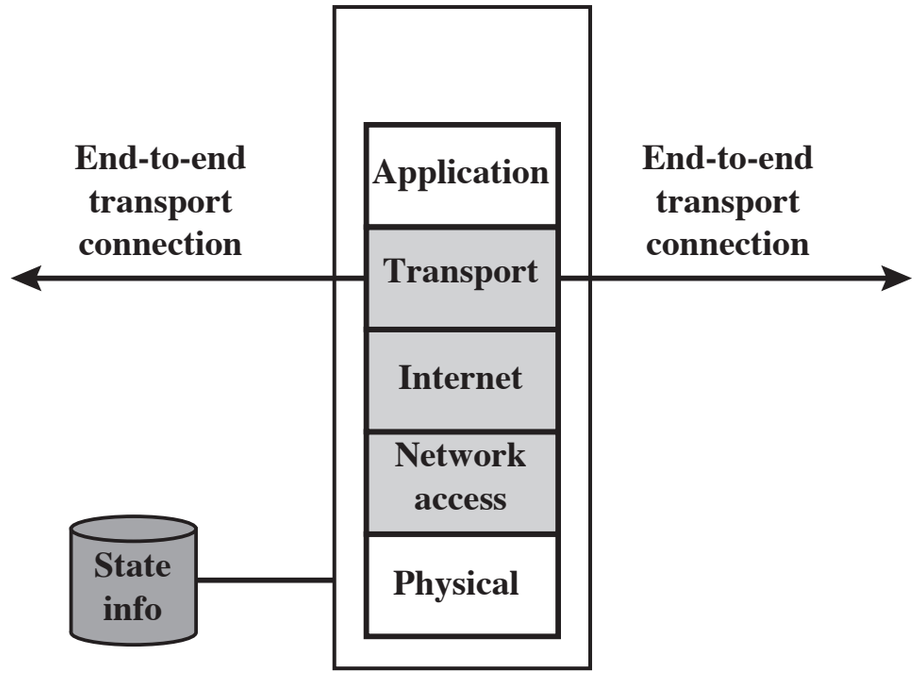


- Example of a circuit-level gateway implementation is **SOCKet Secure (SOCKS):**
  - SOCKS is an Internet protocol that routes network packets between a client and server through a proxy server.

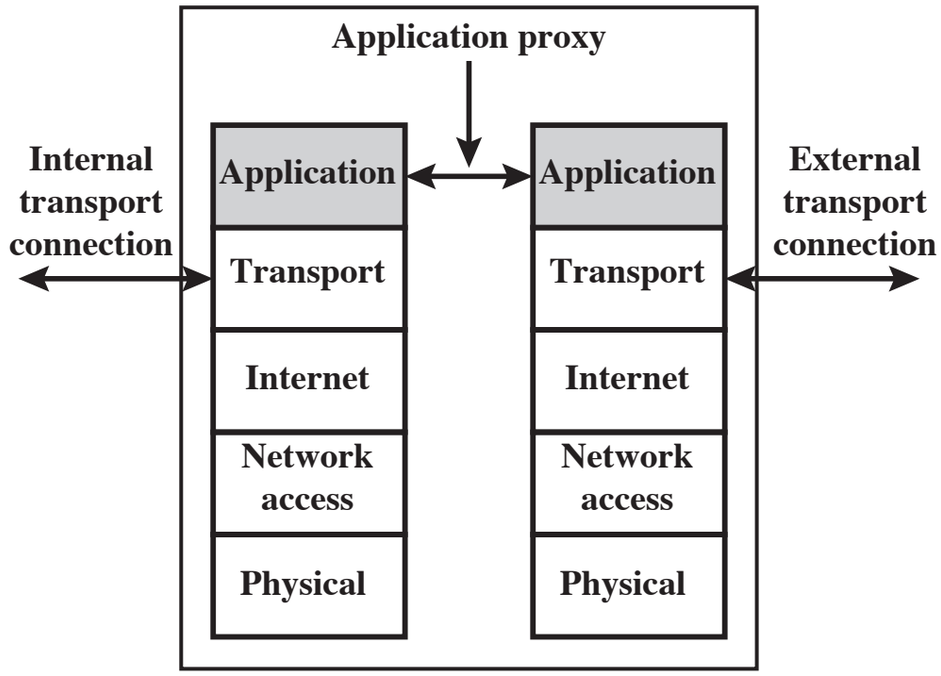
Recap



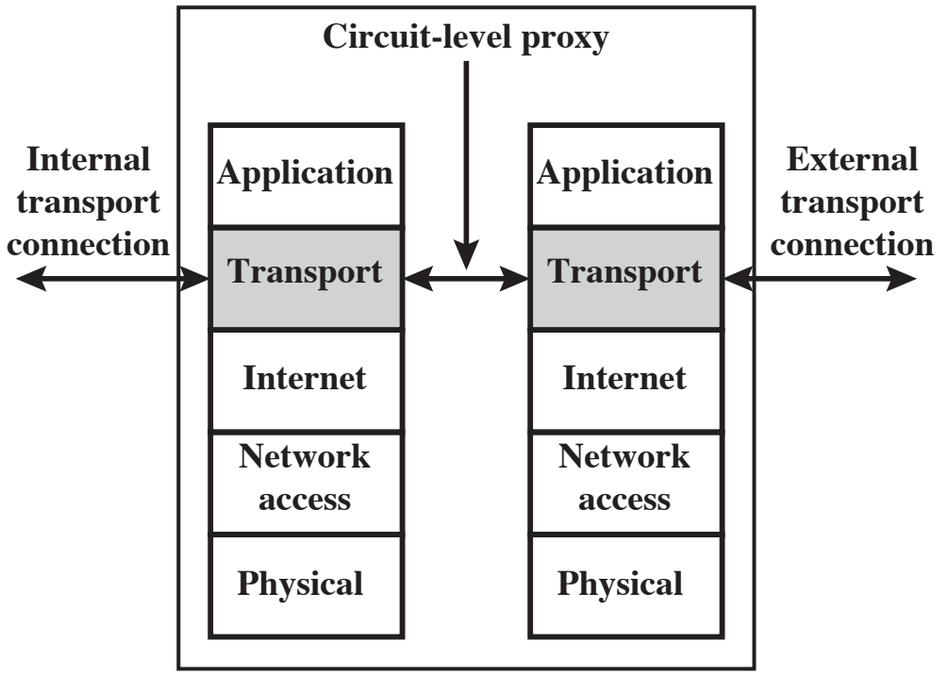
(b) Packet filtering firewall



(c) Stateful inspection firewall



(d) Application proxy firewall



(e) Circuit-level proxy firewall

Firewall basing

- It is common to base a firewall on a stand-alone machine running a common operating system, such as UNIX or Linux.
- Firewall functionality can also be implemented as a software module in a router or LAN switch.
- We now look at some additional firewall basing considerations.

Bastion host

- A **bastion host** is a system identified by the firewall administrator as a critical strong point in the network's security.
  - Typically, bastion host serves as a platform for an application-level gateway (or provides externally accessible services).
  - Potentially exposed to "hostile" elements, hence is secured to withstand this:
    - Bastion host hardware platform executes a secure version of its operating system, making it a hardened system.
    - Only services that network admin considers essential are installed on bastion host.
      These could include proxy applications for DNS, FTP, HTTP, SMTP.

Bastion host

- Bastion host may require additional authentication before a user is allowed access to proxy services.
  Each proxy:
    - May require its own authentication before granting user access.
    - Is configured to support only a subset of the standard application's command set.
    - Is configured to allow access only to specific host systems.
    - Maintains detailed audit information by logging all traffic, each connection, and duration of each connection.
        - Audit log essential for discovering and terminating intruder attacks.
    - Is a very small software package specifically designed for network security and thus it is easier to check for security flaws.
        - For example, a typical UNIX mail application may contain > 20,000 lines of code, while a mail proxy may contain < 1000.
    - Is independent of other proxies on the bastion host.
        - It can be uninstalled without affecting the operation of the other proxy applications.
        - If user population requires support for a new service, network admin can easily install the required proxy on the bastion host.
    - Runs as a nonprivileged user in a private and secured directory on the bastion host.

Host-based firewall

- A **host-based firewall** is a software module used to secure individual host.
  - Available in many operating systems
  - or can be provided as an add-on package.
- Like conventional stand-alone firewalls, host-resident firewalls filter and restrict the flow of packets.
- A common location for such firewalls is a server.

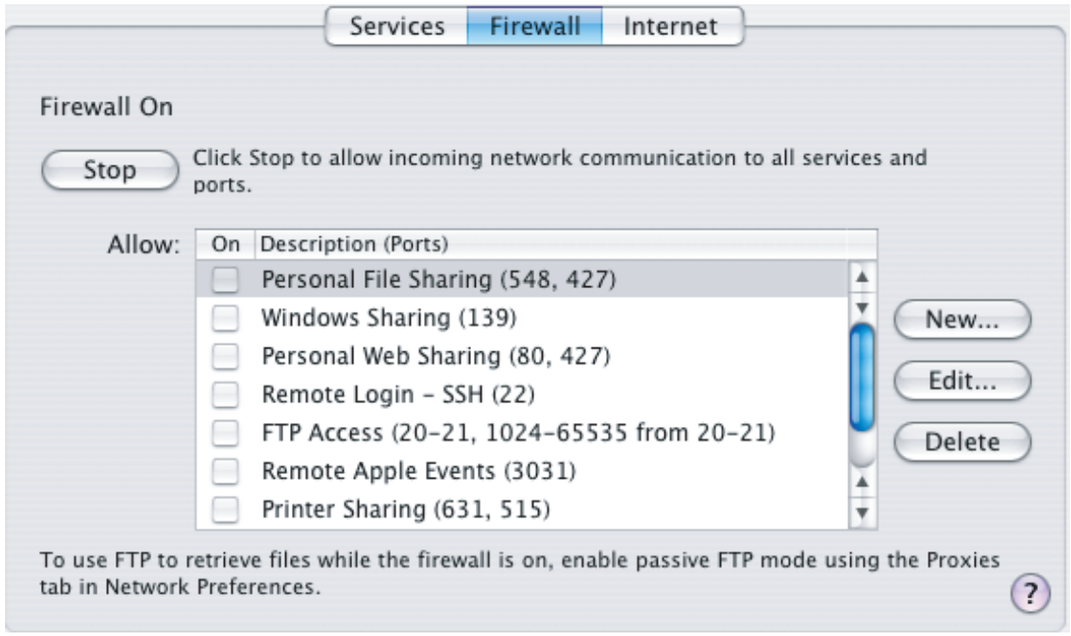
Host-based firewall

- There are several advantages to the use of a server-based or workstation-based firewall:
  - Can tailor filtering rules to host environment.
    - Specific corporate security policies for servers can be implemented, with different filters for servers used for different application.
  - Protection is provided independent of topology.
    - Thus both internal and external attacks must pass through the firewall.
- Used in conjunction with stand-alone firewalls, host-based firewall provides an additional layer of protection.
  - A new type of server can be added to network, with its own firewall, without necessity of altering the network firewall configuration.

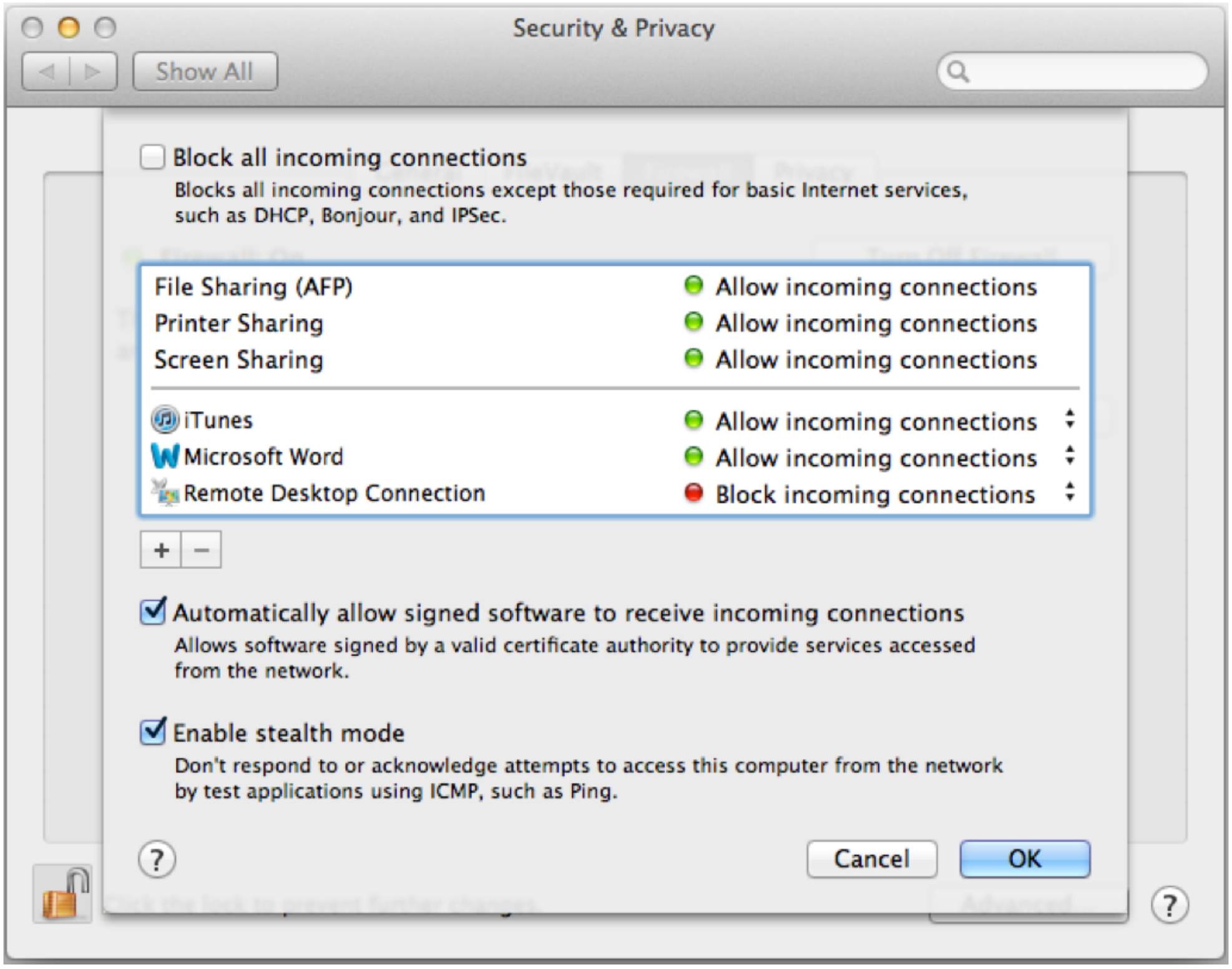
Personal firewall

- A **personal firewall** controls traffic between PC/workstation and Internet or enterprise network.
- A software module on personal computer or in home/office DSL/cable/ISP router.
- Typically much less complex than other firewall types.
- Primary role is to deny unauthorised remote access to the computer and monitor outgoing activity for malware.

Example: when a user enables personal firewall in Mac OS X, all inbound connections are denied except for those the user explicitly permits.



Personal firewall

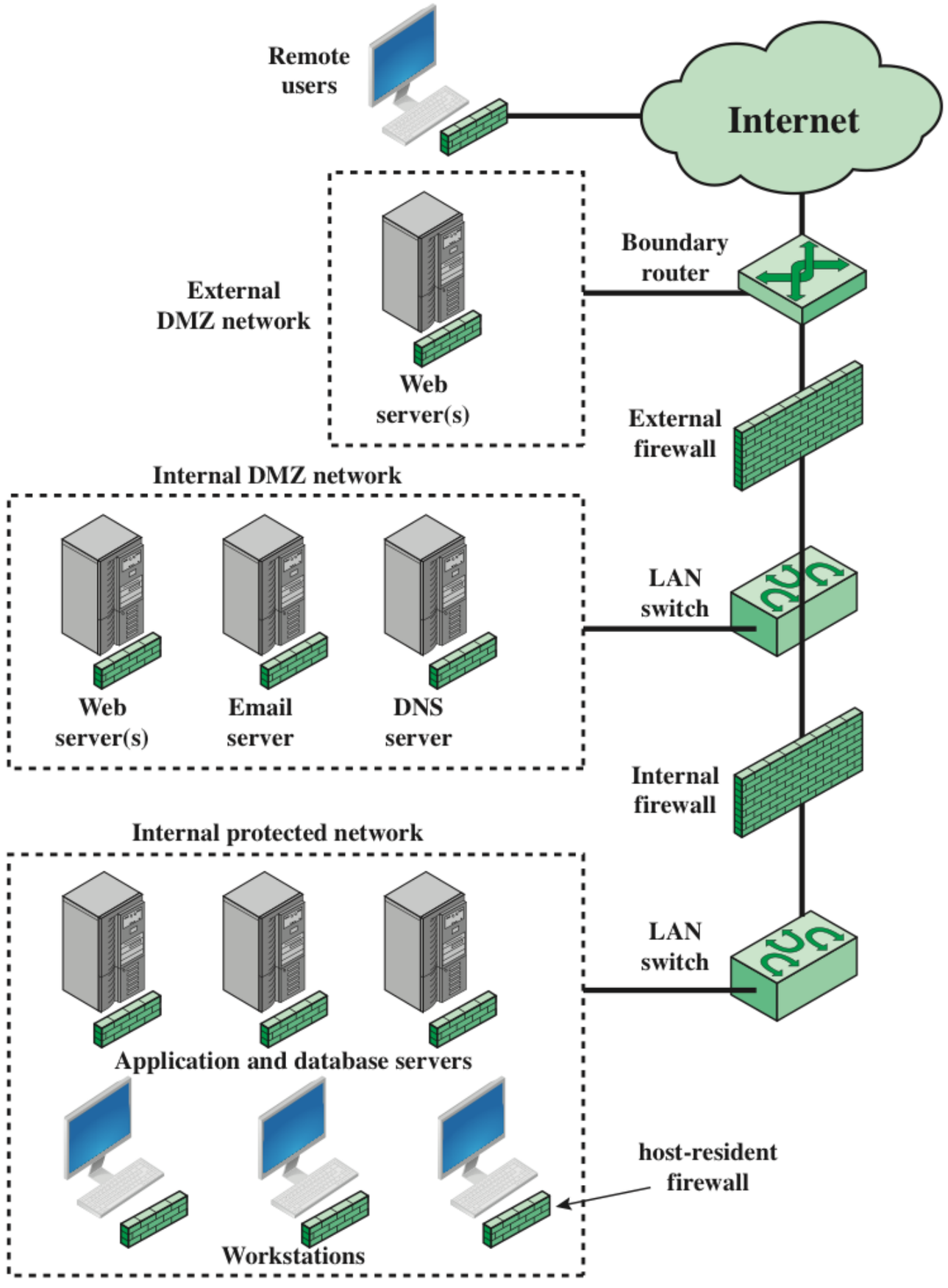


Personal firewall and configuration

- A firewall is positioned to provide a protective barrier between an external, potentially untrusted source of traffic and an internal network.
- A security administrator must decide on the location and on the number of firewalls needed.
- Let us look at some common options:
  - DMZ networks.
  - VPN (Virtual private network).
  - Distributed firewalls.

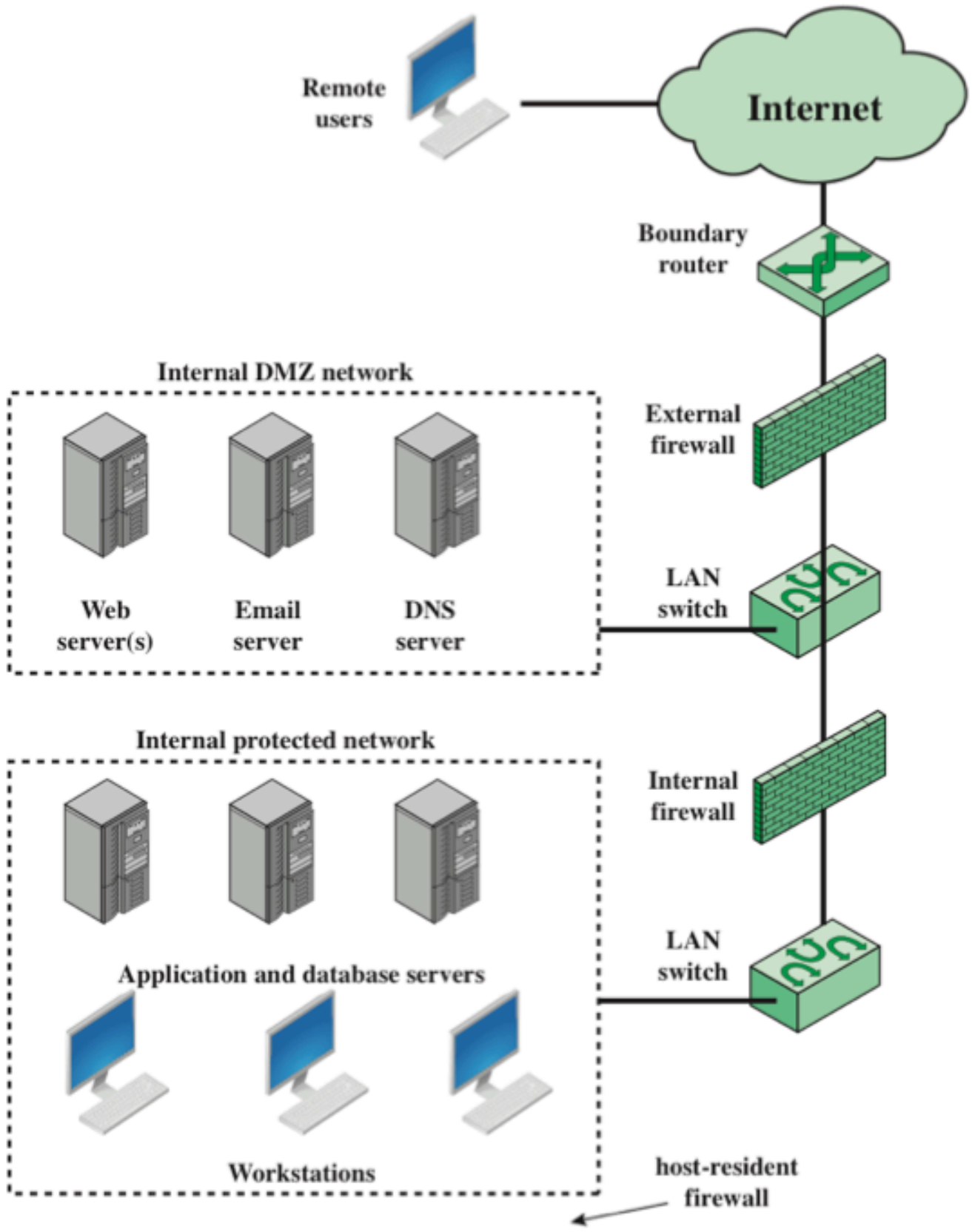
DMZ networks

- Most common distinction: internal or external firewall.
- **External firewall:** placed at edge of a local or enterprise network, just inside the boundary router that connects to Internet or some wide area network (WAN).
- One or more **internal firewalls** protect bulk of enterprise network.



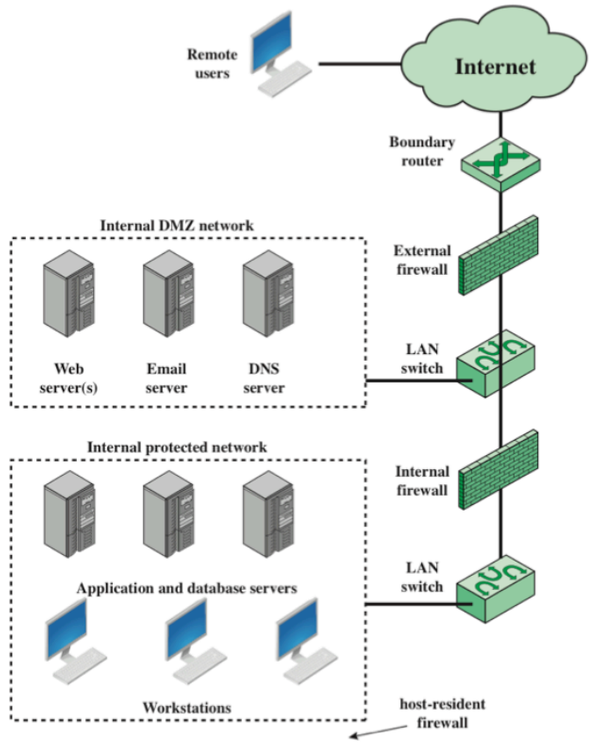
DMZ networks

- Between 2 types of firewalls: networked devices in a region called **DMZ (demilitarized zone) network**
- Systems that are externally accessible but need some protections are usually located on DMZ networks.
- Typically, systems in DMZ require or foster external connectivity, such as a corporate Web site, an e-mail server, or a DNS server.

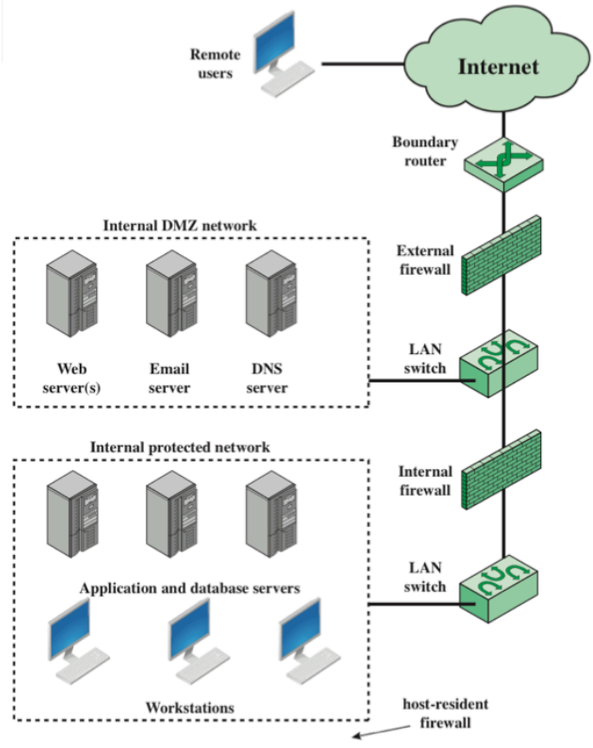


DMZ networks

- External firewall provides
  - A measure of access control and protection for DMZ system consistent with their need for external connectivity.
  - A basic level of protection for remainder of network.
- Internal firewall:
  - Adds more stringent filtering capability to protect enterprise servers and workstations from external attack.
- Provides two-way protection with respect to DMZ:
  - Protects remainder of the network from attacks launched from DMZ systems (such attacks might originate from worms, rootkits, bots, or other malware lodged in a DMZ system).
  - Protects DMZ systems from attacks from internal protected network.



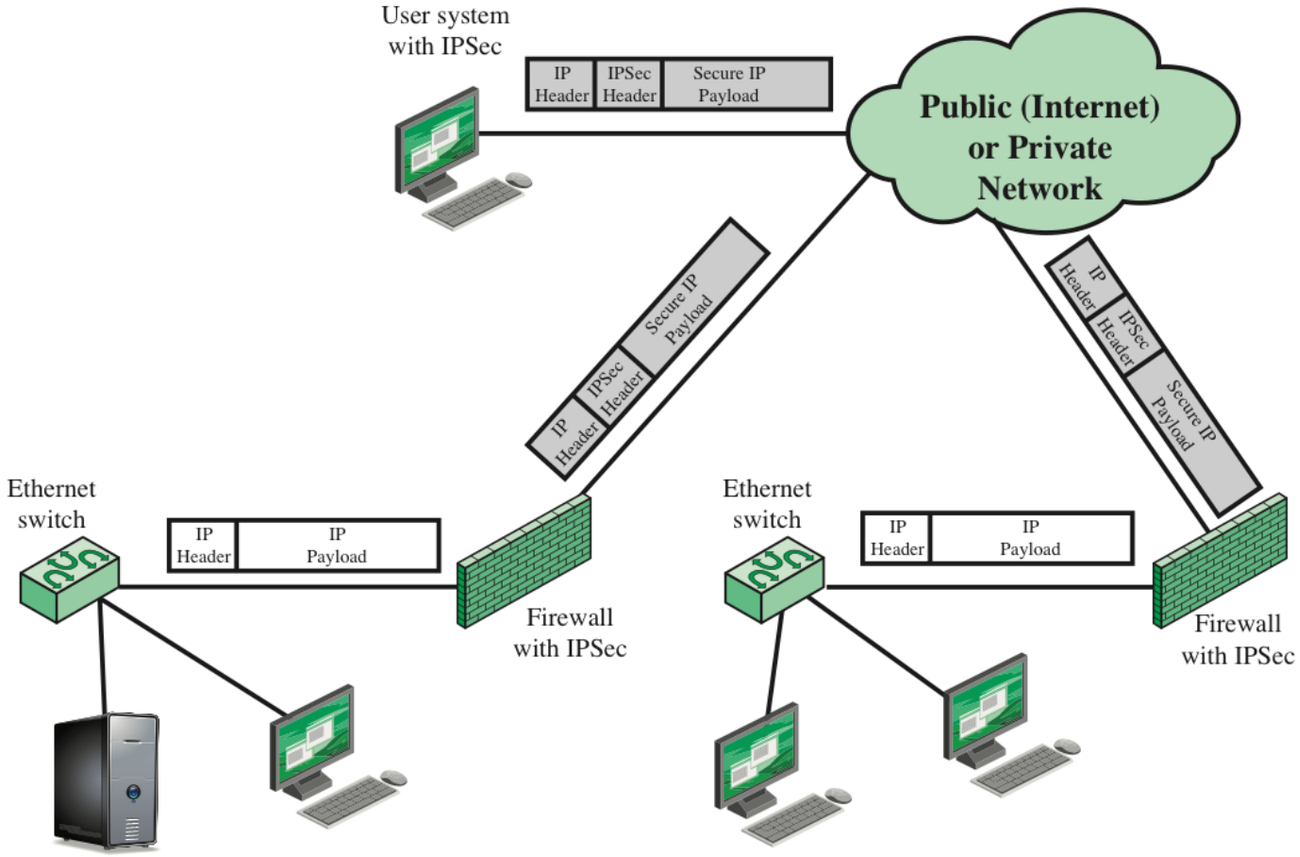
DMZ networks



- Multiple internal firewalls can be used to protect portions of the internal network from each other.
  - For example, firewalls can be configured so that internal servers are protected from internal workstations and vice versa.
  - A common practice is to place the DMZ on a different network interface on the external firewall from that used to access the internal networks.

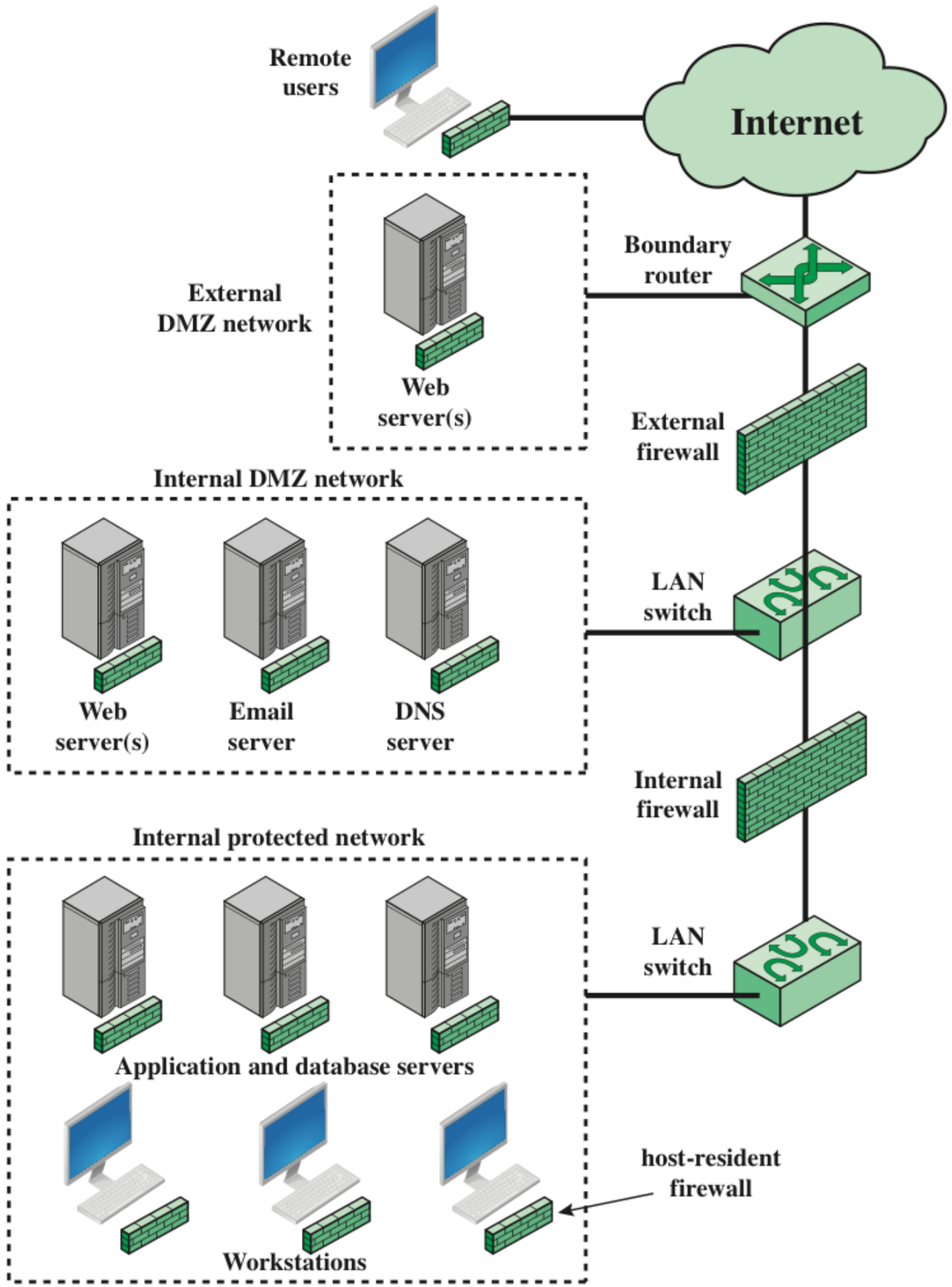
Virtual private networks (VPNs)

- **Virtual private network (VPNs):**
  - Consists of a set of computers that interconnect by means of a relatively insecure network and that make use of encryption and special protocols to provide security.
  - Uses encryption and authentication in the lower protocol layers to provide a secure connection through an otherwise insecure network, typically the Internet.
- Encryption may be performed by firewall software or by routers.
  - Most common protocol mechanism used for this purpose is at the IP level and is known as **IPsec** (cf. lecture on IPsec).



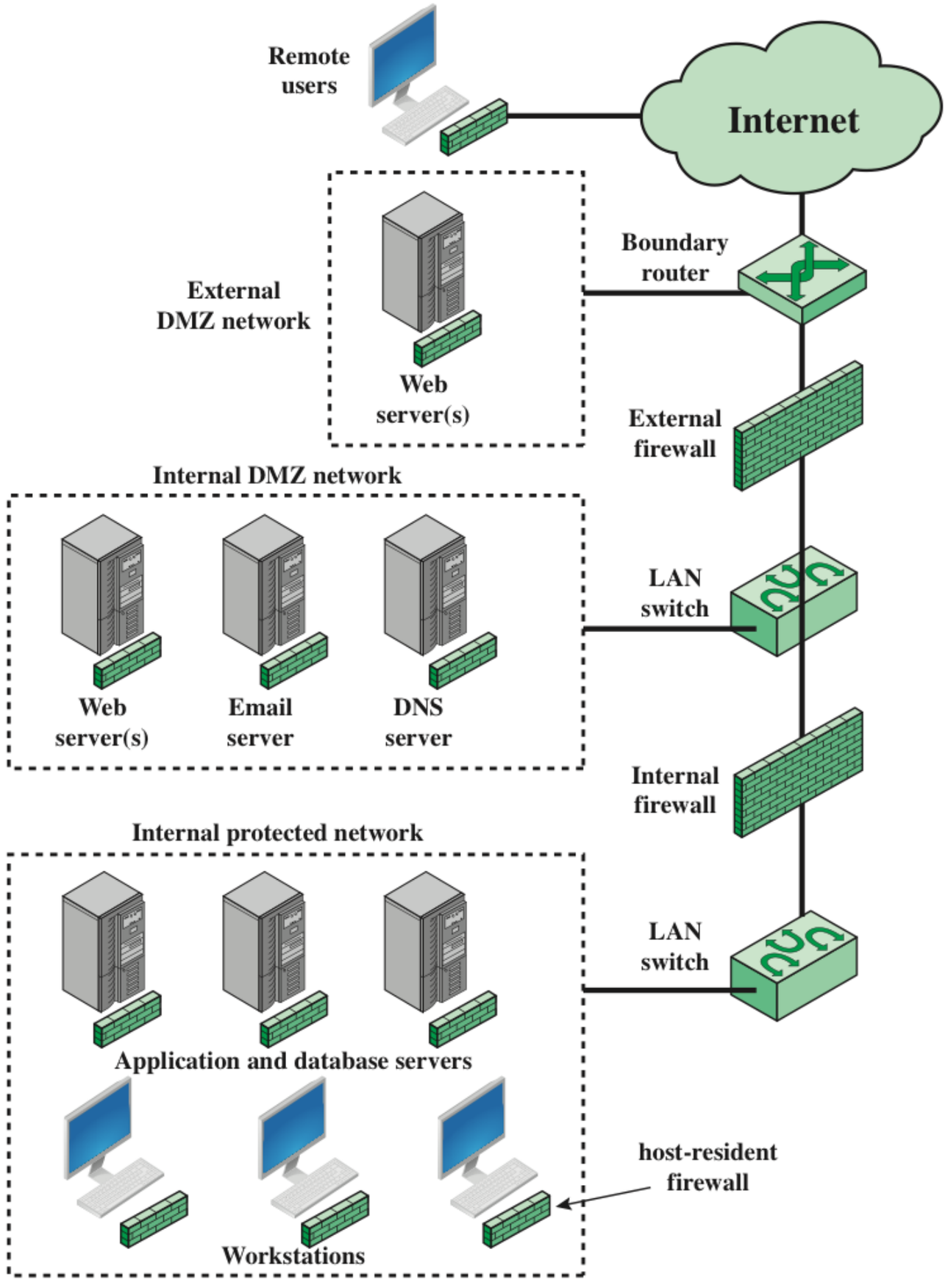
Distributed firewalls

- A **distributed firewall configuration** involves stand-alone firewall devices and host-based firewalls working together under a central administrative control.
- Admins can configure
  - host-resident firewalls on hundreds of servers and workstations,
  - personal firewalls on local and remote user systems.
- These firewalls protect against internal attacks and provide protection tailored to specific machines and applications.



Distributed firewalls

- With distributed firewalls, it makes sense to have both an internal and an external DMZ.
  - Web servers that need less protection (have less critical information on them) are placed in an external DMZ, outside external firewall. Their protection is provided by host-based firewalls.
- Important aspect of this configuration: **security monitoring**.
  - Includes log aggregation and analysis, firewall statistics, fine-grained remote monitoring of individual hosts.



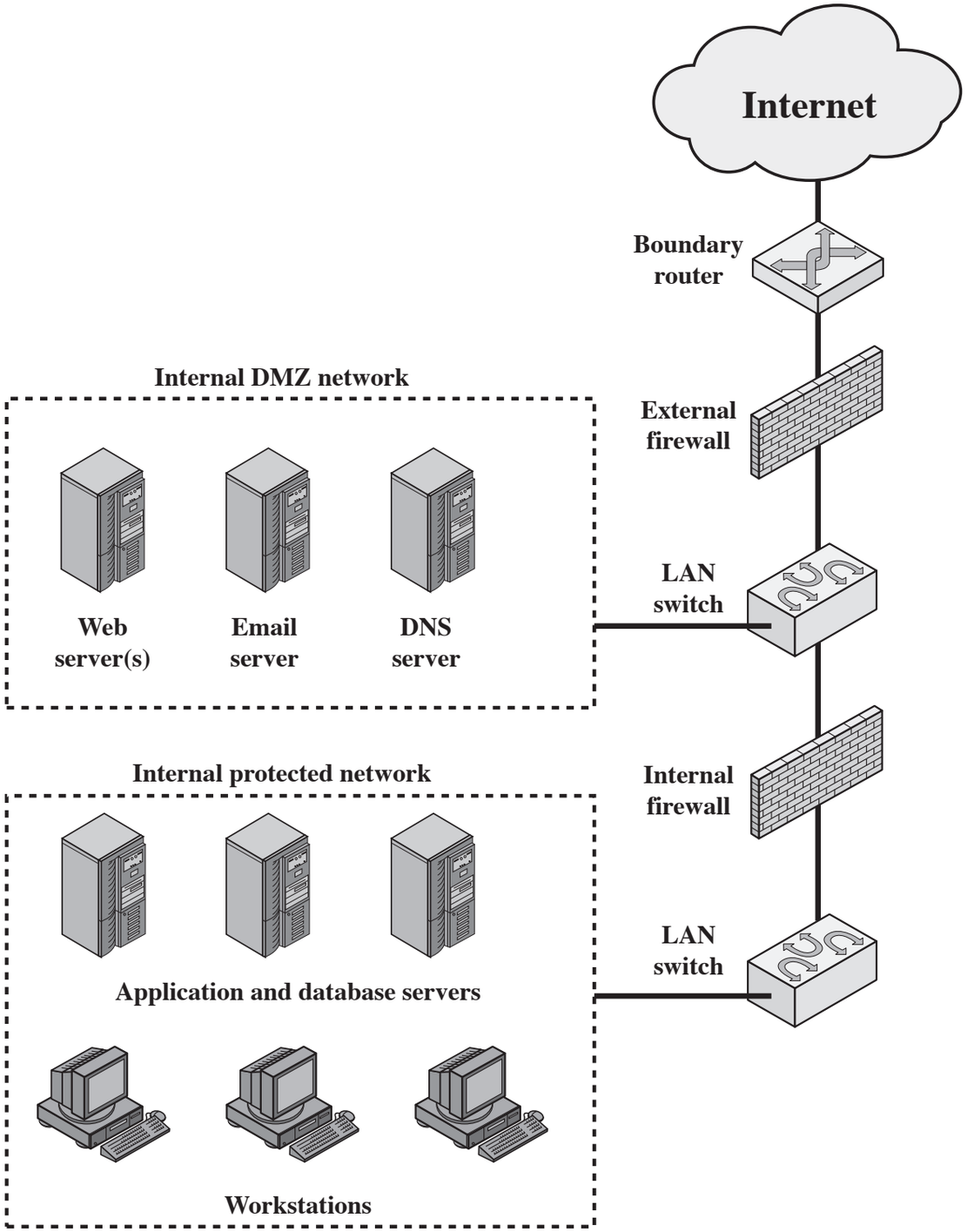
Summary of firewall locations and topologies

- **Host-resident firewall:**
  - Includes personal firewall software and firewall software on servers.
  - Can be 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 applications.
- **Single bastion inline:**
  - A single firewall device between an internal and external router.
  - Firewall may implement stateful filters and/or application proxies.
  - Typical configuration for small to medium-sized organisations.
- **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.
  - Typical configuration for small to medium-sized organisations.

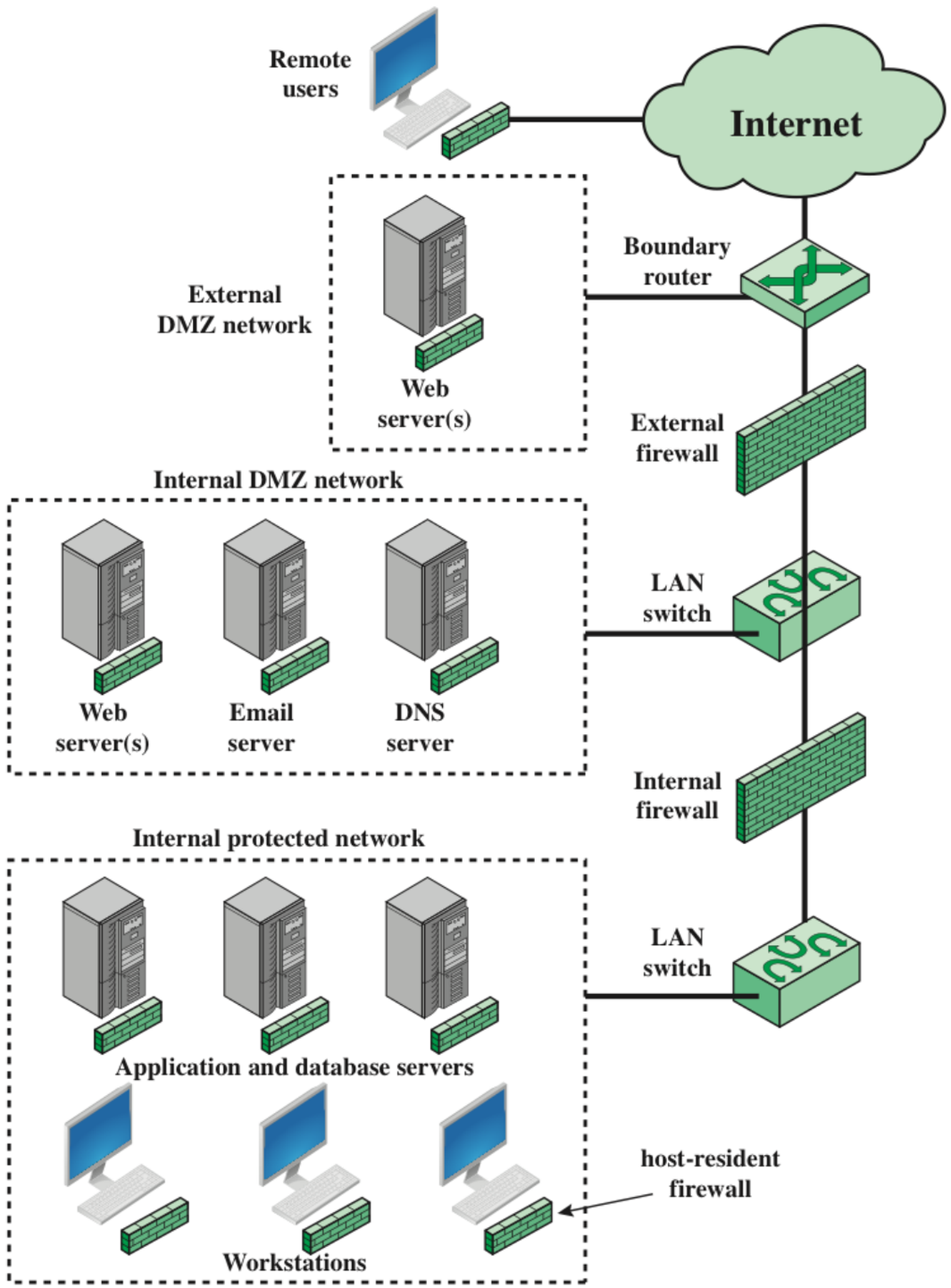
Summary of firewall locations and topologies

- **Double bastion inline:**
  - DMZ is sandwiched between bastion firewalls.
  - Common for large businesses and government organisations.
- **Double bastion T:**
  - DMZ is on a separate network interface on the bastion firewall.
  - Common for large businesses and government organisations and may be required.
    For example, this configuration is required for Australian government use (Australian Government Information Technology Security Manual – ACSI33).
- **Distributed firewall configuration:**
  - Used by some large businesses and government organisations.

Summary of firewall locations and topologies



Double bastion inline



Distributed firewall configuration

RECAP: What have we learnt?

- **What is a firewall?** A firewall forms a barrier through which the traffic going in each direction must pass.

- **What is a policy?** A firewall security policy dictates which traffic is authorized to pass in each direction.

- **How do firewalls operate?** There are different types of firewalls. A firewall may be designed to operate as a filter at the level of IP packets, or may operate at a higher protocol layer.

- **Where do firewalls operate?** Firewalls can be an effective means of protecting a local system or network of systems from network-based security threats.

Don’t (always) blame the firewall

LAB time (EPISODE 4)

- BPG
  - `command: traceroute, whois`

- DNS
  - `command: dig`