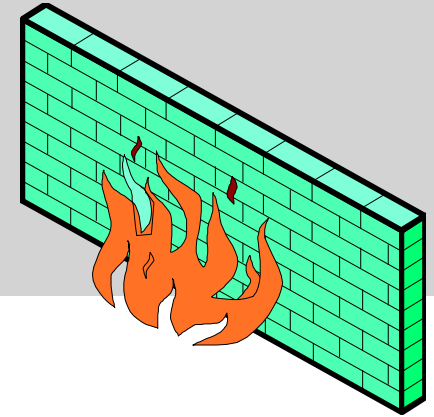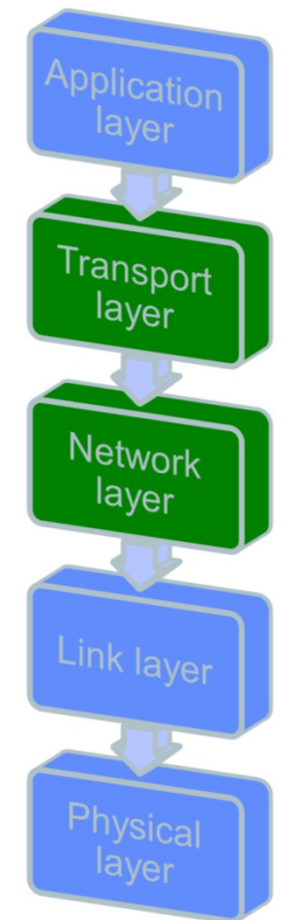# Firewalls

Chapter 21.1

The danger of overload

- The firewall …
  - Can be a dedicated box, a router, a PC or a collection of hosts
  - You get what you pay for – cost from 10,000 SEK to 1,000,000 and even more

- If a firewall is overloaded and cannot handle the traffic,
  it must drop unprocessed packets
  - What would the alternative be?
  - This results in denial of service to its users

- Firewalls must have the capacity to handle all potential traffic peaks
  - Must be able to work at wire speed
  - Logs and alarms must be configured accordingly

- Many solutions are designed to handle normal traffic but not heavy loads/attacks
  - Logging could be one problem
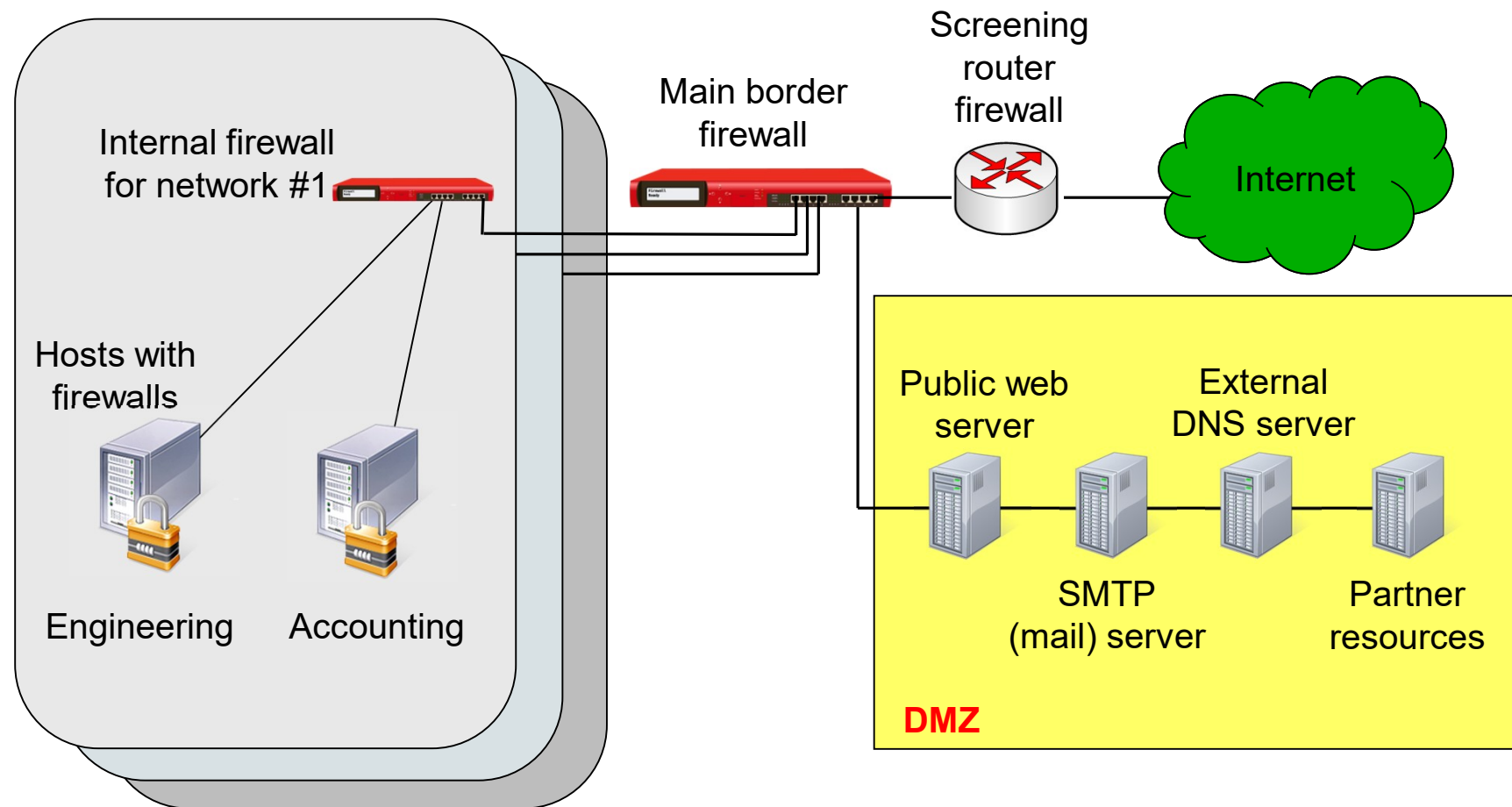 - Suppose it is configured to send an email message for each suspect packet…
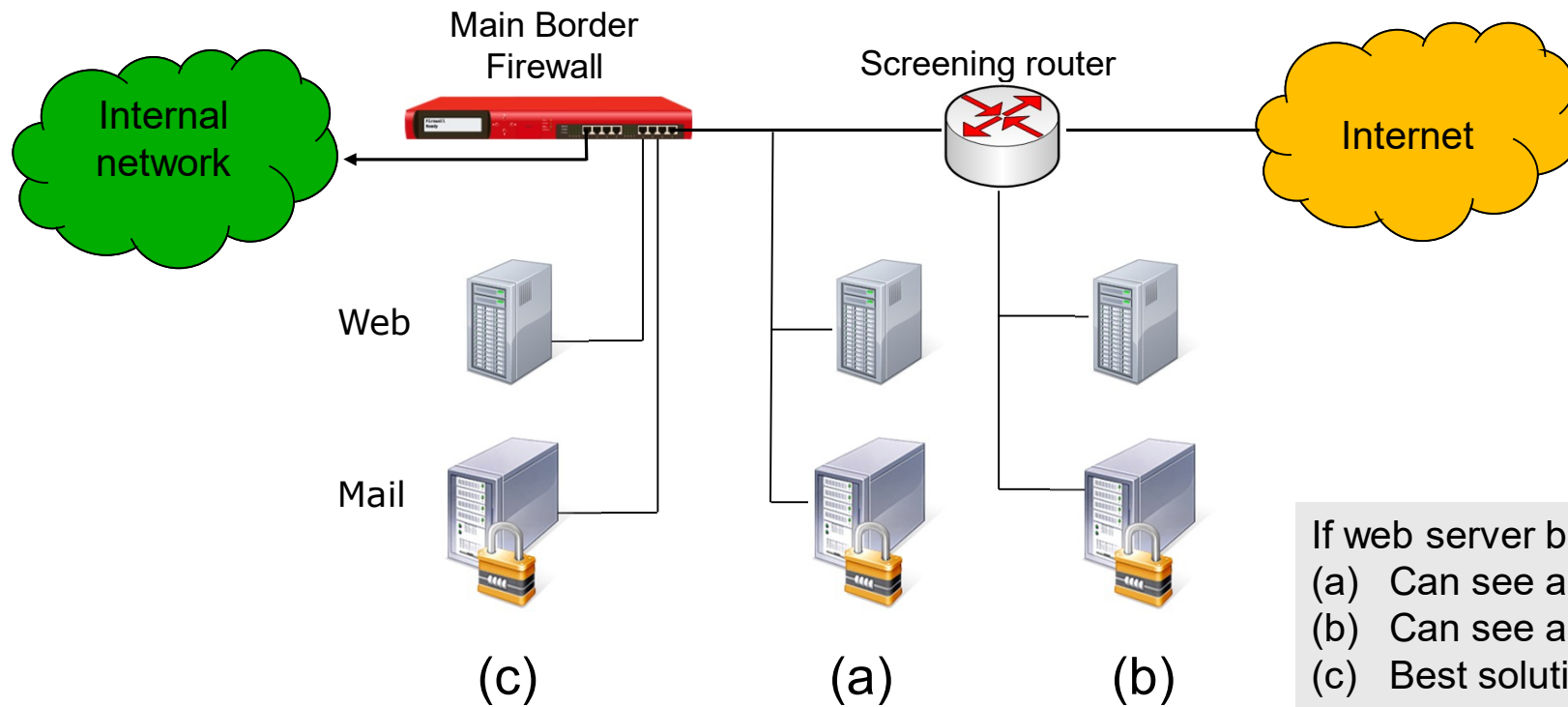
What layer(s) to inspect?

- **Application layer**
  - Number of attacks at the application layer is increasing
  - Application-layer firewalls inspect and understand application layer protocols such as FTP, HTTP, SMTP, etc.
  - Can do virus/worm/Trojan filtering (i.e. malware filtering)

- **Transport and Network layer**
  - Attacks and therefore firewalls started here
  - Most firewalls focus on this layer (TCP, UDP, IP)

- **Data-link layer**
  - Usually no or little protection, although attacks can occur here
  - However, attacker must have access to the local network

- **Physical layer**
  - No protection

Application layer

Transport layer

Network layer

Link layer

Physical layer

In-depth security is important

Protecting the DMZ

Internal network

Main Border Firewall

Web

Mail

(c)

Screening router

Internet

(a)

(b)

If web server breached:
(a) Can see all in- and outgoing traffic
(b) Can see all email traffic
(c) Best solution if firewall has enough interfaces

Multiple layers of DMZ and firewalls

- How can switches increase security?
  More in link-layer security lecture

- More than one DMZ can be built

- Multiple firewalls:
  - May offload each other from work
  - May catch problems if one is mis-configured

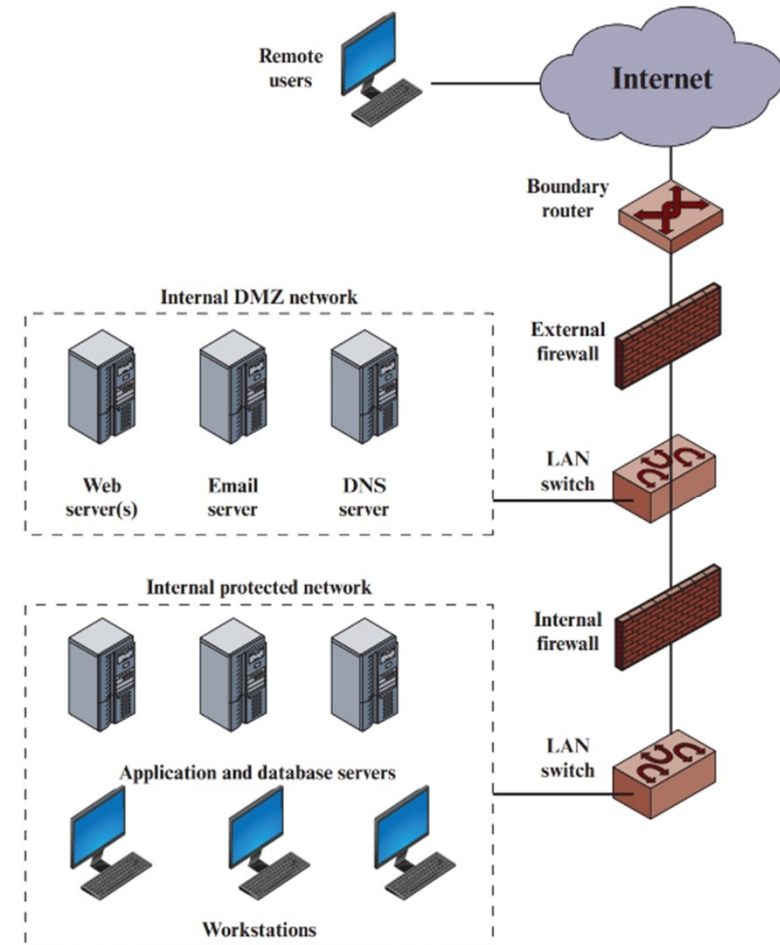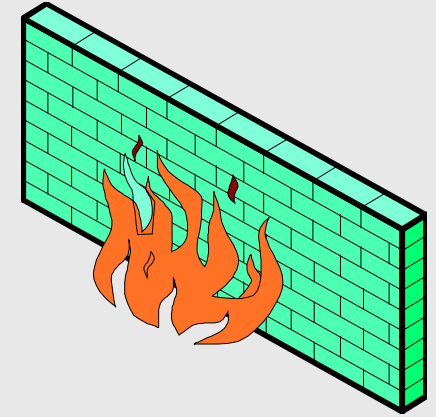- Firewalls have alarm functionality and log suspicious events

Figure 21.3 Example Firewall Configuration

6

Types of firewalls

Chapter 23.3

Firewall history, 1980 →

- Routers preventing own problems to spread (segmentation) [~1980]

- Routers with static packet filters (stateless) removing unwanted traffic [~1990]
  - Table with block/allow consulted for each packet
  - Goal: to build a wall around the own network

- Bastion hosts/proxies
  - Specially hardened hosts (proxies) taking care of a few necessary services
  - May support user authentication

- Stateful (packet) inspection [1994]
  - Technique becoming more and more sophisticated
  - Each packet is investigated, states are kept

- Application-layer inspection ("deep" packet inspection)
  - Over time, inspection has moved upwards in the OSI model
  - One inspection module needed for each protocol to be analyzed

- Firewalls in end devices: host-based/personal firewalls [~2000]
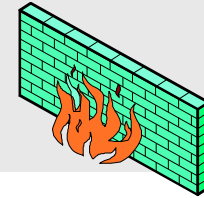
Packet inspection is more powerful than packet filters

Stateful normally means that the full protocol is understood. But terminology may vary.

Firewalls are normally multi-layer: inspects up to the indicated layer

Types of firewalls

Note: There exist many hybrid solutions and some vendors may use different terminology
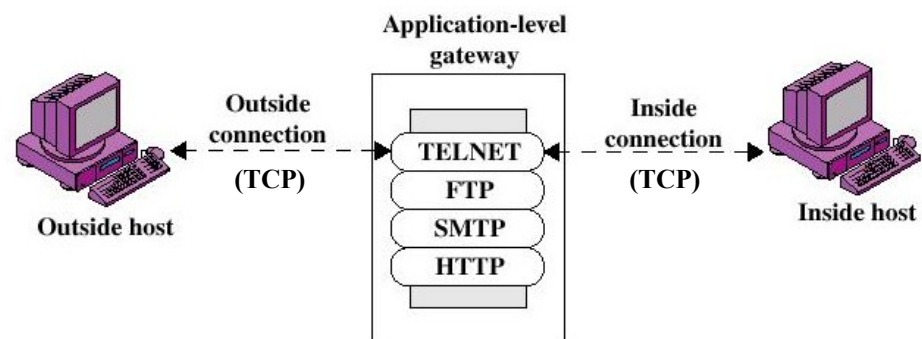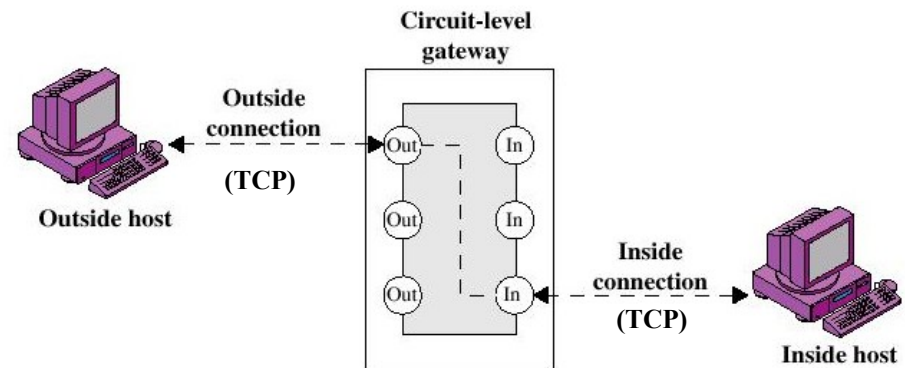
- (Static) Packet **filter** firewalls [OSI layer 3]
    - Stateless. Does not understand TCP
    - Compares IP addresses and port numbers with an access control list
    - IP address spoofing can be detected
    - Fast and cheap, often part of a router
    - Only a complement to a "real" firewall

- Dynamic (self-modifying) packet **filters** [OSI 4]
    - Rules may be changed dynamically by the filter itself to understand SYN – SYN/ACK – ACK sequences
    - Still does not really understand TCP

- Stateful packet **inspection**, SPI [OSI 4+]
    - ACL lists consulted at connection establishment; a state table is used for established connections
    - State aware = knows about TCP states
    - Most SPI firewalls can also inspect some application-layer protocols

- Circuit-level gateways (generic **proxies**) [OSI 4]
    - Relays connections on TCP level: unpacks TCP contents and repacks it for the other side
    - Hides internal structure
    - May perform authentication

- Application level (or layer) gateways (application **proxies**, bastion hosts) [OSI 7]
    - Designed for one application (FTP, HTTP, …)
    - Unpacks application data and repacks it for the other side
    - May perform authentication

- Stateful application-level packet inspection (or **deep packet inspection**) [OSI 7]
    - Inspects several application layer protocols
    - Can have signatures for known problems (IDS), detect Ping-of-death attacks, nonstandard commands in FTP, etc.
    - May require special hardware for performance

- **Air-Gap** Firewalls (special-purpose)

Proxies (or Gateways)

| App. layer GW |
| Circuit layer GW |
| TCP |
| IP |
| Ethernet |
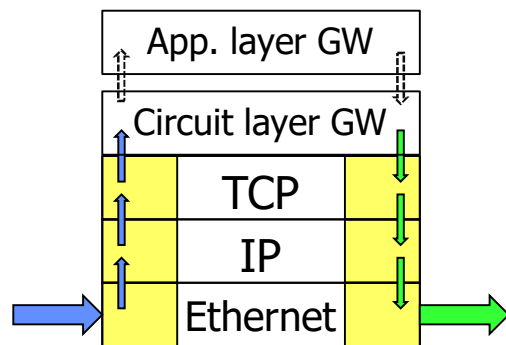
All headers are removed, data is extracted and inspected. If Ok, new headers are created by the proxy.

Circuit-level gateway

Outside host — Outside connection (TCP) → Out / In — Inside connection (TCP) → Inside host

Application-level gateway

Outside host — Outside connection (TCP) → TELNET / FTP / SMTP / HTTP — Inside connection (TCP) → Inside host

"Next Generation" Firewalls (NGFW)

2nd ed. 2016

- Not as new as the name implies, concept introduced around 2010

- Deep packet inspections with support for NAT, VPN, QoS (to limit or prioritize some types of traffic)

- Reputation based malware detection and URL filters/blocks

- Main "new" feature: moving from "port based" to "application aware" firewalls

- Application aware = knows what application is in charge:
  - Understands application behavior (ports, protocols, expected behavior, contents, …)
  - Example: warns if text editor starts to send http or UDP messages
  - Traffic on port 80 or 443 is not just surf
    - Used bygames, file transfers, VPN, … – better control needed
  - May be self-learning and detect changed behavior
  - May be aware of user-IDs and groups and their behavior

- Inspection is done just once for each packet
  - Not separate modules repeating the inspection
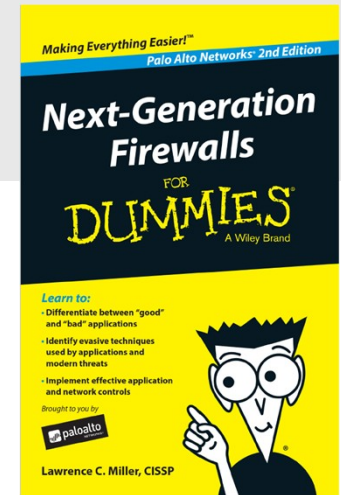
Air-gap "firewalls"

- Air-gap "firewalls" work on the physical layer
  - Isolates computers more or less from the Internet
  - Still not 100% secure

- Offers one-way communication between two hosts
  - No traditional physical network connections work: no TCP, IP, ...
  - Example: relays for mail traffic, input to control system, etc.

Untrusted network

X
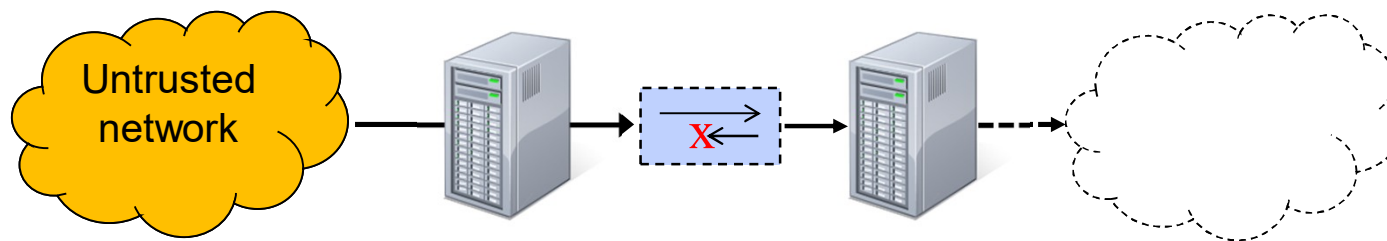
- Optical fiber with only one transmitter (data in one direction only)
- Ethernet cable with receive cable removed

Air-gap firewall ("data diode")

single chan: 46,800 SEK
redundant: 82,500 SEK

Physical data separation is maintained by use of independent data paths connected only by single one-way optical links.

Operates on the principle of a unidirectional stream of data. This stream of data is fed on the low security side through a TCP/IP Socket connection.
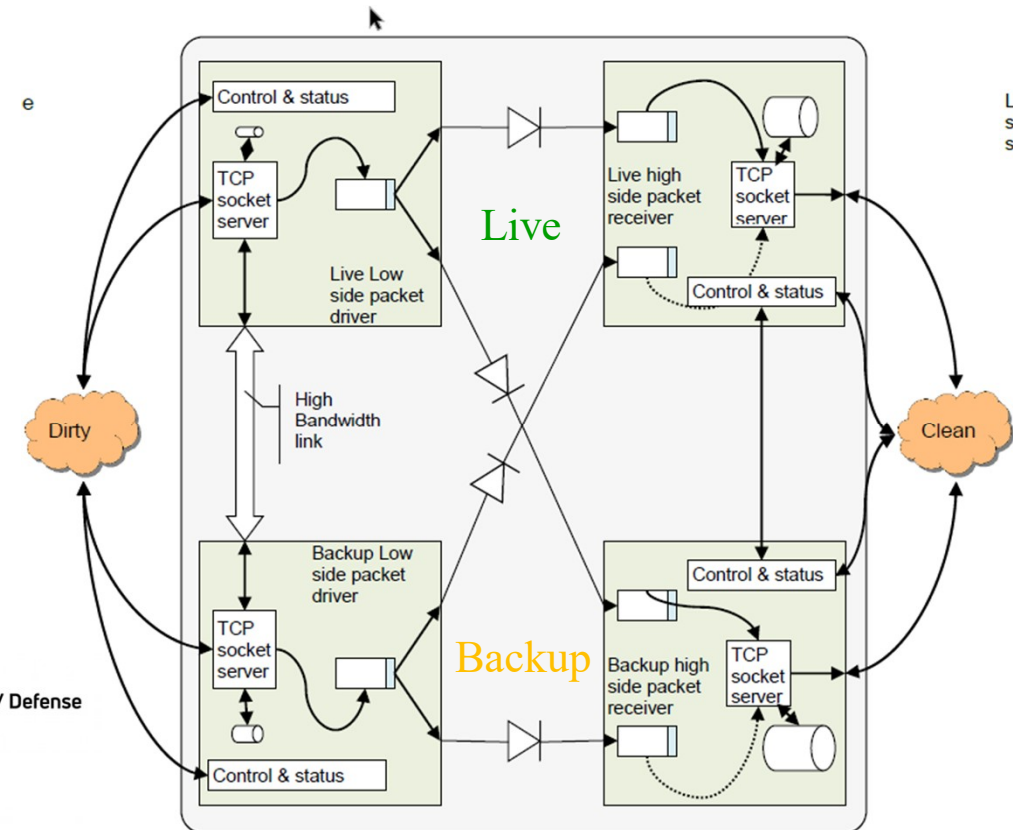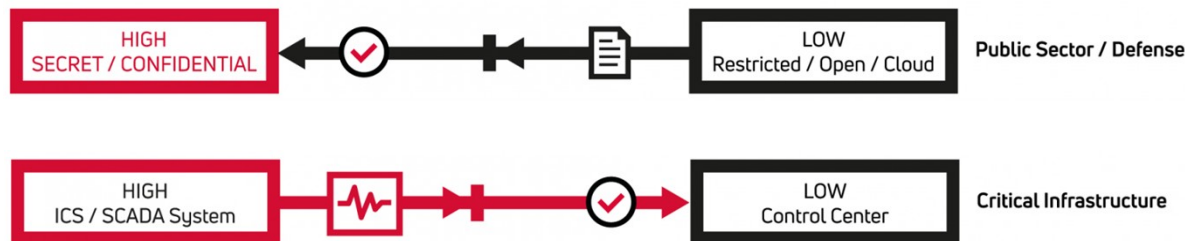
Firewall technology and security
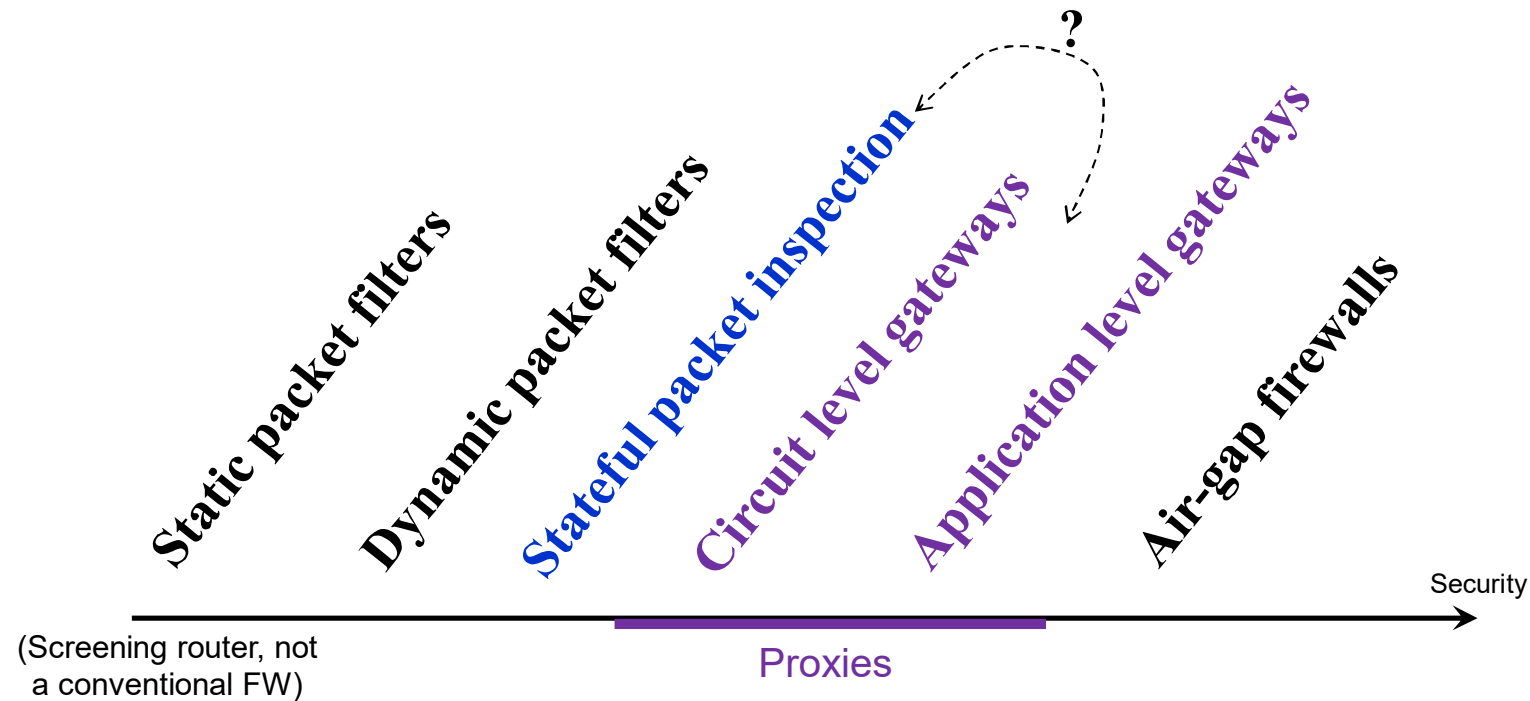
Note: The achieved level of security depends on selecting the right technology for the problem. High-security firewalls are often based on proxy technology.

Industrial strength vs. mainstream products

- Logging and notification ability
  - Good logging ≠ logging everything: too much data or unstructured data does not help understanding
  - Support for SMNP, syslog, etc. and central management
  - A centralized logging system may be designed to receive > 100,000 events per second

- Protocol support
  - Number of supported protocols may vary
  - Good FWs allow us to define rules per interface

- High-volume packet inspection
  - Must be able to handle the traffic in wire speed
  - Application-level protocol inspection requires processing power or hardware (ASICs)

- Device security and redundancy important
  - Underlying system (e.g. Linux) in FW must be secure enough
  - Redundancy (if one device fails, another continues)

- Multiple network interfaces, network throughput, multi-protocol support, advanced rule syntax, QoS functions, VPN performance, user authentication, VLAN support, SNMP, IDS functionality, clustering, redundancy, logging and report functions, ...

More firewall examples (Cisco, Checkpoint)

Firepower 2100 Series

- For Internet edge to data center environments
- Firewall throughput from 2.0 to 8.5 Gbps
- Threat inspection from 2.0 to 8.5 Gbps
- Stateful firewall, Application Visibility and Control, NGIPS, Advanced Malware Protection, URL Filtering

Firepower 9000 Series

- For Service provider, data center
- Firewall throughput up to 225 Gbps
- Threat inspection up to 90 Gbps
- Firewall, Application Visibility and Control, NGIPS, Advanced Malware Protection, URL Filtering, DDoS

Midsize Enterprise

6000 Series

Quantum Security Gateways include the power of Gen V in a single security gateway engineered to meet all your business needs today and in the future.

Branch Office

1500/1600/1800/3000 Series

Quantum Security Gateways provide enterprise-grade security in a small form factor, ideal for branch and small office.

Example

28000 SECURITY GATEWAY
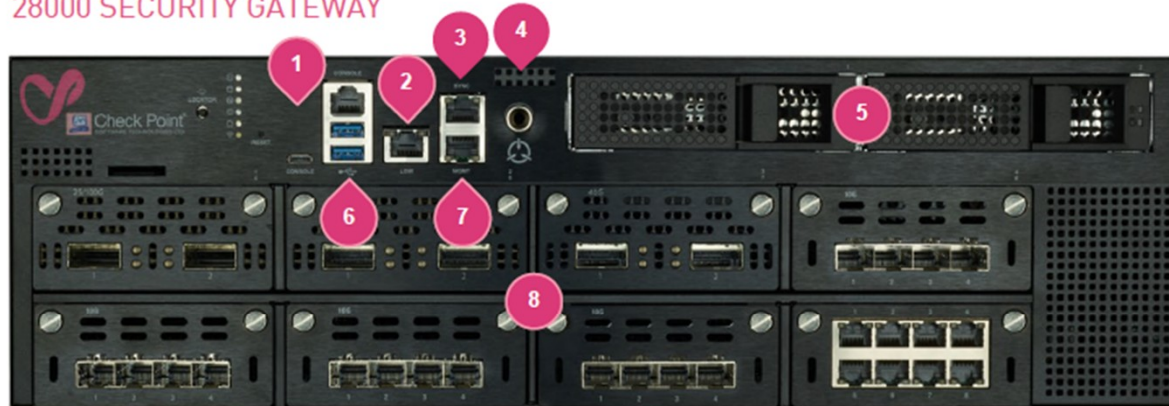
1. RJ45 and USB Type-C console ports
2. Lights-out Management port
3. Sync 10/100/1000 Base-T port
4. ESD ground point
5. 2x 480GB SSD RAID1
6. 2x USB 3.0 ports
7. Management 10/100/1000 Base-T port
8. Eight network card expansion slots
   3x redundant power supplies (back view not shown)
   4x field replaceable fans (back view not shown)

DATA CENTER GRADE PLATFORM

	1 GbE copper	1 GbE fiber	10 GbE	40 GbE	100/25 GbE	Memory	Redundant Power	Redundant Storage
Base model	2	0	4	0	0	64 GB	●	○
Plus model	10	0	12	2	0	96 GB	●	●
Max capacity	66	32	32	16	16	128 GB	●	●

○ optional accessory

List price: from $250,000

17
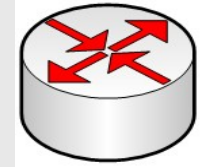
Screening Routers

Normally Static Packet Filters (stateless)

Screening Routers

Block **in** from IP=1.2.3.0/8
Block **out** not from IP=1.2.3.0/8
Block **in** ICMP ECHO
…
Permit All (last rule)

Network 1.2.3.0/8

Internet

Filters traffic based on:

• Source and destination IP address

• Source and destination port

• Interface packet arrives on
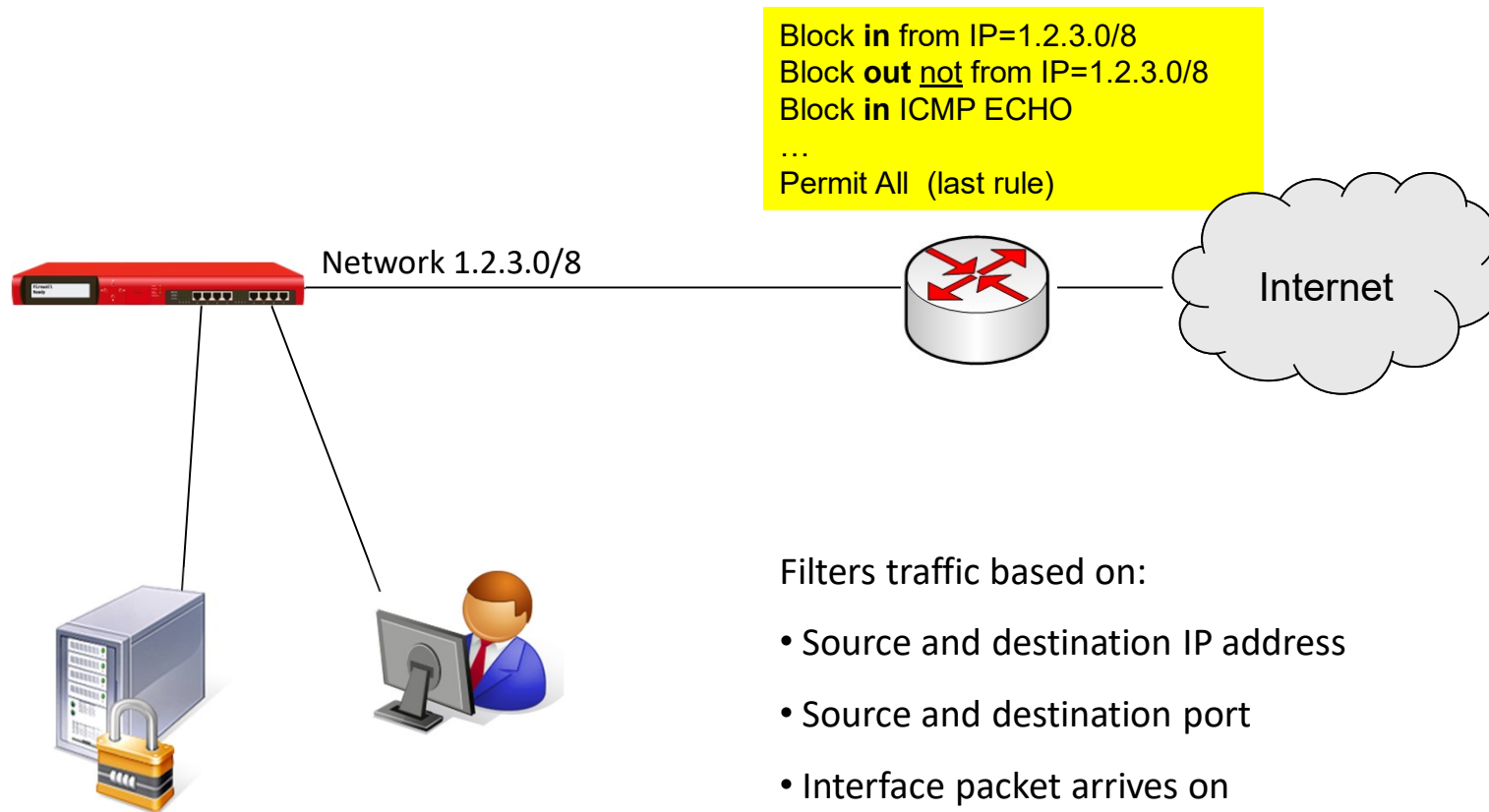
Screening Routers

- Screening Router – our first defense firewall
  - Normally done by adding filter rules in the border router
  - Reduces the load on the main border firewall
  - Good location for Ingress and Egress Filtering
  - Filters out many high-frequency, low-complexity attacks (i.e. garbage)
  - Normally filters on source and destination address, TCP and UDP ports and ICMP message type

- Also reduces risk if main firewall erroneously configured

- Uses Static Packet Filtering (stateless)
  - Fast processing at a competitive price
  - Packets examined one at a time in isolation – this misses many attacks
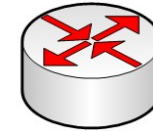
Ingress and Egress filters [RFC 3704]

[Private RFC1918 addresses – should never be forwarded by border router]

    source_addr = 10.*.*.*, DROP
    source_addr = 172.16.*.* to 172.31.*.*, DROP
    source_addr = 192.168.*.*, DROP
    source_addr = 169.254.*.*, DROP   // Auto-configuration addr.

[Reserved IP addresses]

    source_addr = 0.*.*.*, DROP
    source_addr = 127.*.*.*, DROP   // localhost – loopback interface
    source_addr = 240.*.*.* to 255.255.255.255, DROP   // reserved (today)

[Broadcast and TCP multicast]

    source_addr = 111.222.255.255, DROP   // Our broadcast address (or addresses)
    source_addr = 224.*.*.* to 239.*.*.*, DROP   // Multicast address as source
    dest_addr = 224.*.*.* to 239.*.*.* proto TCP, DROP   // TCP is not multicast

[Malformed packets]

    TCP SYN=1 AND FIN=1, DROP   // packet makes no sense (example)
    tiny_fragment, DROP   // fragment overwriting TCP or application layer headers (e.g. HTTP)
    …

Ingress Filters (incoming traffic)

[Internal addresses in incoming traffic]

     source_addr = 111.222.*.*, DROP   // internal IP address range

[ICMP messages – guidelines only]

     ICMP Type = 0, PASS   // incoming "echo reply"
     ICMP Type = 3, PASS   // destination unreachable: net/host/port, Fragmentation needed
     ICMP Type = 4, DROP   // "source quench" messages   [depreciated RFC 6633 but may be transmitted]
     ICMP, DROP   // drop all other incoming ICMP packets??

[Other traffic]

     PASS ALL   // leave rest for main border firewall

Egress filters (outgoing traffic)

[Check that source addresses belong to us – anti-spoofing]

    source_addr NOT 111.222.*.*, DROP // Not our own address as source

[ICMP – guidelines only]

    ICMP Type = 8, PASS   // Allow outgoing echo messages (ping)
    ICMP Type = 3, DROP   // Host/net/port unreachable, … (prevent remote scanning)
    ICMP Type = 11, DROP   // TTL exceeded (prevent remote scanning)
    ICMP, DROP   // Most not needed but more thought needed → → →

[Black-listed ports – may be needed at times]

    TCP source_port = …., DROP   // e.g. Trojan "phone home" port…

[Other traffic]

    PASS ALL   // What is not explicitly forbidden will be accepted !

```
ICMP4_ECHO_REPLY = 0,
ICMP4_DST_UNREACH = 3,
ICMP4_SOURCE_QUENCH = 4,
ICMP4_REDIRECT = 5,
ICMP4_ECHO_REQUEST = 8,
ICMP4_ROUTER_ADVERT = 9,
ICMP4_ROUTER_SOLICIT = 10,
ICMP4_TIME_EXCEEDED = 11,
ICMP4_PARAM_PROB = 12,
ICMP4_TIMESTAMP_REQUEST = 13,
ICMP4_TIMESTAMP_REPLY = 14,
ICMP4_MASK_REQUEST = 17,
ICMP4_MASK_REPLY = 18
```

Other services to consider

- **Windows services** – most should never traverse a firewall:
  RPC   TCP + UDP 135
  NetBIOS   TCP + UDP 137-139
  SMB file sharing   TCP 445

- Simple Network Management Protocol – **SNMP**   UDP 161-162

- Allow **SMTP** only to our corporate mail server   TCP 25
  - To prevent systems from sending spam
  - Drawback: users cannot connect to other mail servers (good?)

SGS student home network, possible to choose Open (no protection), Normal or Protected (no incoming connections allowed).

Normal profile, Inbound filter:
  TCP: 21 (FTP), 23 (Telnet), 25, 53 (DNS), 80 (HTTP), 111 (NFS/RPC), 137-139, 443 (HTTPS)
  UDP: 53, 111, 137-139, 161-162 (SNMP), 1900 (UpNp), 5351 (NAT)

Normal profile, Outbound filter:
  TCP 137-139

SGS student bostäder

Spärrade portar på Chalmers nätverk: (old list)

Beskrivning	Port: tcp	/ udp
Windowstjänster	135	135
	137	137
	138	138
	139	139
	445	445
sunrpc	111	111
mssql	1433	1433
	1434	1434
tftp		69
snmp		161
		162
syslog		514
lpd	515	
nfs	2049	2049
upnp	5000	5000
x-windows	7100	7100
	6000-6255	
Symantec PcAnywhere	5631	
	5632	

Main Border Firewalls

Normally achieved by Stateful Packet Inspection (SPI) firewalls

The Main Border Firewall

Internal firewall
for network 1

Main border
firewall

Screening
router
firewall

Internet

Hosts with
firewalls

Engineering   Accounting

The Main Border Firewall
sits between the Internet
and the internal network
(after the border router)

Predominantly uses
stateful packet inspection
filtering techniques
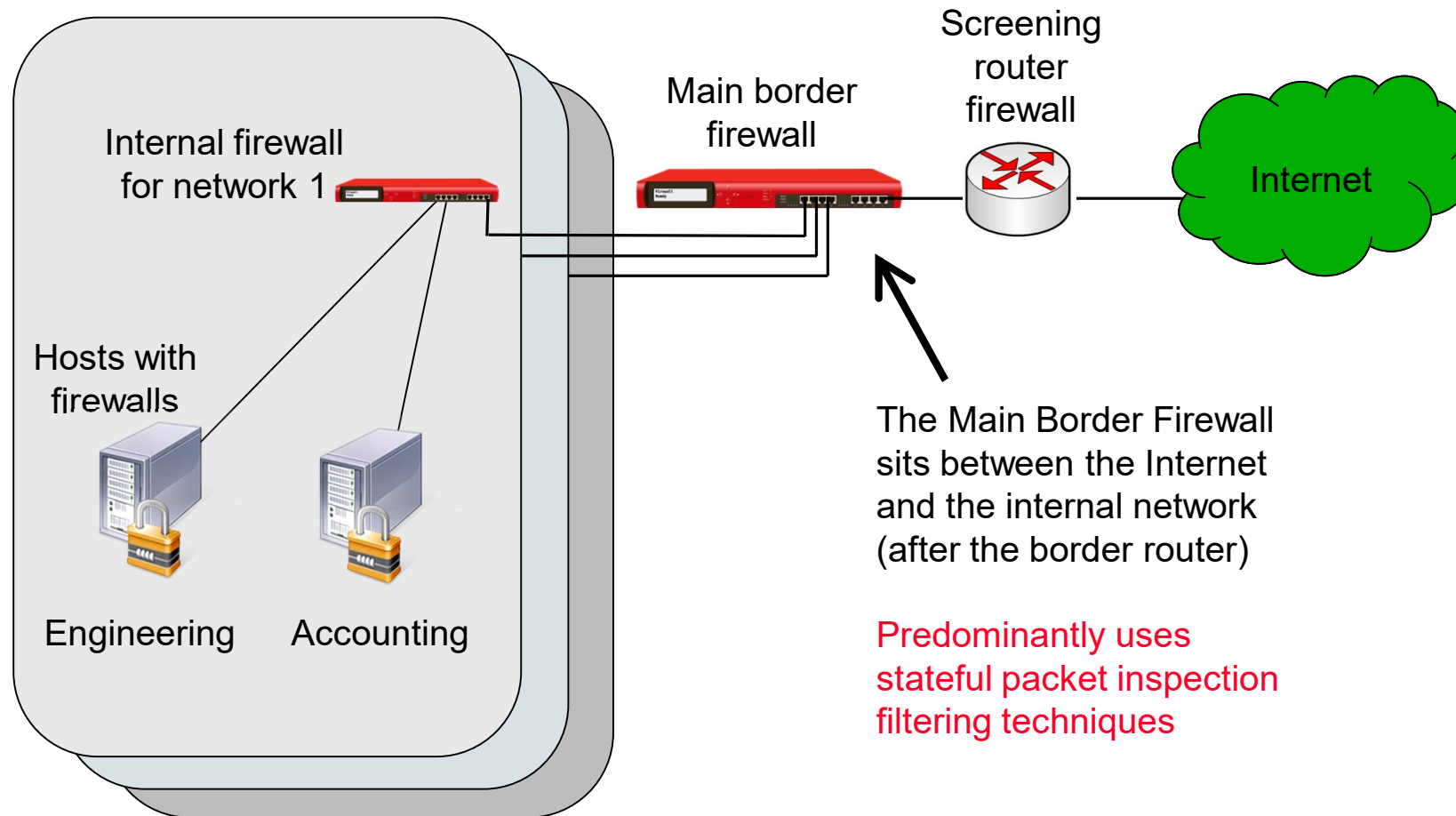
Stateful Packet Inspection (step 1)

Internal
client PC
60.55.33.12

TCP SYN Segment
From: 60.55.33.12 : 42013
To: 123.80.5.34 : 80

External
web server
123.80.5.34
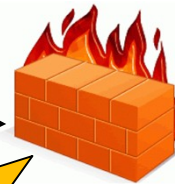
RULE TABLE:
Outgoing always allowed
Incoming always denied

Established
connections are
placed in the
connection table

CONNECTION TABLE (or state table):

Type	Internal IP addr	Internal Port	External IP addr	External Port	State	Seq. number incoming	Seq. number outgoing
TCP	60.55.33.12	42013	123.80.5.34	80	SYN sent	--	1,324,125,328

Stateful Packet Inspection (step 2)

Internal
client PC
60.55.33.12

TCP SYN/ACK Segment
From: 123.80.5.34:80
To: 60.55.33.12:42013

External
web server
123.80.5.34

Check table for SYN/ACK:
If it's Ok, pass the packet
and update the state

A good FW also
stores window sizes,
TCP options, etc.

CONNECTION TABLE (or state table):

Type	Internal IP addr	Internal Port	External IP addr	External Port	State	Seq. number incoming	Seq. number outgoing
TCP	60.55.33.12	42013	123.80.5.34	80	SYN rec'd	387 824 116	1 324 125 328

Multiple interfaces

- Most firewalls allow interfaces in ACLs
  - With two interfaces, naming the rules "in" and "out" works
  - With more interfaces, they must be numbered or named

- Example:
  EXTERNAL=if_1

  ....
 block in log on EXTERNAL {10.0.0.0/8} to any
  Rule blocks and logs incoming packets from 10.* network if received on interface EXTERNAL

- The connection table needs to contain information about interfaces:

Proto	Inter-face	IP addr	Port	Inter-face	IP addr	Port	State, seq. no, etc.	
TCP	1 (ext)	60.55.33.12	42013	8	10.8.1.5	443	…	From outside to our web server (DMZ)
TCP	2	10.2.2.32	2313	7	10.7.1.2	445	…	Internal traffic to file server
TCP	2	10.2.2.19	31295	7	10.7.1.2	445	…	

1 (EXT)   8 (DMZ)
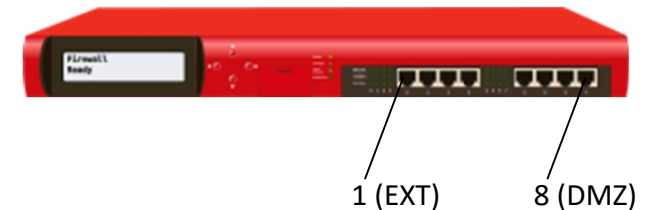
UDP and Stateful Packet Inspection

- For UDP: store IP addresses and port numbers in the connection table

- Store a timeout value for the entry

- Note: during this time period, the firewall is open for external traffic to the host
  - Attacker needs to know both IP address and port number to succeed
  - It may be possible to guess this? Perhaps to constantly send faked DNS responses to an internal system waiting for the window to open ??

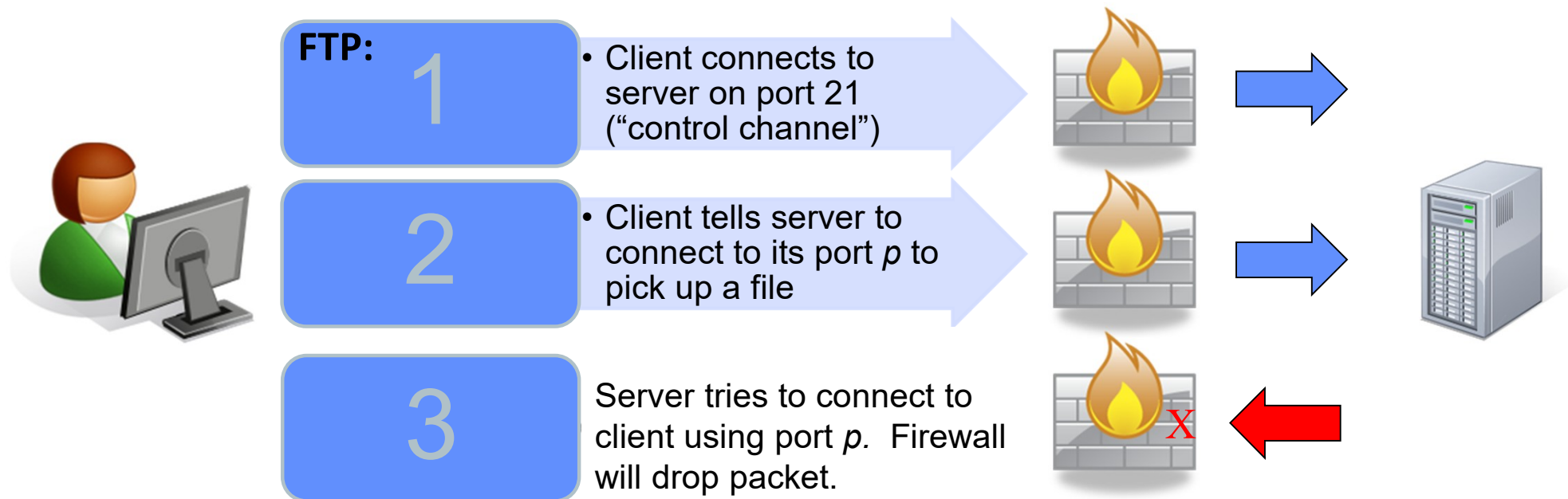| Type	Internal IP addr	Internal Port	External IP addr	External Port	State
TCP	60.55.33.12	42013	123.80.5.34	80	…
UDP	60.55.33.12	42111	1.8.33.4	53	Time=10s

Firewall principles

- Incoming packets:
  1. Check connection (state) table
  2. If not found, check the rule table (ACL list)
  3. If ok, pass the packet and update the state in connection table

- The rule table is normally evaluated in order: top → down
  - First match stops evaluation
  - Last rule in the main border firewall is always "Drop All"

- Each application-level protocol to inspect requires a module
  - More protocols → more processing power →may need dedicated hardware
  - Can detect that traffic is not legitimate
  - Example: lots of non-web traffic uses port 80 to become "firewall friendly"
  - Solution: do HTTP content filtering

Problems with FTP-like protocols

- Protocols that dynamically open new ports are problematic
  Avoid them!

FTP:

1 • Client connects to server on port 21 ("control channel")

2 • Client tells server to connect to its port p to pick up a file

3 Server tries to connect to client using port p. Firewall will drop packet.

SPI Firewalls – Summary

- **Dominating technique for Main Border Firewalls**

- **Security**
  - Very flexible – can handle almost any TCP/UDP/ICMP flow
  - Probably best choice if only one FW is used

- **Can do some application-level filtering**
  - Other important protocols: FTP, HTTP, DNS, …

- **Simple and can therefore be rather cheap**
  - Rules could be complex (good)
  - Checking ongoing connections is simple and fast
  - But if connection table grows very large → takes longer time

- **Stateful inspection is cost effective**
  - Many firewalls run on standard operating systems
  - Hardened system with special packet handling software

Example: FW-1 firewall features

Network Layer

	Attack Prevention Safeguards	Attacks Blocked
IP	- Enforce minimum header length - Restrict IP-UDP fragmentation - Enforce that header length indicated in IP header is not longer than packet size indicated by header - Enforce that packet size indicated in IP header is not longer than actual packet size - Scramble OS fingerprint - Control IP options	- IP Address Sweep Scan - IP Timestamp Attack - IP Record Route Attack - IP Source Route Attack - IP Fragment Denial-of-Service Attack - Loose Source Route Attack - Strict Source Route Attack - IP Spoofing Attack
ICMP	- Block large ICMP packets - Restrict ICMP fragments - Match ICMP requests and responses	- Ping-of-Death Attack - ICMP Flood

Firewall-1 was selected here because of a nice table. Other firewalls have similar features.
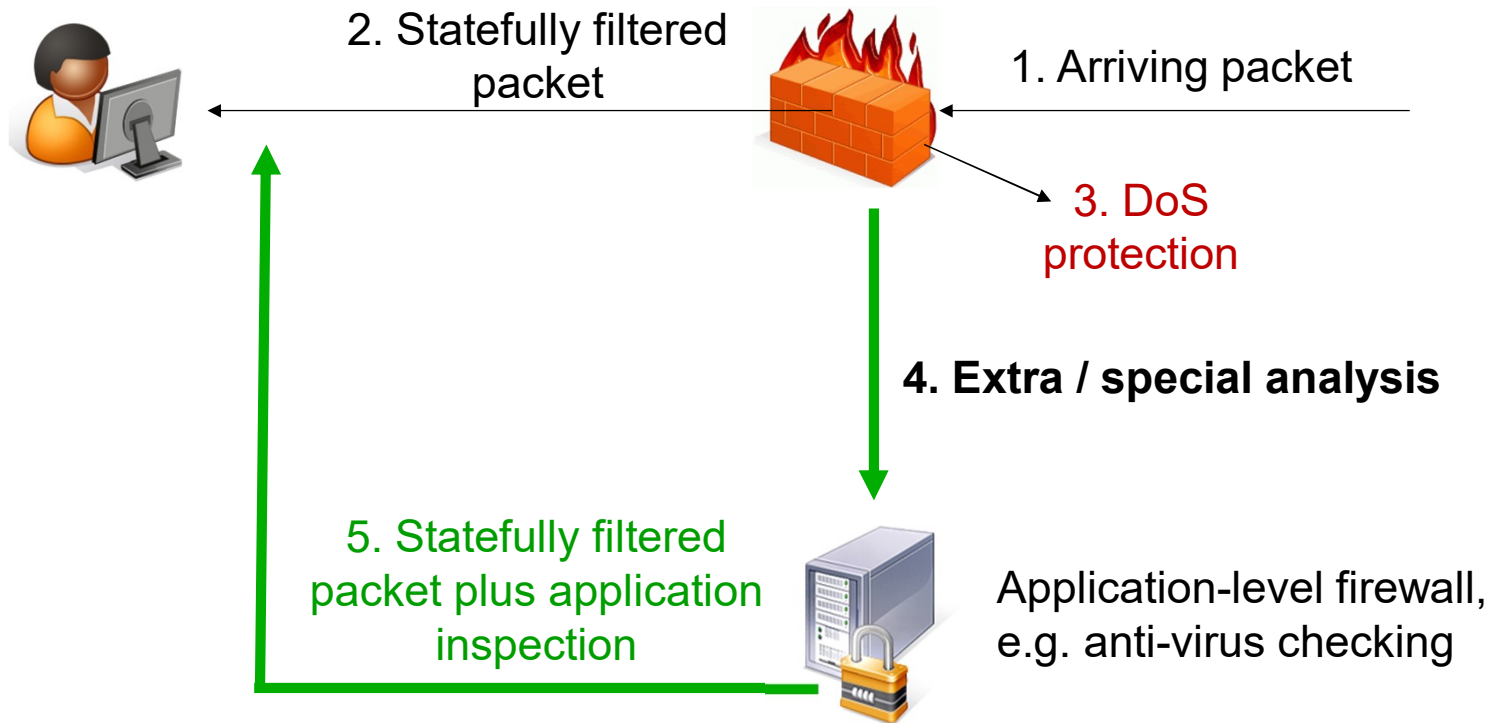
Example: FW-1 firewall features

Transport Layer

	Attack Prevention Safeguards	Attacks Blocked
TCP	- Enforce correct usage of TCP flags - Limit per-source sessions - Enforce minimum TCP header length - Block unknown protocols - Restrict FIN packets with no ACK - Enforce that TCP header length as indicated in header is not longer than packet size indicated by header - Block out state packets - Verify that first connection packet is SYN - Enforce 3-way handshake: Between SYN and SYN-ACK, client can send only RST - Enforce 3-way handshake enforcement: Between SYN and connection establishment, server can send only SYN-ACK or RST - Block SYN on established connection before FIN or RST packet is encountered - Restrict server-to-client packets belonging to old connections - Drop server-to-client packets belonging to old connections if packets contain SYN or RST - Enforce minimum TCP header length - Block TCP fragments - Block SYN fragments - Verify TCP packet sequence number for packets belonging to an existing session	- ACK Denial-of-Service Attack - SYN Attack - Land Attack - Tear Drop Attack - Session Hijacking Attack - Jolt Attack - Bloop Attack - Cpd Attack - Targa Attack - Twinge Attack - Small PMTU Attack - TCP-Based Attacks Spanning Multiple Packets - XMAS Attacks - Port Scan
UDP	- Verify UDP length field - Match UDP requests and responses	- UDP Flood Attacks - Port Scan

Example: FW-1 firewall features

Application Layer/Presentation Layer

	Attack Prevention Safeguards	Attacks Blocked
HTTP Server	• Limit maximum URL length • Limit maximum request header length • Prohibit binary characters in HTTP response headers • Prohibit binary characters in HTTP requests • Block user-defined URLs • Restrict non-RFC HTTP methods • Enforce HTTP security on non-standard ports (ports other than 80) • Restrict download of user-defined files	• Cross-Site Scripting Attacks • HTTP-based attacks spanning multiple packets • User-Defined Worms & Mutations
DNS	• Restrict DNS zone transfers • Restrict usage of DNS server as a public server • Provide separate DNS service for private vs. public domains	• DNS Query Malformed Packet Attacks • DNS Answer Malformed Packet Attacks • DNS Query-Length Buffer Overflow • DNS Query Buffer Overflow - Unknown Request/Response • Man-in-the-Middle Attack
FTP	• Analyze and restrict hazardous FTP commands	• Passive FTP Attacks • FTP Bounce Attack • Client and Server Bounce Attacks • FTP Port Injection Attacks • Firewall Traversal Attack • TCP Segmentation Attack

Other systems can help the firewall

2. Statefully filtered packet

1. Arriving packet

3. DoS protection

4. **Extra / special analysis**

5. Statefully filtered packet plus application inspection

Application-level firewall, e.g. anti-virus checking
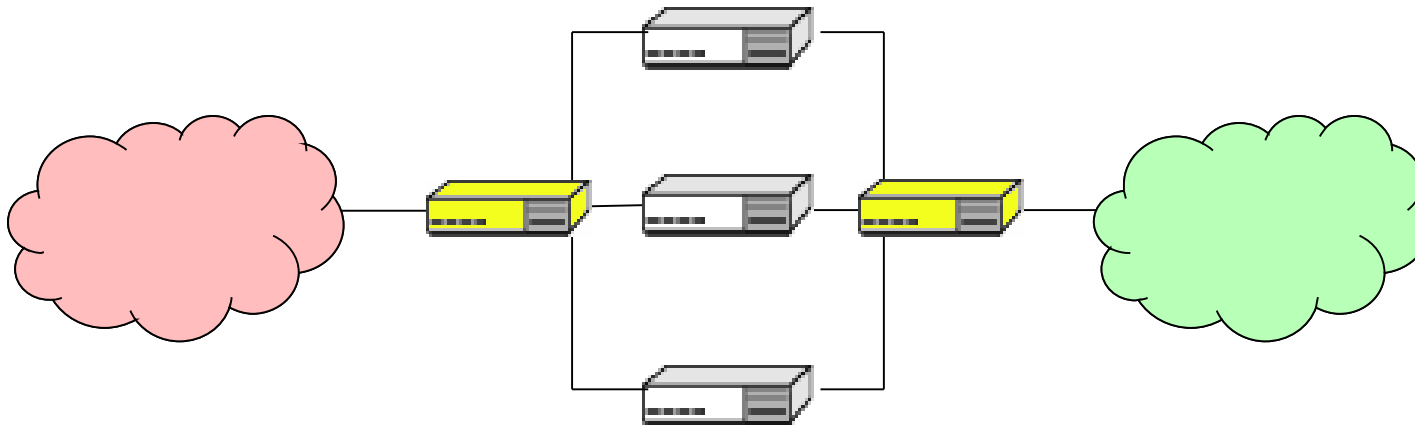
Clustering for Redundancy

- It is possible to use multiple firewalls (clustering) to achieve higher availability

- All firewalls in the cluster must have the same state information
  - They communicate via a private network connection

- One method:
  - Both FWs listen to network traffic but only the master forwards traffic
  - The slave gets periodic messages from the master ("I'm alive")
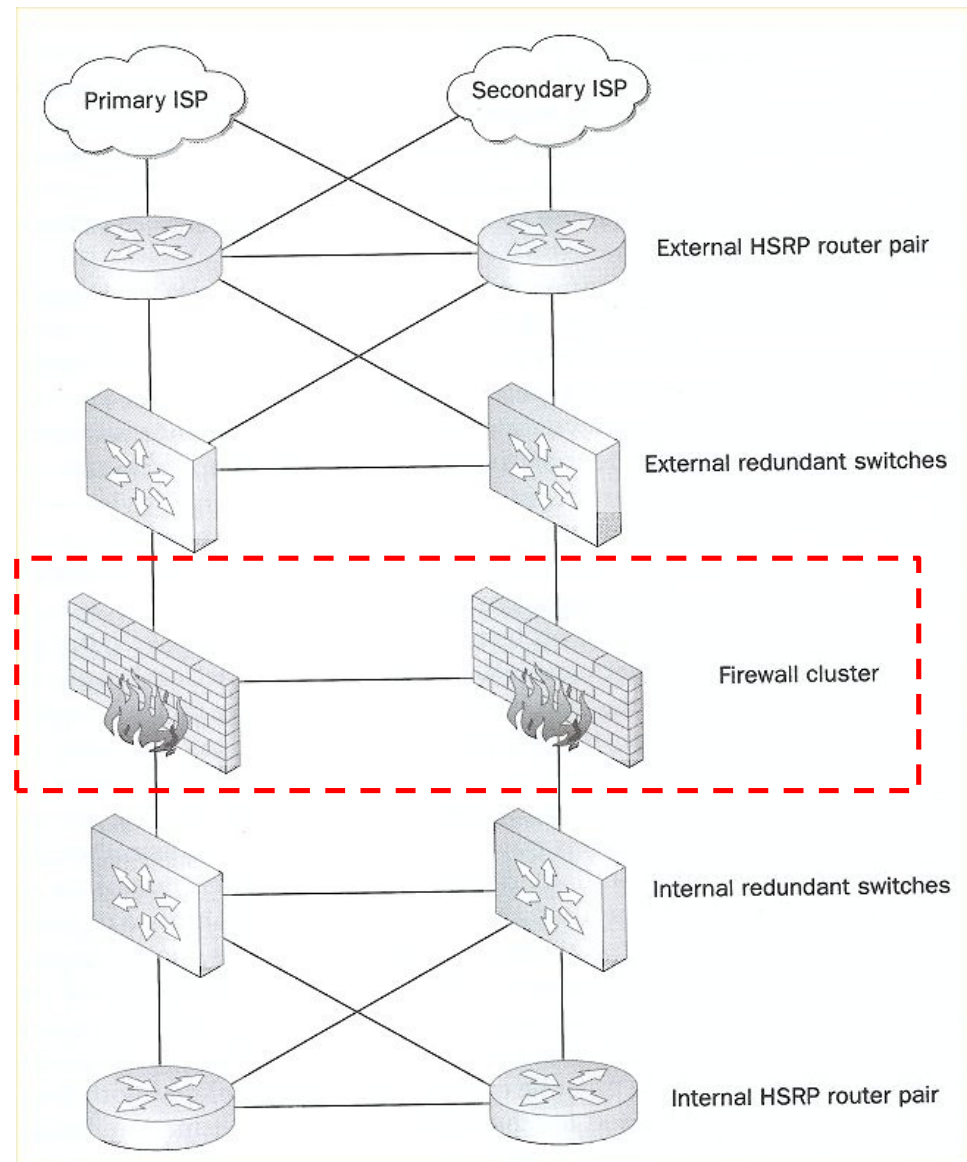 - If messages cease, slave immediately takes over and forwards packets
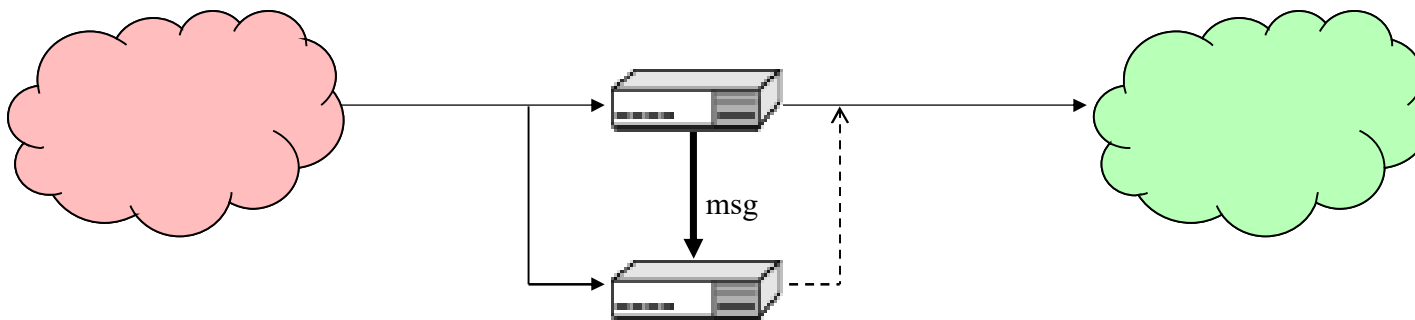
Clustering for load sharing

- Front-end load balancers distribute traffic
  - Same firewall always handles one traffic flow
  - State not shared between firewalls

- If a firewall dies, its connections are broken

- But if the load balancer dies? More complex solutions needed, next slide

Clustering for redundancy and load sharing

Takeaway: it is complicated to make it truly redundant

Primary ISP

Secondary ISP

External HSRP router pair

External redundant switches

Firewall cluster

Internal redundant switches
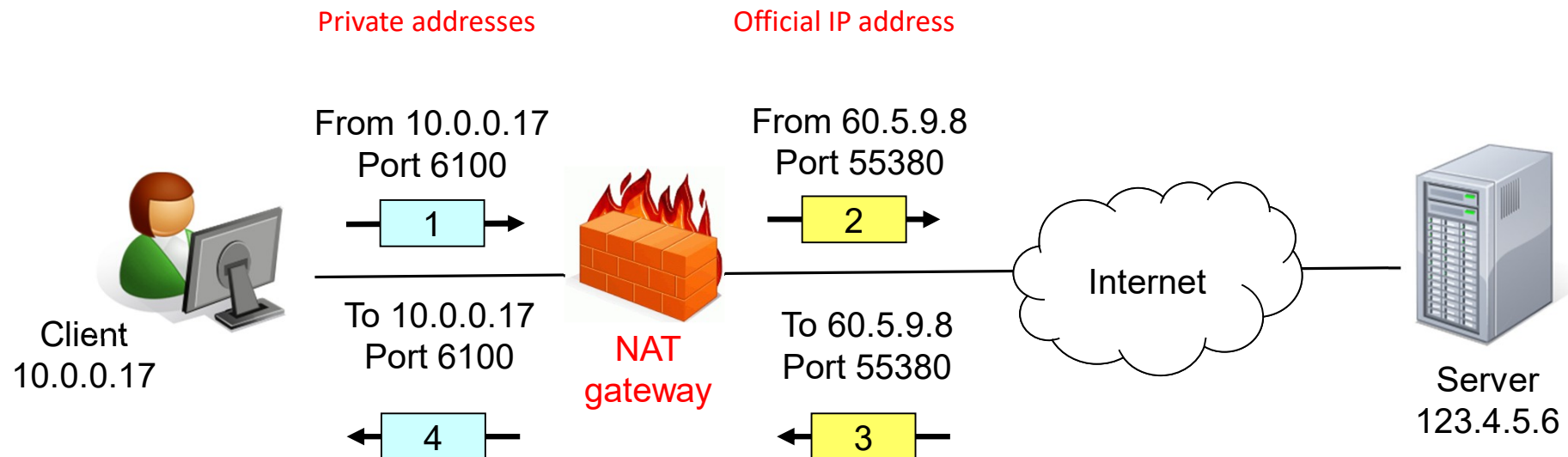
Internal HSRP router pair

Network Address Translation (NAT)

Can be used as a firewall – it blocks external connections to the network

Network Address Translation (NAT)

- Internal IP addresses translated to one or more official addresses (RFC 2663)
    - Can use private (RFC 1918) addresses internally

- NAPT – Network Address Port Translation is what most consider as NAT ("Traditional NAT" – RFC 3022)

- Translates [IP_address, port_number] on internal hosts to an official IP address and another port number
    - Dynamic: translation is done per connection and only when needed
    - An organization may have a limited number of public IP addresses
    - Can have thousands of internal IP addresses for each public IP address
    - Each external IP address can have one connection per port number (i.e. 65,535 active connections)
    - Multiple external IP addresses can be used if one is not enough

- Clients can request NAT device to open ports – Port Control Protocol (PCP – RFC7843)
    - Typically used for home users
    - Part of Universal Plug and Play (UPnP) for Internet Gateway Devices
    - UDP port 5350 (client) and 5351 (server)

Network Address Translation (NAT)

Private addresses   Official IP address

Client
10.0.0.17

From 10.0.0.17
Port 6100

| 1 |

To 10.0.0.17
Port 6100

| 4 |

NAT
gateway

From 60.5.9.8
Port 55380

| 2 |

To 60.5.9.8
Port 55380

| 3 |

Internet

Server
123.4.5.6

Translation
Table

Internal		External	
IP Addr	Port	IP Addr	Port
10.0.0.17	6100	60.5.9.8	55380
. . .	. . .	. . .	. . .

Information stored by NAT gateways

- TCP and UDP:
  - Source and dest IP addresses
  - Source and dest ports

- ICMP:
  - Source and dest IP addresses
  - ICMP query ID
  - IP address in the actual reply must also be modified by NAT gateway

- All other (if supported):
  - Source and dest IP addresses
  - protocol field in IP header

- Session termination [more details in RFC 2663]
  - NAT gateway cannot assume that FIN or RSTs will be delivered to hosts
  - Timeout since last seen activity, device dependent
  - For TCP typically 4 minutes after FIN messages

Security benefits with NAT

- **Hides internal IP addresses** and port numbers
  - Externally only translated addresses and port numbers visible
  - Much harder to identify individual hosts
  - External attackers cannot create connections to internal computers

- Attackers can only spoof packets for open connections
  - Ports on internal machines are invisible
  - Sniffers can read stand-in IP addresses and port numbers
  - ACK scans possible for open connections

- Good complement to a stateful firewall
  - Normally implemented in the border (screening) router
  - Or in the firewall itself if no screening router used

- No active clients → translation table empty → no incoming traffic allowed

NAT disadvantages

- No inspection of packet contents

- NAT gateways need to reassemble IP fragments before forwarding
  - TCP header only present in first fragment – needed to look up entry in table
  - Fragments can be forwarded but it must at least do "virtual" fragmentation
  - Keep state? Drop?

- Problems with protocols relying on clients' IP addresses
  - Applications sending client IP-addresses will not work (IPsec, VoIP, SIP, …)
  - Protocols requesting the external party to connect back will not work (FTP)
  - Good from a security point of view, but…

- Examples of problematic protocols:
  - Virtual private networks (VPN) using IPsec
  - VoIP, h323, FTP, P2P, IRC
  - Mobile IP, …

- Authorizing users based on the IP address from a NATed site will fail
  - All users seem to originate from the same computer
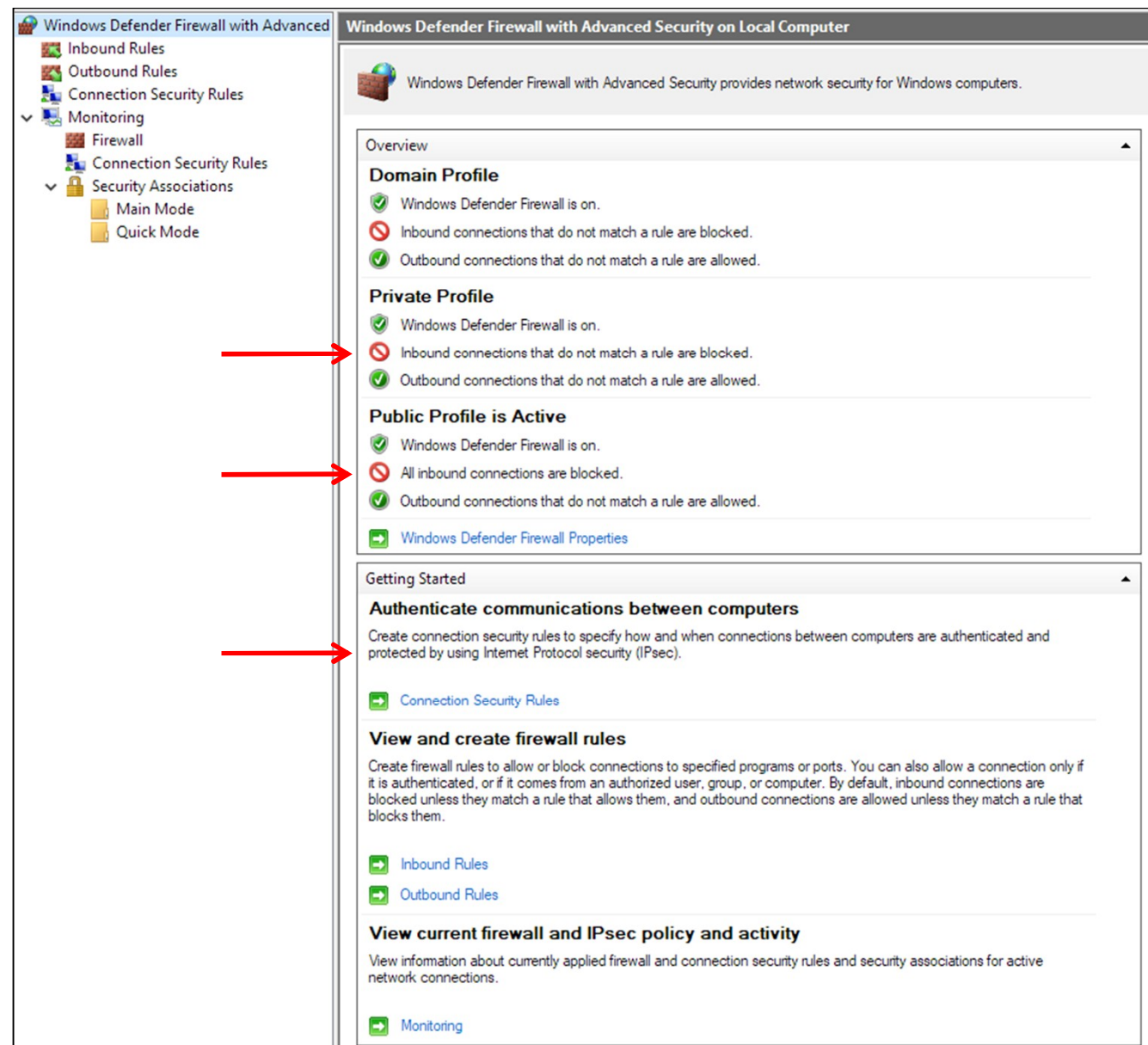 - Never offer services that rely only on the IP addresses!
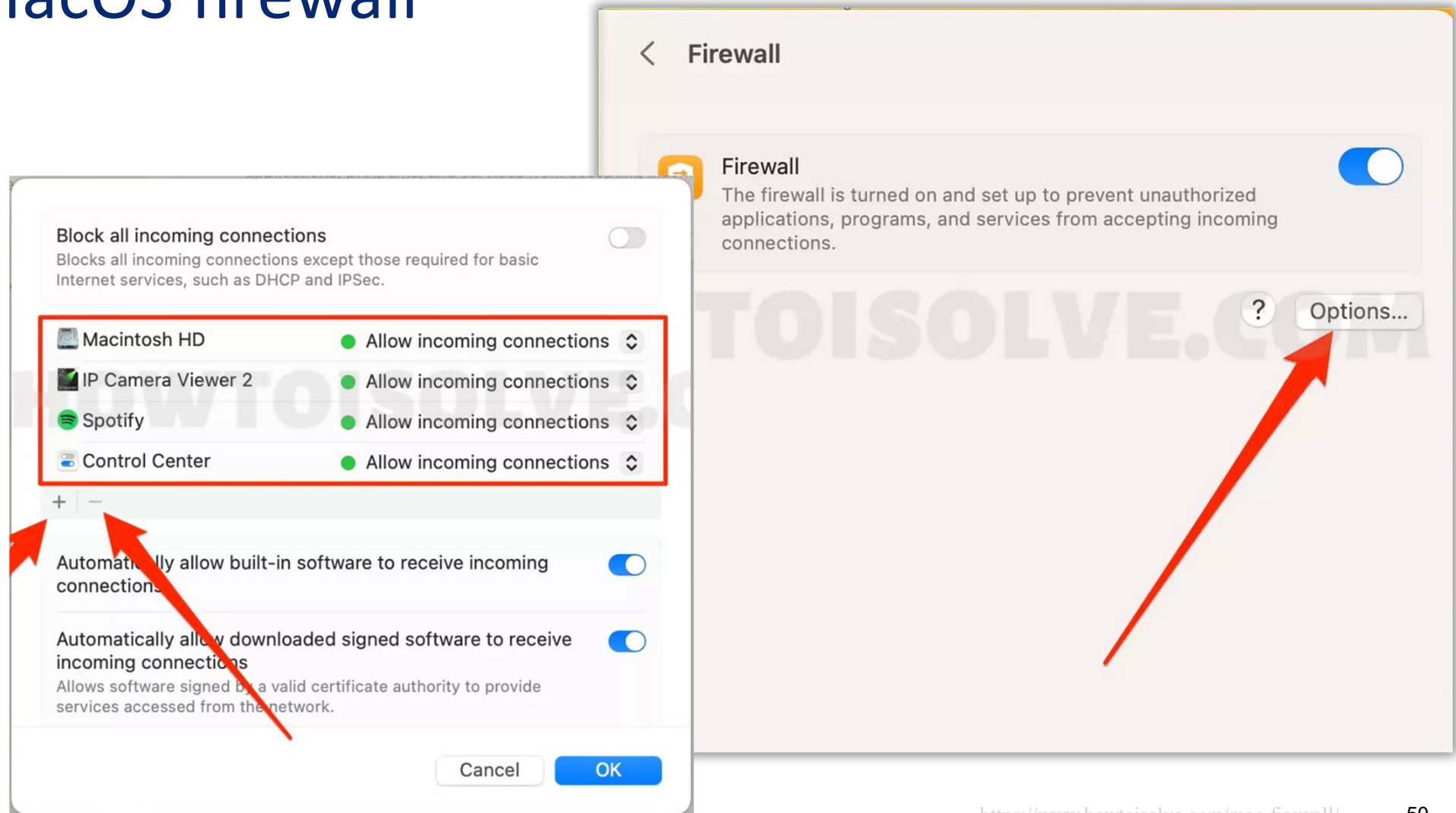
Host-based / personal firewalls

Windows

- Stateful firewall introduced with Windows Vista 2007 which was <u>enabled by default</u>
  - GUI for user interaction
  - Domain users can have remote configured firewalls
  - Few users know what is legitimate traffic

- Different profiles can be active based on location
  - Home, domain and public
  - A profile can be associated with a network interface

- Possible to control which application can use a port (i.e. to be a service)
  - Can notify user if a program is blocked
  - Linux similar functionality: AppArmor kernel security module

- Possible to control what interface rule applies to
  - May be different rules for LAN, WLAN, 3G, …

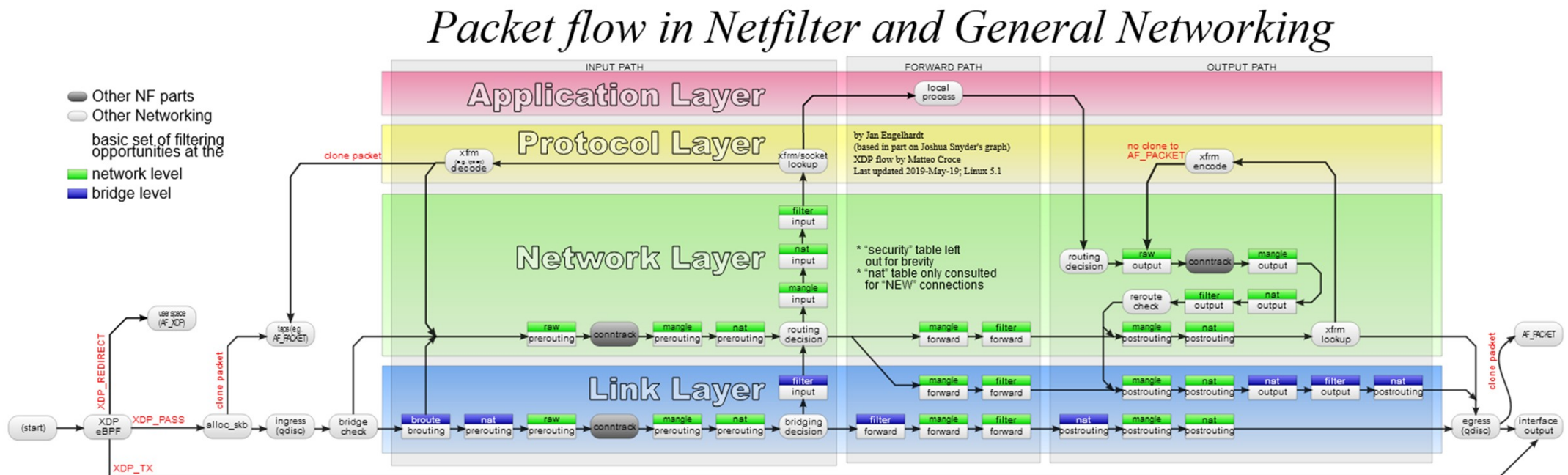- Possible to allow remote connections (in and out) only if IPsec (encrypted IP) is used
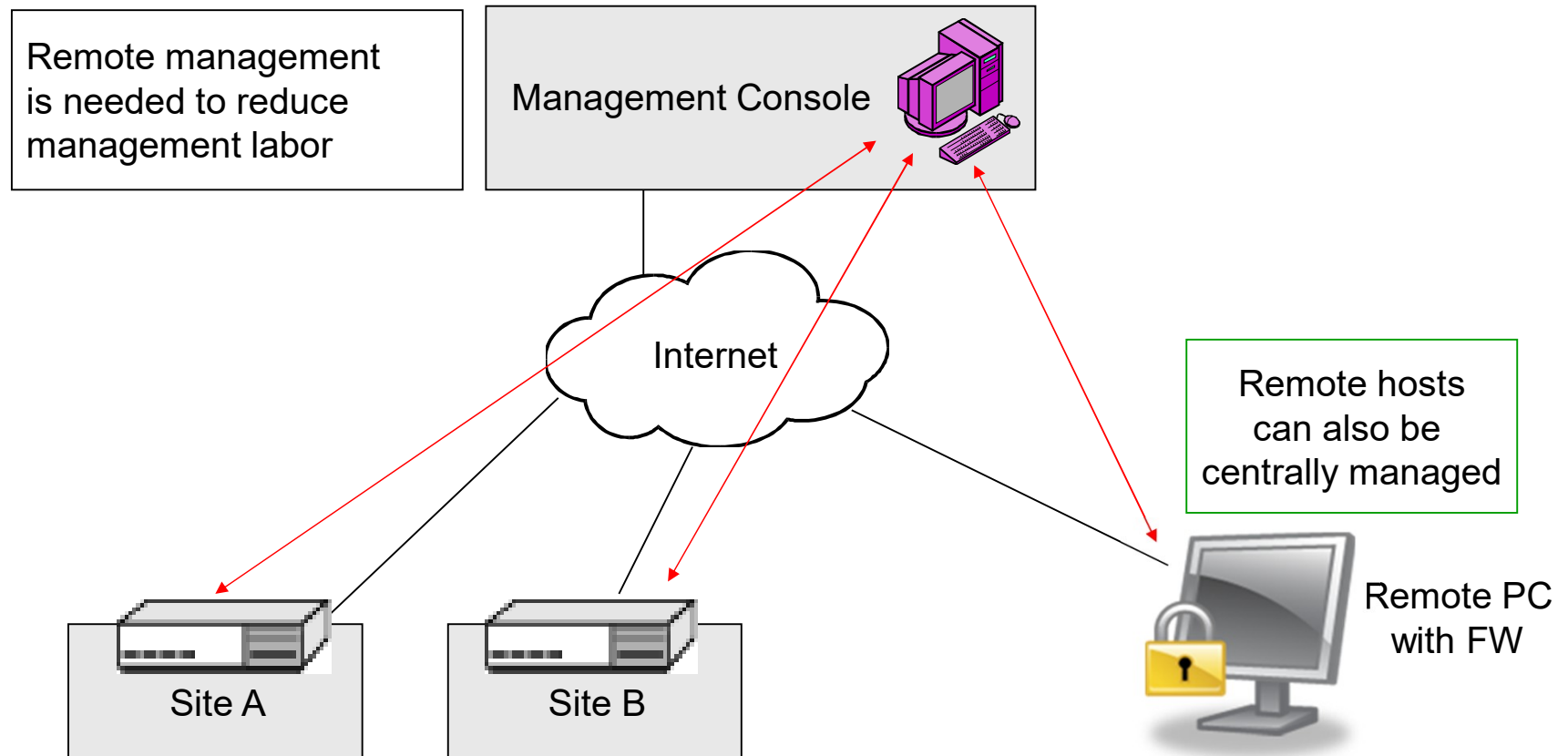
Windows defender firewall profiles

MacOS firewall

Linux iptables / Netfilter

Packet flow in Netfilter and General Networking

Centralized Firewall Management

Remote management is needed to reduce management labor

Management Console

Internet

Remote hosts can also be centrally managed

Site A

Site B

Remote PC with FW

Summary

- Firewall types
  - Static and dynamic (stateless) filters
  - Stateful packet inspection (transport and/or application layer)
  - Circuit-level gateways and application proxies

- Use screening router + main border firewall
  - Screening router removes all obvious garbage and does ingress and egress filtering
  - Main border firewall performs deeper inspection

- Internal firewalls for segmentation
  - Many layers of defense

- NAT for security

- Host-based (personal) firewalls on clients and servers
  - Remote clients' only defense, e.g. on guest networks
  - Also final outpost if problems occur on internal networks