Network Security - Week 8

Manuel E. Correia

DCC/FCUP

2025

Countermeasures to DoS Attacks

- DoS attacks cannot be prevented entirely
- High traffic volumes may be legitimate

1. Attack prevention and preemption
2. Attack detection and filtering
3. Attack source traceback and identification
4. Attack reaction

Countermeasures to DoS Attacks

- DoS attacks cannot be prevented entirely
- High traffic volumes may be legitimate

1. Attack prevention and preemption
   - Before the attack occurs
   - Enforce policies for resource consumption
   - Provide backup resources available on demand
2. Attack detection and filtering
3. Attack source traceback and identification
4. Attack reaction

Countermeasures to DoS Attacks

- DoS attacks cannot be prevented entirely
- High traffic volumes may be legitimate

1. Attack prevention and preemption
2. Attack detection and filtering
   - During the attack
   - Look for suspicious patterns of behavior
   - Filter packets likely to be part of the attack
3. Attack source traceback and identification
4. Attack reaction

Countermeasures to DoS Attacks

- DoS attacks cannot be prevented entirely
- High traffic volumes may be legitimate

1. Attack prevention and preemption
2. Attack detection and filtering
3. Attack source traceback and identification
   - During/after the attack
   - Identify sources of attack
   - Prepare whitelists/blacklists
4. Attack reaction

Countermeasures to DoS Attacks

- DoS attacks cannot be prevented entirely
- High traffic volumes may be legitimate

1. Attack prevention and preemption
2. Attack detection and filtering
3. Attack source traceback and identification
4. Attack reaction
   - After the attack
   - Eliminate effects of the attack
   - I.e. cleanup the system

DoS Attack Prevention

Block Spoofed Source Addresses (RFC 2827)

- Ingress filtering
- On routers as close to the source as possible
- Still far too rarely implemented

DoS Attack Prevention

Block Spoofed Source Addresses (RFC 2827)

- Ingress filtering
- On routers as close to the source as possible
- Still far too rarely implemented

Rate control in upstream distribution nets

- Target specific packet types
- E.g. some ICMP, some UDP, TCP/SYN
- Leverage known amplification attacks

DoS Attack Prevention

Block Spoofed Source Addresses (RFC 2827)

- Ingress filtering
- On routers as close to the source as possible
- Still far too rarely implemented

Rate control in upstream distribution nets

- Target specific packet types
- E.g. some ICMP, some UDP, TCP/SYN
- Leverage known amplification attacks

Use modified TCP connection handling

- SYN cookies when table is full
- Selective/random drop when table is full
- Avoid a state where no further connections can be established

DoS Attack Prevention - High level

- Block IP directed broadcasts
- Block suspicious services and combinations
- Use mirrored and replicated servers when high-performance and reliability is required
- Manage application-level attacks with a form of graphical puzzle to distinguish legitimate human requests from bots

Responding to DoS Attacks

Having a good incident response plan...

- Details on how to contact technical personal for ISP
- Needed to impose traffic filtering upstream
- Details of how to respond to the attack

Responding to DoS Attacks

Having a good incident response plan...

- Details on how to contact technical personal for ISP
- Needed to impose traffic filtering upstream
- Details of how to respond to the attack

... and good proactive measures

- Antispoofing, directed broadcast and rate limiting filters
- Ideally, network monitors and Intrusion Detection Systems (soon) to detect and raise warnings over abnormal traffic patterns
  - How can we distinguish normal from abnormal?

Diagnosing DoS Attacks

- Identify the type of attack
- Have ISP trace packet flow back to the source
- Implement contingency plan
- Update incident response plan

Diagnosing DoS Attacks

- Identify the type of attack
  - Capture and analyse packets
    - Intrusion Detection Systems (soon)
  - Design filters to block attack traffic upstream
    - Firewalls (soon)
  - ... or identify and correct system application/bug
- Have ISP trace packet flow back to the source
- Implement contingency plan
- Update incident response plan

Diagnosing DoS Attacks

- Identify the type of attack
- Have ISP trace packet flow back to the source
  - May be difficult and time consuming
  - Necessary if planning legal action
  - Accountability is key
- Implement contingency plan
- Update incident response plan

Diagnosing DoS Attacks

- Identify the type of attack
- Have ISP trace packet flow back to the source
- Implement contingency plan
  - Switch to alternate backup servers
  - Commission new servers at a new site with new addresses
- Update incident response plan

Diagnosing DoS Attacks

- Identify the type of attack
- Have ISP trace packet flow back to the source
- Implement contingency plan
- Update incident response plan
  - Analyze the attack and the response for future handling
  - Attack strategies are not static
  - So neither can be the response plan

Responding to DoS Attacks

- Identify type of attack
  - Capture and analyze packets - Intrusion Detection Systems
  - Design filters to block attack traffic upstream - Firewalls
  - Identify and correct system/application bugs

Responding to DoS Attacks

- Identify type of attack
  - Capture and analyze packets - Intrusion Detection Systems
  - Design filters to block attack traffic upstream - Firewalls
  - Identify and correct system/application bugs
- Have ISP trace packet flow back to source
  - Often difficult and time consuming
  - Necessary when planning legal action

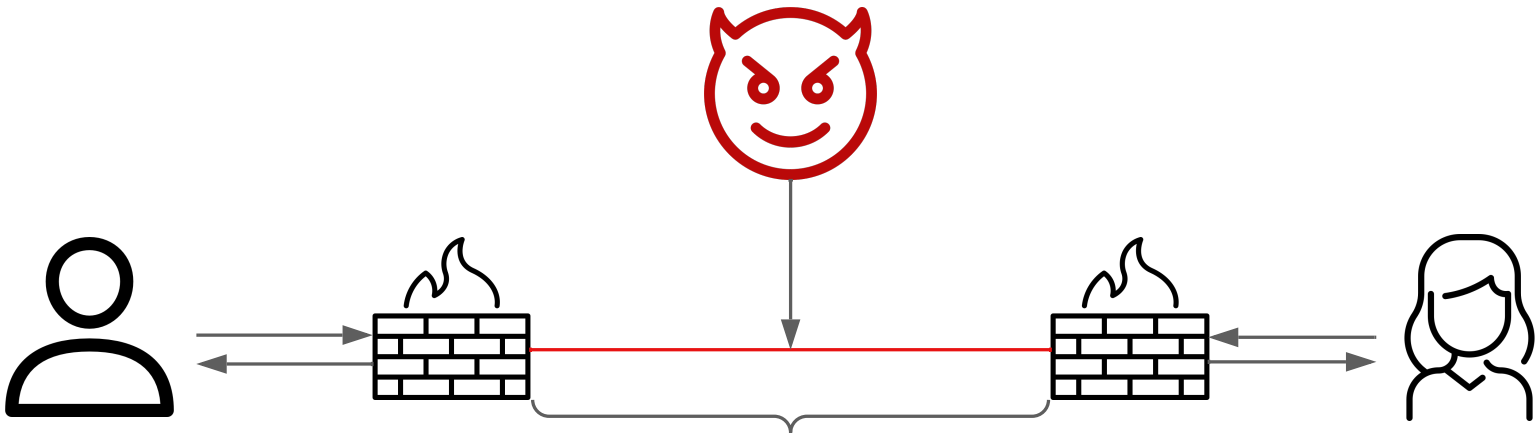
Responding to DoS Attacks

- Identify type of attack
  - Capture and analyze packets - Intrusion Detection Systems
  - Design filters to block attack traffic upstream - Firewalls
  - Identify and correct system/application bugs
- Have ISP trace packet flow back to source
  - Often difficult and time consuming
  - Necessary when planning legal action
- Implement contingency plan
  - Switch to alternate backup servers
  - Commission fresh servers at a new site with new addresses

Responding to DoS Attacks

- Identify type of attack
  - Capture and analyze packets - Intrusion Detection Systems
  - Design filters to block attack traffic upstream - Firewalls
  - Identify and correct system/application bugs
- Have ISP trace packet flow back to source
  - Often difficult and time consuming
  - Necessary when planning legal action
- Implement contingency plan
  - Switch to alternate backup servers
  - Commission fresh servers at a new site with new addresses
- Update incident response plan
  - Analyze the attack and response for future handling

Firewalls



- Firewall decides what to let in to internal network and/or to let out
- Access control for the network
- At a multitude of granularity levels

Managing what comes and and goes out

A firewall is like a **secretary**

- To meet with an executive:
  1. Contact the secretary
  2. Secretary will assess if the meeting is important
  3. Many requests are filtered according to relevance metrics

Managing what comes and and goes out

A firewall is like a **secretary**

- To meet with an executive:
  1. Contact the secretary
  2. Secretary will assess if the meeting is important
  3. Many requests are filtered according to relevance metrics

- If you want to meet the chair of CS department...
  - Secretary will do some filtering
- If you want to meet the President
  - Secretary will do a lot of filtering

Firewall Access Policy

Criteria under which “meetings can be scheduled”

- Filtering done according to an access policy
- Types of traffic
- Address ranges and protocols
- Applications and content types

Firewall Access Policy

Criteria under which “meetings can be scheduled”

- Filtering done according to an access policy
- Types of traffic
- Address ranges and protocols
- Applications and content types

- Specification of which traffic types the org needs to support
- Then refined to detail the filter elements, implemented with an appropriate firewall topology
- Bad configuration **can lead to loss of communication**

Capabilities and Limits

Capabilities

- Defines a single choke point
- Provides a location for monitoring security events
- Convenient platform for several internet functions that are not security related (e.g. NAT)
- Can serve the platform for IPSec (tunnel mode)

Capabilities and Limits

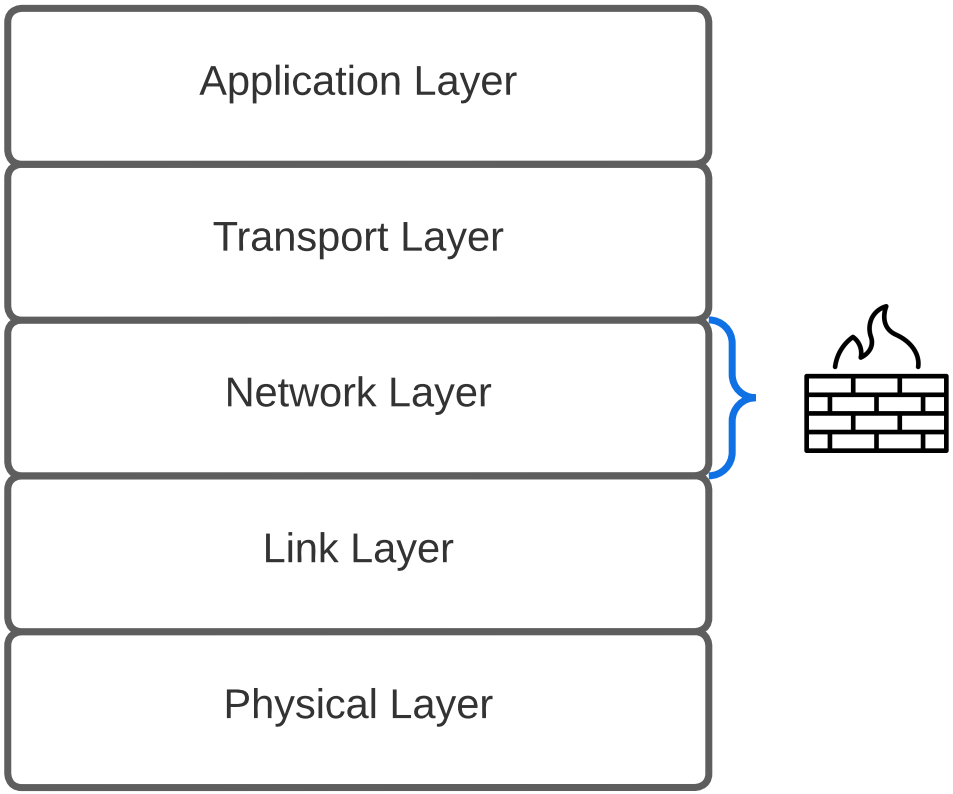
Capabilities

- Defines a single choke point
- Provides a location for monitoring security events
- Convenient platform for several internet functions that are not security related (e.g. NAT)
- Can serve the platform for IPSec (tunnel mode)

Limitations

- Cannot protect against attacks bypassing the firewall
- May not protect fully against internal threats
- Laptop, PDA, or portable storage device may be infected outside corporate network, and then used internally
- Improperly secured wireless LAN can be accessed outside the organization

Packet Filter- High-level view



- Operates at the network layer
- Observes IP packets and assesses their importance
- Why can this be incompatible with IPSec?

Packet Filter - Criteria

Packet filtering depends on:

- Source IP address
- Destination IP address
- Source Port
- Destination Port
- Flag bits (SYN, ACK, etc.)
- Egress or ingress

Packet Filter - Criteria

Packet filtering depends on:

- Source IP address
- Destination IP address
  - Allows for excluding problematic sources (blacklist/whitelist)
- Source Port
- Destination Port
- Flag bits (SYN, ACK, etc.)
- Egress or ingress

Packet Filter - Criteria

Packet filtering depends on:

- Source IP address
- Destination IP address
  - Allows for excluding problematic sources (blacklist/whitelist)
- Source Port
- Destination Port
  - Easy to profile and avoid attacks on problematic services
- Flag bits (SYN, ACK, etc.)
- Egress or ingress

Packet Filter - Criteria

Packet filtering depends on:

- Source IP address
- Destination IP address
  - Allows for excluding problematic sources (blacklist/whitelist)
- Source Port
- Destination Port
  - Easy to profile and avoid attacks on problematic services
- Flag bits (SYN, ACK, etc.)
  - Remember how we can DoS on TLS?
- Egress or ingress

Packet Filter - Strengths x Weaknesses

Advantages

- Speed
- Simplicity
- Transparent to users

Packet Filter - Strengths x Weaknesses

Advantages

- Speed
- Simplicity
- Transparent to users

Disadvantages

- No concept of state
- Vulnerable to attacks on TCP/IP bugs
- Cannot see TCP connections
- Unknowing of application data and context

Packet Filter - Configuration

Configured via Access Control Lists (ACLs)

Action	Source IP	Dest IP	Source Port	Dest Port	Protocol	Flag Bits
Allow	Inside	Outside	Any	80	HTTP	Any
Allow	Outside	Inside	80	>1023	HTTP	ACK
Deny	All	All	All	All	All	All

- Traffic is restricted to web browsing:
- Accept all outgoing HTTP traffic to port 80
- Accept all incoming HTTP ACK replies
- Reject everything else

Packet Filter Workarounds

Issues

- Cannot prevent attack on application bugs
- Limited logging functionality
- Advanced user authentication not supported
- Improper configuration can lead to breaches

Packet Filter Workarounds

Issues

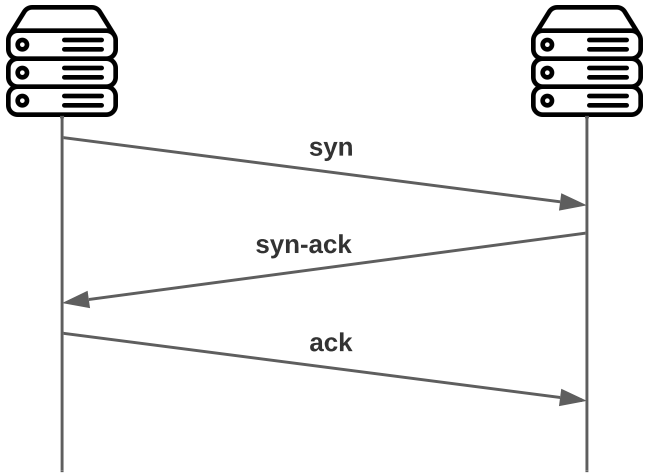
- Cannot prevent attack on application bugs
- Limited logging functionality
- Advanced user authentication not supported
- Improper configuration can lead to breaches

Attacks

- IP address spoofing
- Source route attacks
- Tiny fragment attacks

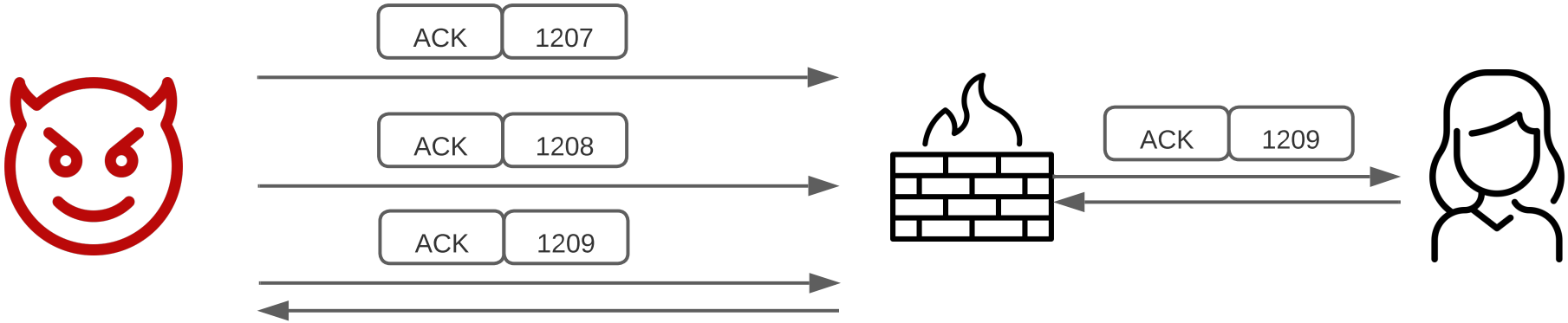
Packet filter exploits - Port scanning via TCP

Recall TCP (again)



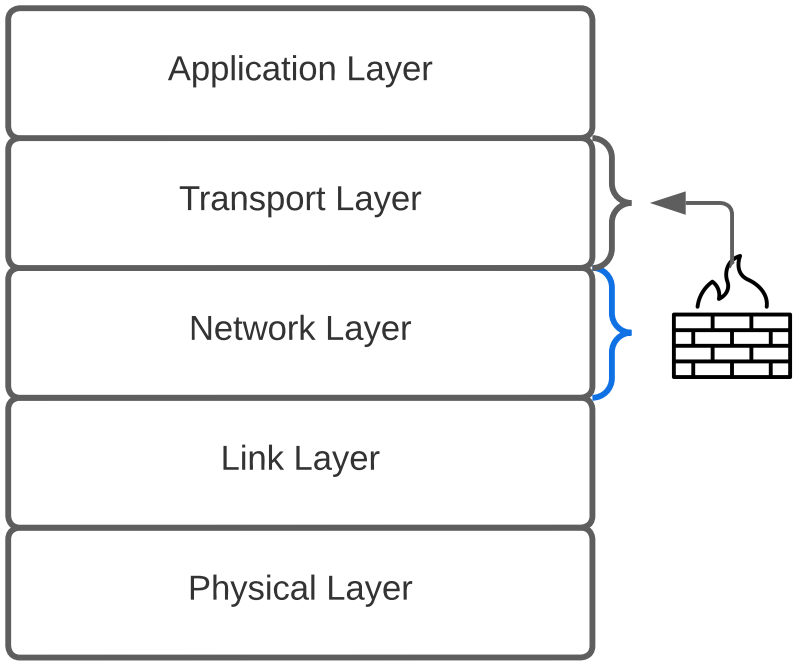
1. A client sends a SYN (synchronize) message
2. The server replies with a SYN-ACK message
3. The client concludes with a ACK (acknowledge) message
   - What if we sent an unrelated ACK message?
   - Mismatched sequence numbers
   - The server replies with RST (TCP reset)
   - I think you are confused, buddy. Try again

Packet filter exploits - Port scanning via TCP



- Attacker gets to know 1029 is operational
- Handshake was unsuccessful, but that was never the point
- Firewall knows TCP traffic is allowed...
- ... but lacks *context* to know if it makes sense

Stateful Packet Filter



- Adds state to the packet filter
- Operates at the transport layer
- Remembers TCP connections (e.g. flag bits)
- Can even remember UDP packets (e.g. DNS requests)

Connection State Table - Example

Source Address	Source port	Destination Address	Destination Port	Connection State
192.168.1.100	1030	210.9.88.29	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
219.22.123.32	2112	192.168.1.6	80	Established
210.99.212.18	3321	192.168.1.6	80	Established
24.102.32.23	1025	192.168.1.6	80	Established

Stateful Packet Filter

Advantages

- Can do everything a packet filter can
- Keeps track on ongoing connections
- Relies on protocol logic to detect misbehaviors
  - Avoids TCP ACK scan

Stateful Packet Filter

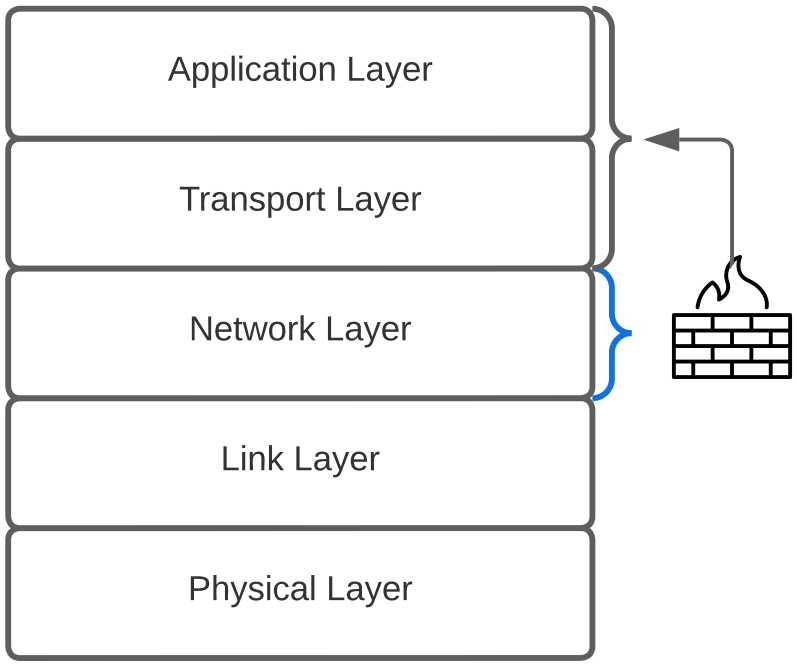
Advantages

- Can do everything a packet filter can
- Keeps track on ongoing connections
- Relies on protocol logic to detect misbehaviors
  - Avoids TCP ACK scan

Disadvantages

- Cannot see application data
  - Lacks internal application logic
  - Thus cannot accurately detect deviations from *expected behavior*
- Slower than packet filtering

Application Proxy



- A *proxy* is something that acts on your behalf
- Application proxy looks at incoming application data
- Verifies that data is safe before allowing passage

Application Proxy

a.k.a. Application-Level Gateway

Additional security layer

- For every supported application protocol
  - SMTP, POP3, HTTP, SSH, ...
  - Validation done at the data granularity
  - Spoofing packet implies convincing proxy to accept

Application Proxy

a.k.a. Application-Level Gateway

Additional security layer

- For every supported application protocol
  - SMTP, POP3, HTTP, SSH, ...
  - Validation done at the data granularity
  - Spoofing packet implies convincing proxy to accept

- Large amount of processing per connection
- Can enforce application-specific policies
- Highly configurable

Application Proxy

Advantages

- Complete view of connections and application data
  - Can capture nuanced behavior
  - E.g. disable specific features, or specify execution criteria
- Filter bad data at application layer
  - Prevents software-level errors and vulnerability exploitation
  - E.g. macros allowing for SQL injection or buffer overflow

Application Proxy

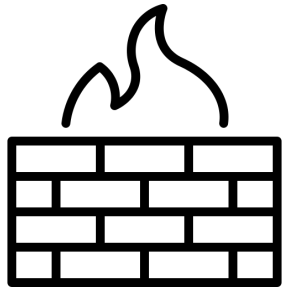
Advantages

- Complete view of connections and application data
  - Can capture nuanced behavior
  - E.g. disable specific features, or specify execution criteria
- Filter bad data at application layer
  - Prevents software-level errors and vulnerability exploitation
  - E.g. macros allowing for SQL injection or buffer overflow

Disadvantages

- Performance takes a toll – yet another security layer
- Each application must have the associated proxy code

Firewall Policies



Permissive

Allow by default; block some

- Easy to make mistakes
- Mistakes can lead to security breaches
- Exploits can be covert, i.e. not obvious that they are occurring

Restrictive

Block by default; allow some

- Much more secure
- Mistakes can lead to availability problems
- Exploits depend on the security requirements and specifications

Firewall Policies

A few examples

Permissive

Allow by default; block some

- IRC (messaging)
- Telnet
- SNMP (routing)
- Echo

Restrictive

Block by default; allow some

- HTTP
- POP3
- SMTP (mail)
- SSH

Rule Order

- A firewall policy is a collection of rules
- Packets can contain several headers (IPSec)
- Systems can be quite heterogeneous

Rule Order

- A firewall policy is a collection of rules
- Packets can contain several headers (IPSec)
- Systems can be quite heterogeneous

When setting a policy, you have to know in which order rules (and headers) are analysed and evaluated.

- Two main options for ordering rules:
  - Apply the *first matching entry* in the list of rules
  - Apply the *entry with the best match* for the packet

A Typical Firewall Ruleset

- Allow from internal network to Internet
  - HTTP, FTP, HTTPS, SSH, DNS
- Allow reply packets
- Allow from anywhere to Mail server
  - TCP port 25 (SMTP) only
- Allow from Mail server to Internet
  - SMTP, DNS
- Allow from inside to Mail server
  - SMTP, POP3
- Block everything else

Packet Filter Rules

Rule Set A

action	ourhost	port	theirhost	port	comment
block	*	*	CARLOS	*	We don't trust these people
allow	{our hosts}	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	ourhost	port	theirhost	port	flags	comment
allow	{our hosts}	*	*	25		Our packets to their SMTP port
allow	*	25	*	*	ACK	Their replies

Rule Set E

action	ourhost	port	theirhost	port	flags	comment
allow	{our hosts}	*	*	*		Our outgoing calls
allow	*	25	*	*	ACK	Replies to our calls
allow	*	*	*	>1024		Traffic to a specific domain

Network Security - Week 8

Manuel E. Correia

DCC/FCUP

2025