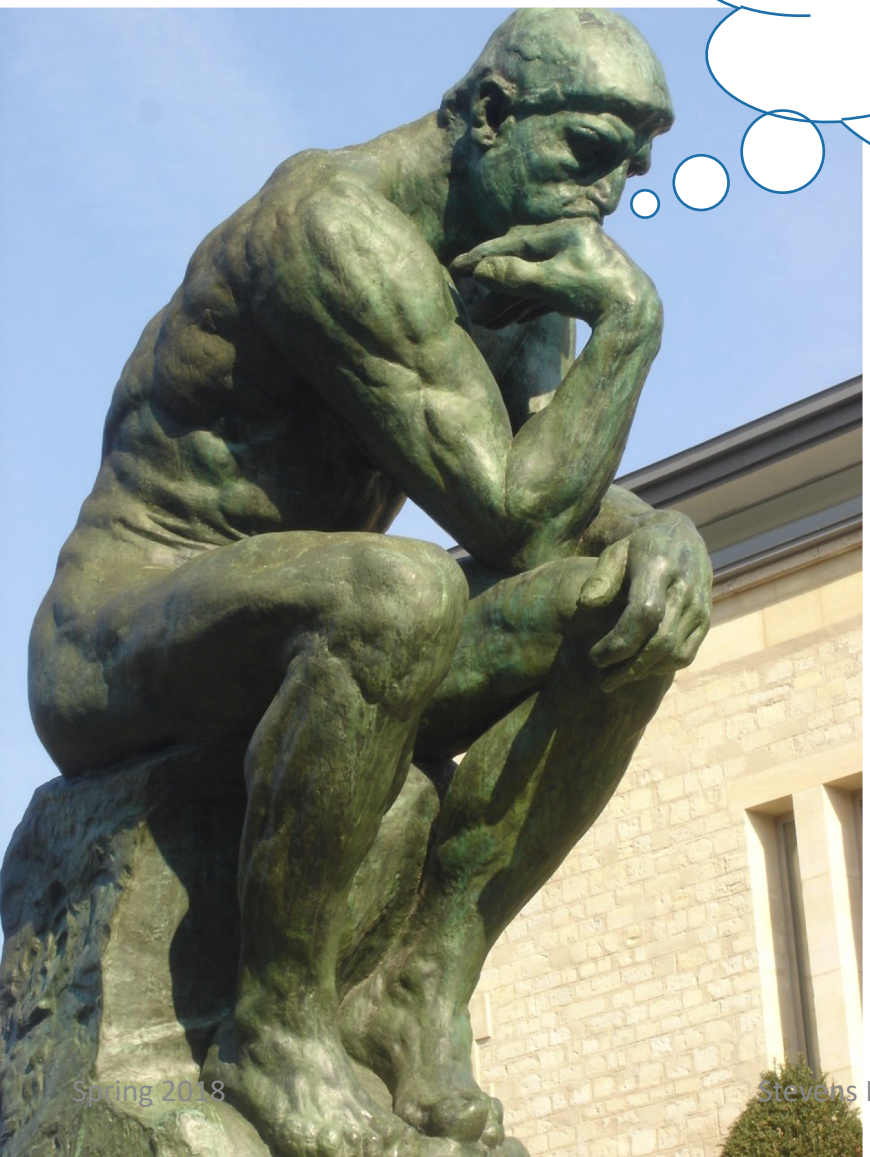
Network Security

CS-576 Systems Security

Instructor: Georgios Portokalidis

Spring 2018



I thought this was systems security

Systems vs Networks Security

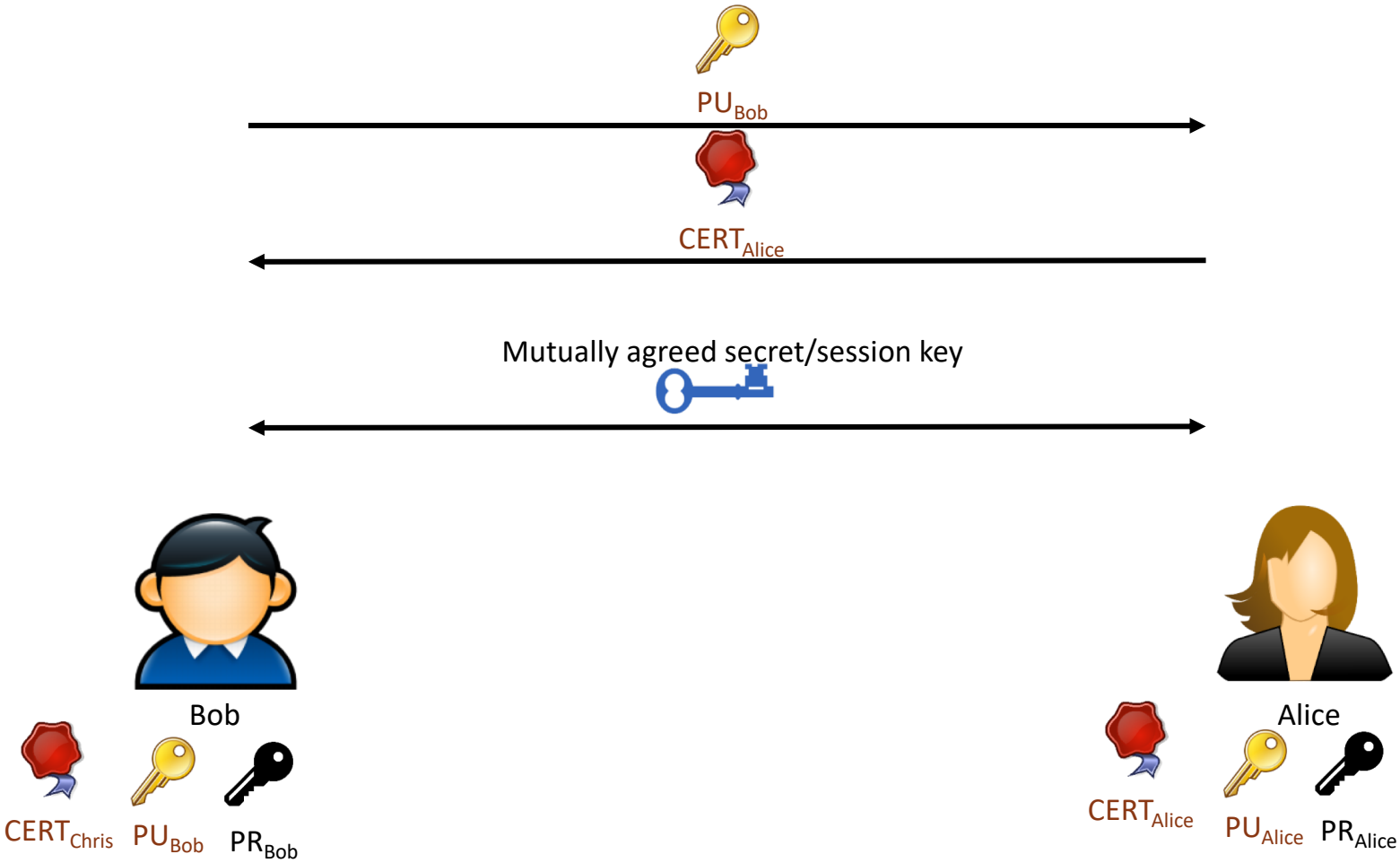
Not always a clear distinction

- Systems are interconnected
- Networks connect systems

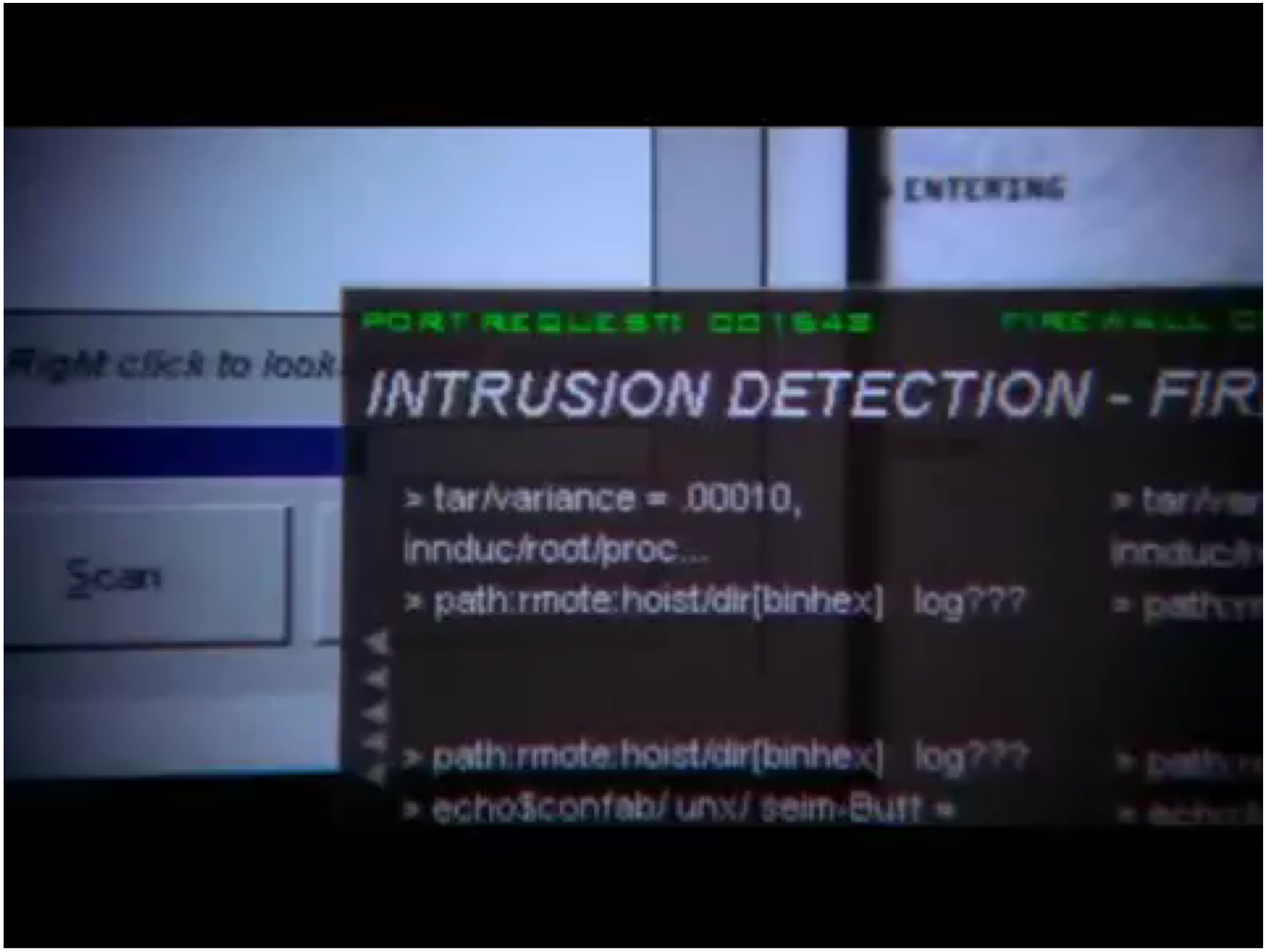
Threats usually get delivered over the network

Complex interactions between (distributed) systems and the network (protocols, infrastructure, etc.)

We Already Talked about Network Security - TLS/HTTPS

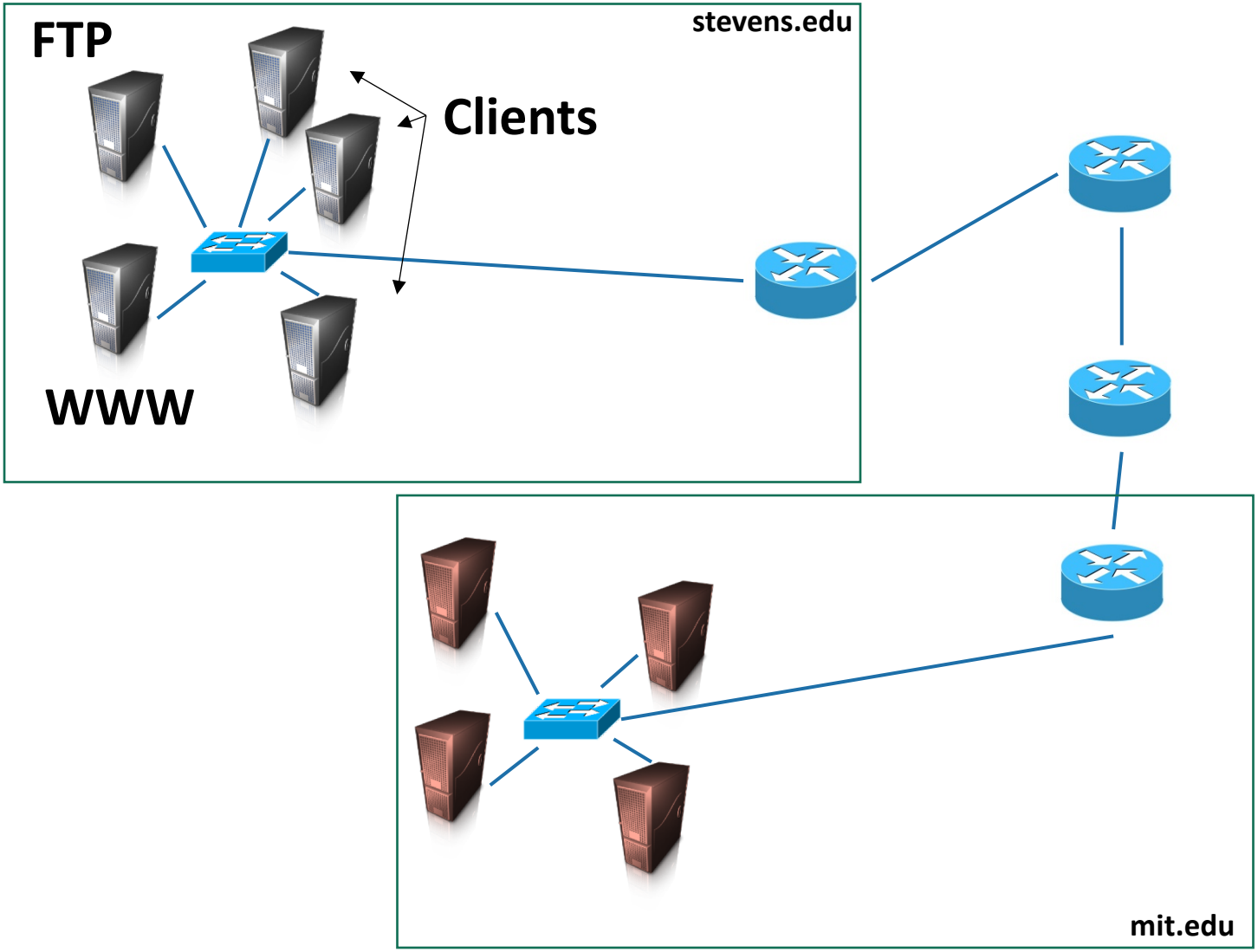


Firewalls

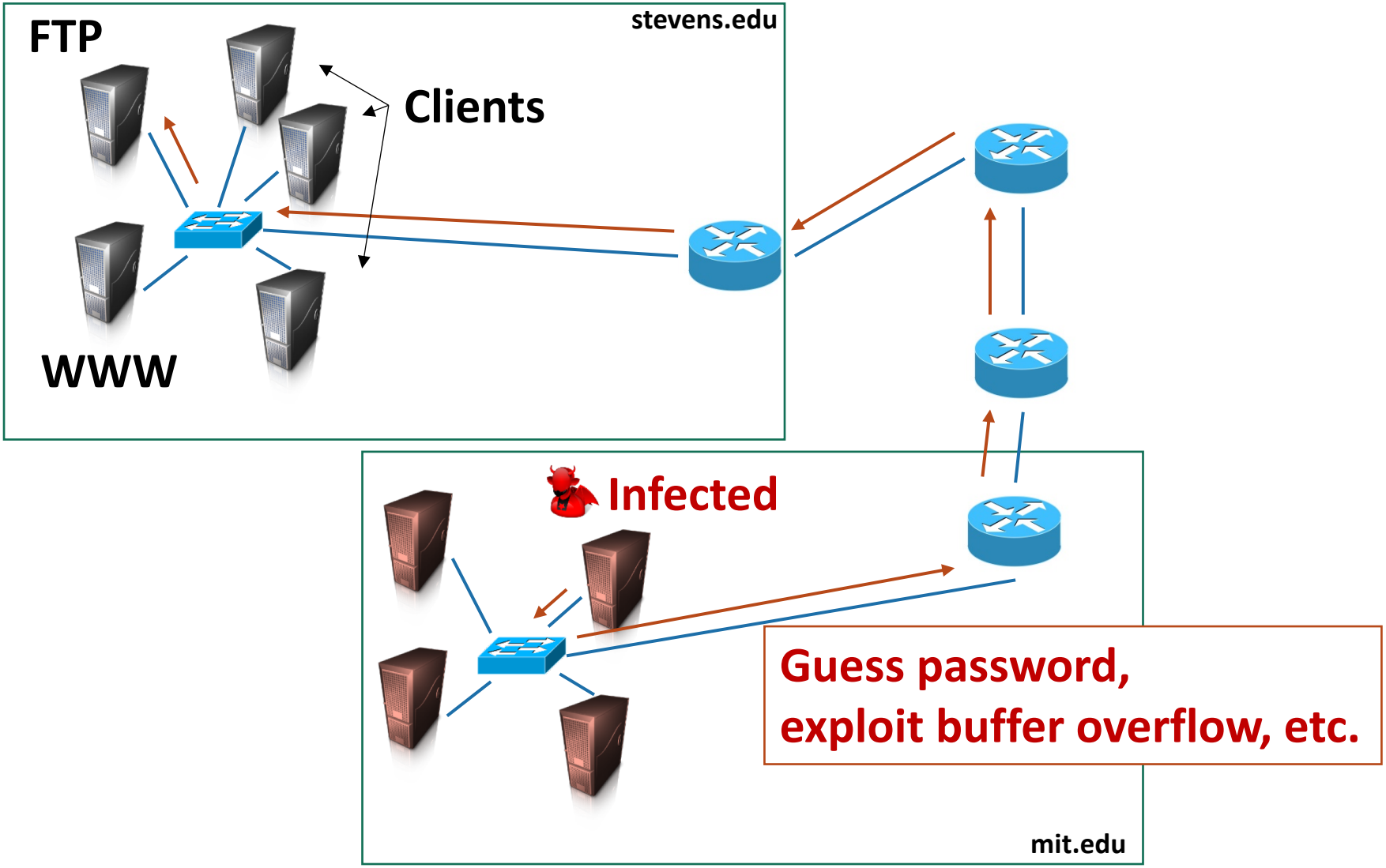

ENTERING
PORT REQUEST: 80 | 643
INTRUSION DETECTION - FIR
> tar/variance = .00010,
innduc/root/proc...
> path:rmote:hoist/dir[binhex] log???
> path:rmote:hoist/dir[binhex] log???
Scan

Firewalls Filter Traffic According to a Policy

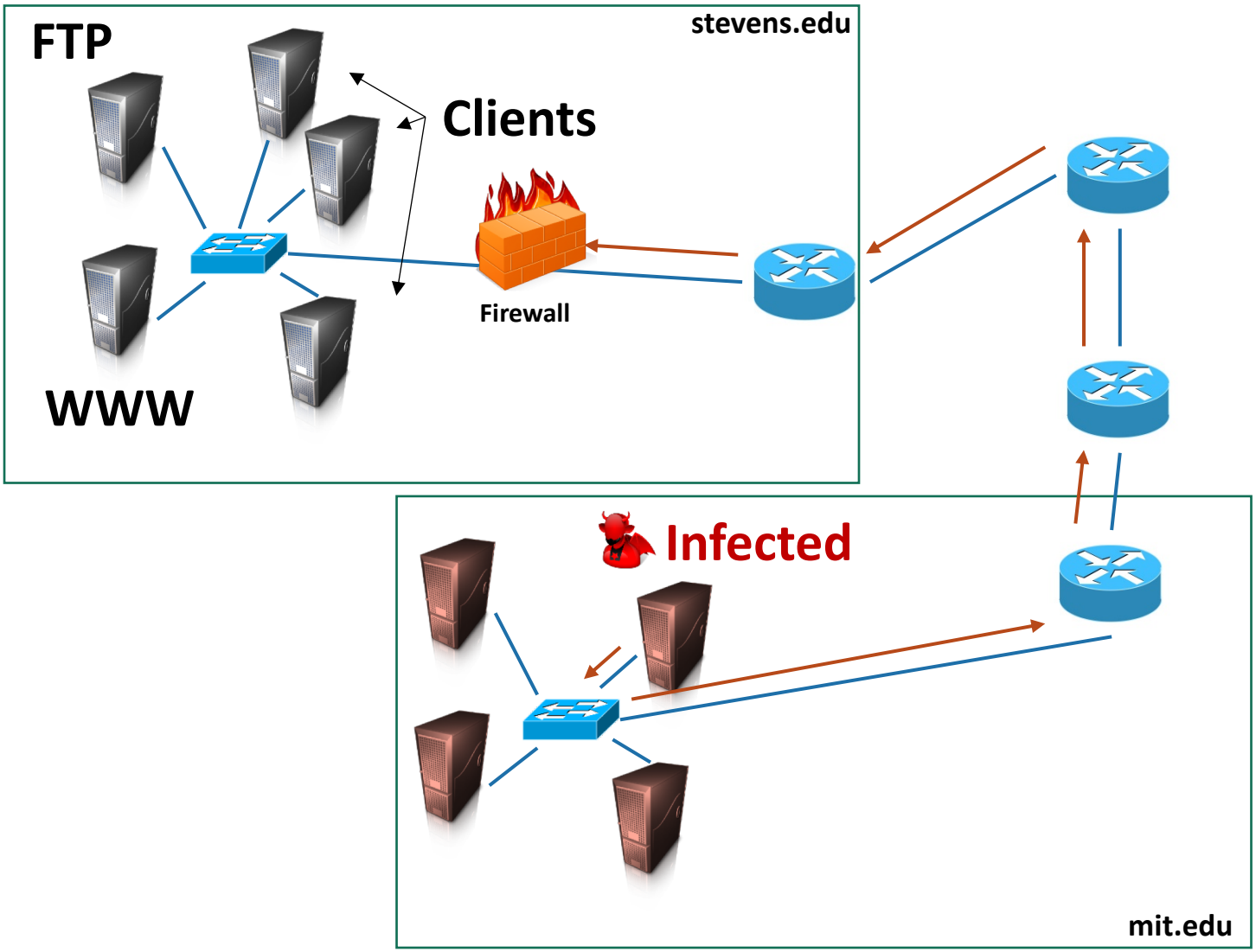
Before the Firewall



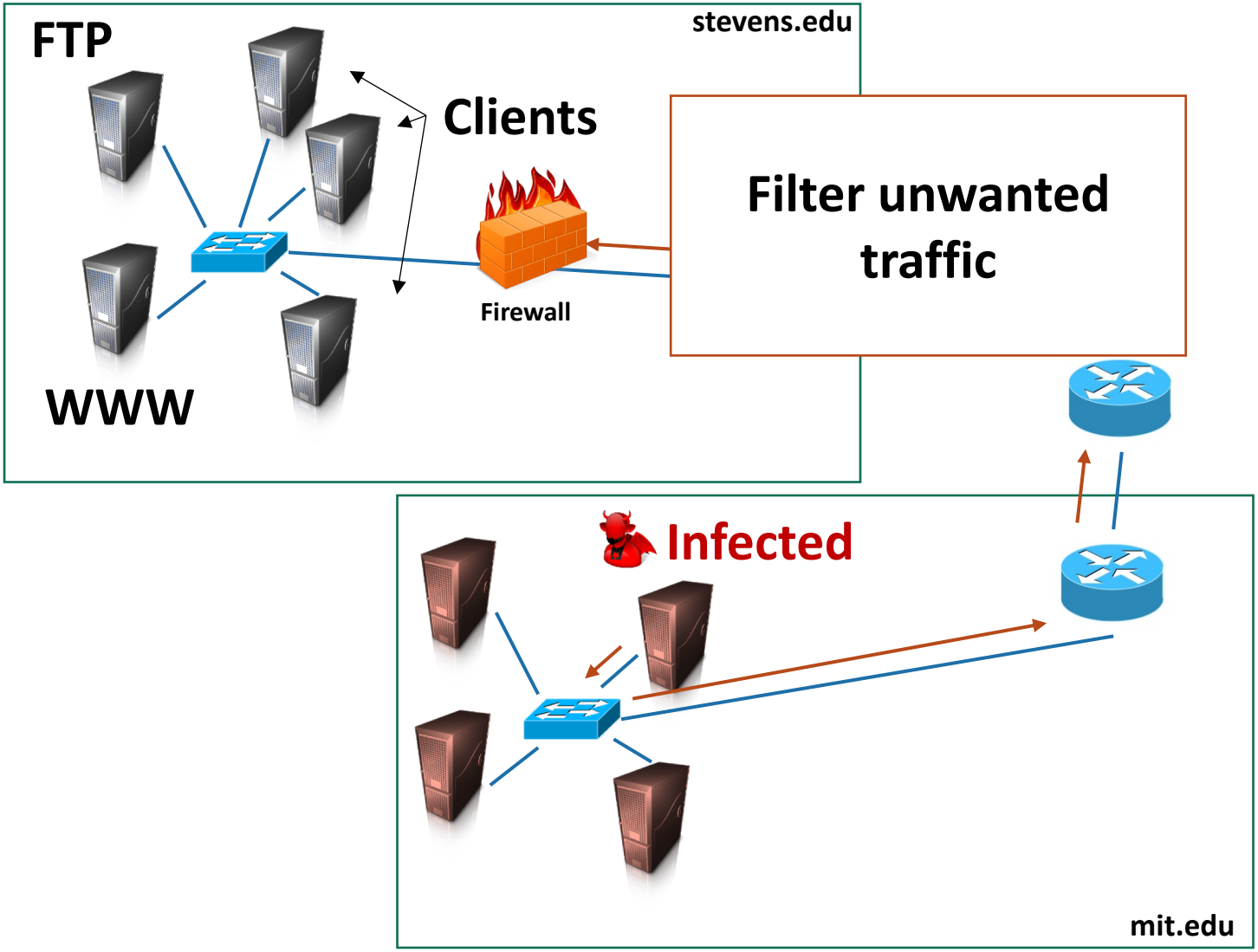
Before the Firewall



With a Firewall



With a Firewall



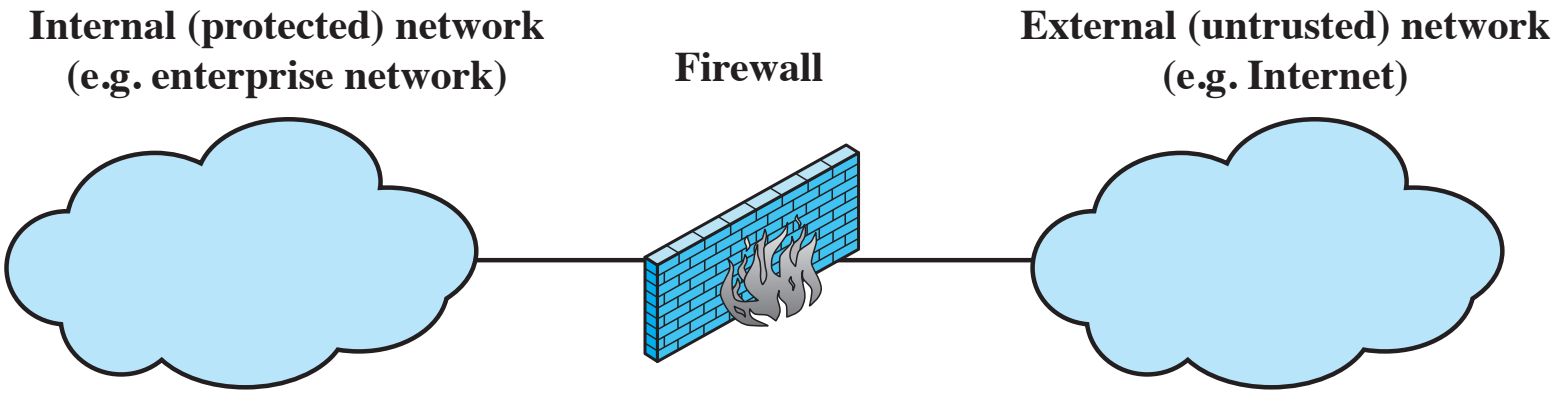
General Firewall Model

Deploy on the periphery

- A network choke point must exist

Separate the

- Safe from the unsafe
- Internal from the external



(a) General model

Why Use a Firewall

Protect services that are not supposed to be public

- They may contain vulnerabilities
- They may provide unauthorized access to data
  - Still a bad idea to rely on a firewall
- Users may start arbitrary/insecure services

Hide network topology (obstruct reconnaissance)

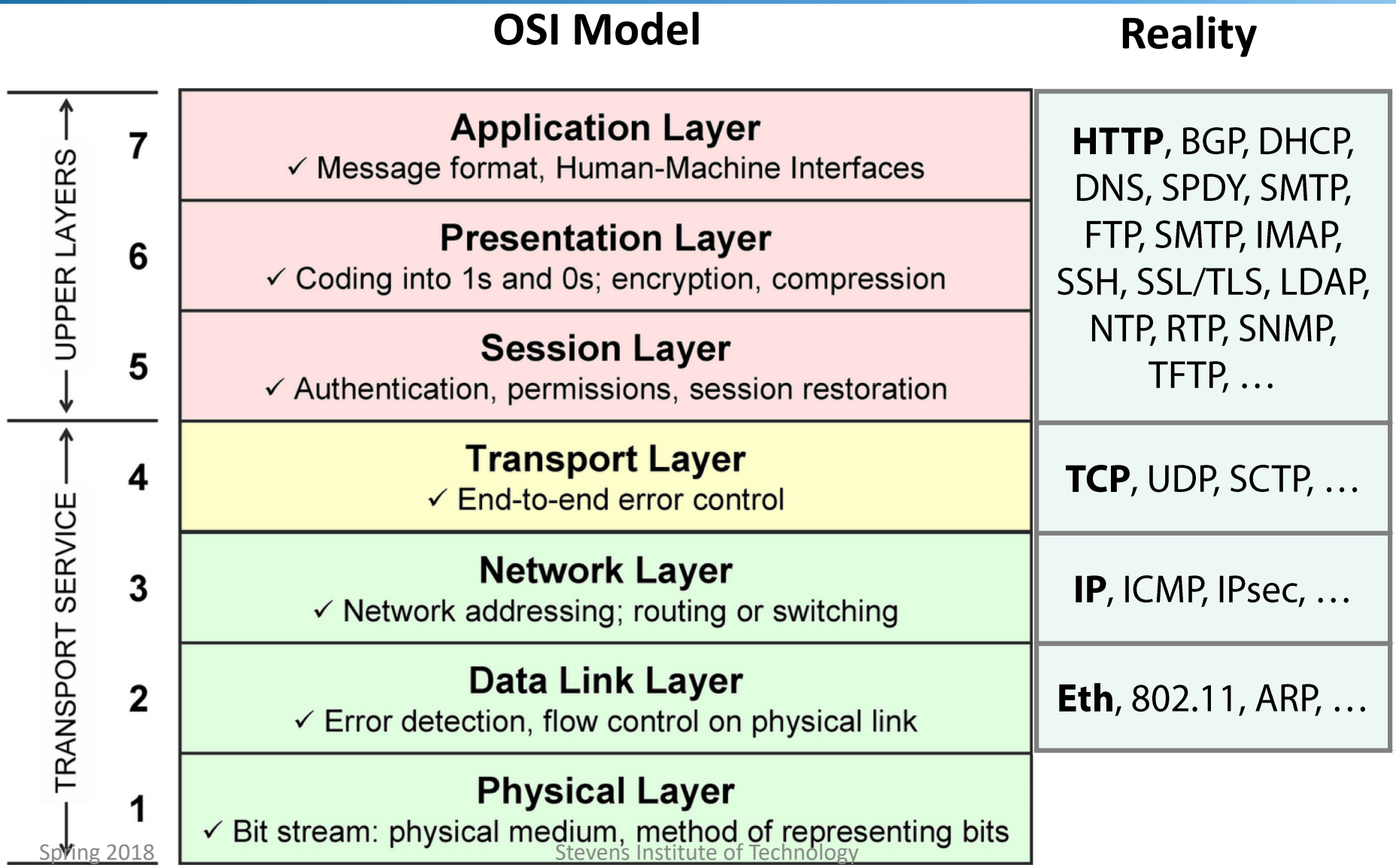
Network intelligence

- Log interesting events

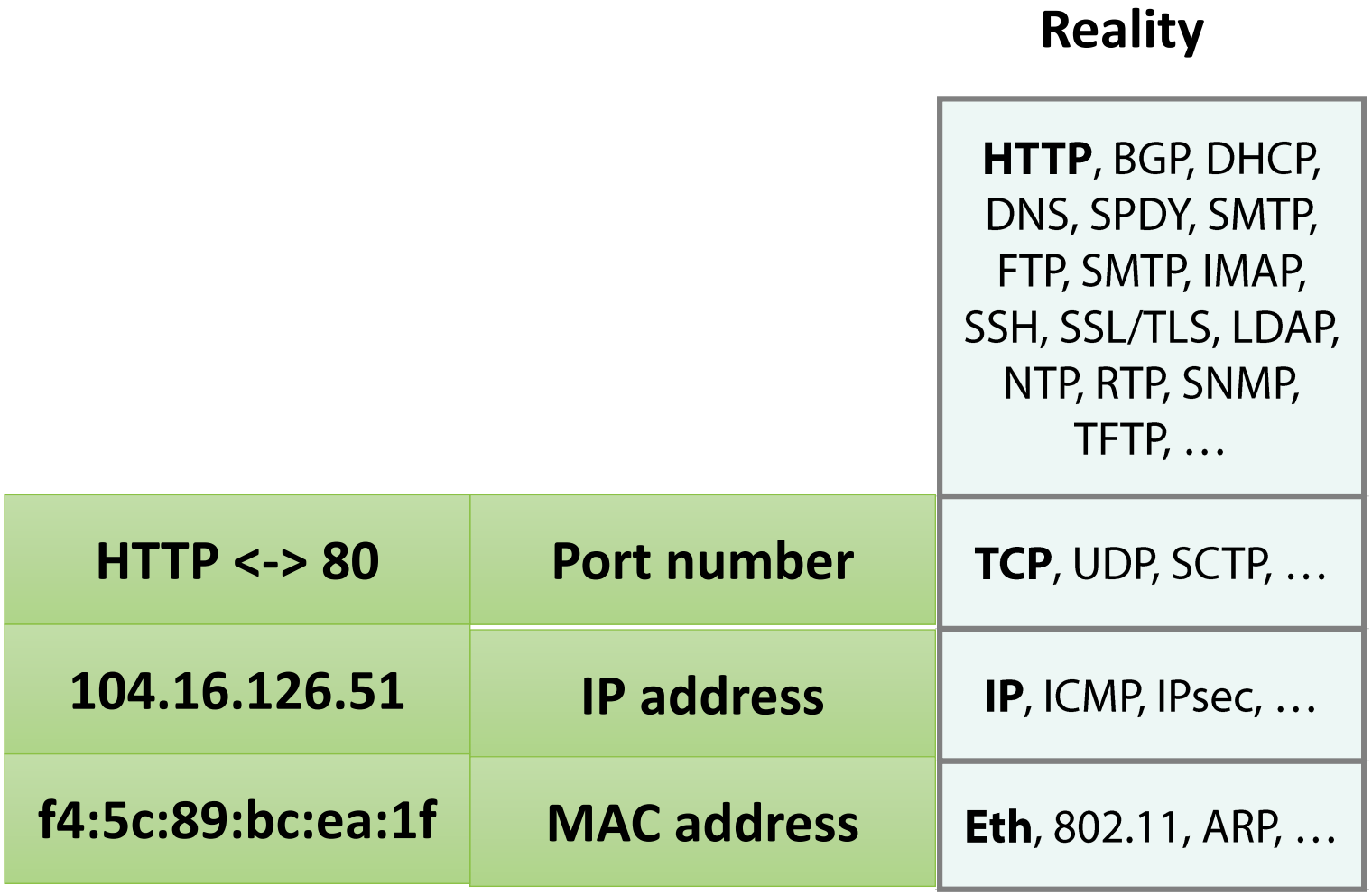
Just block unwanted traffic

How do Firewalls Work?

Revisiting Network Layers



Routing in Different Layers



Packet Filtering Firewalls

Most firewalls filter packets in the network and transport layer

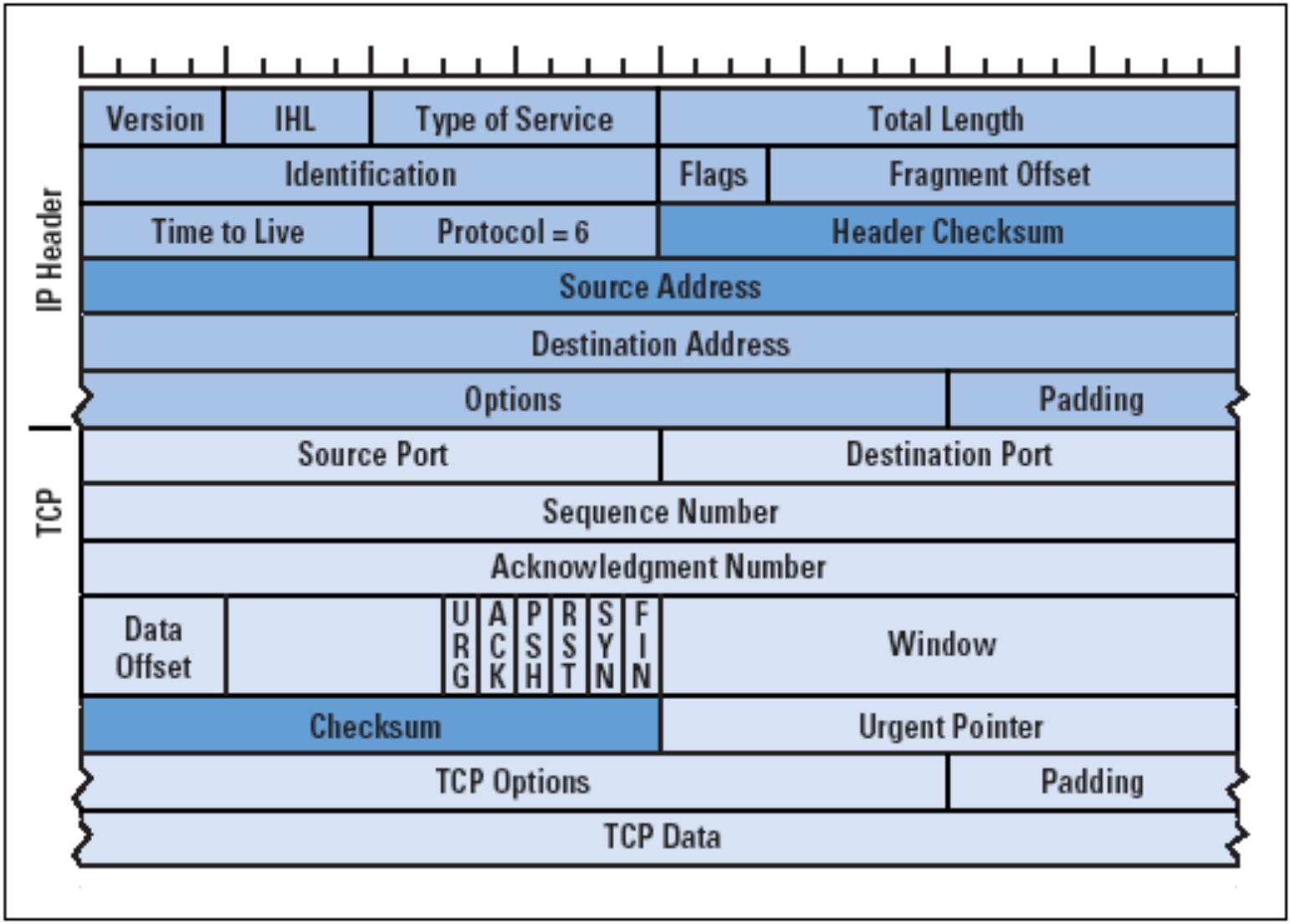
- For example, IP, TCP, UDP, ICMP, SNMP

Filters are based on rules that match protocol fields

- Multiple conditions on fields can be combined using Boolean operations

Transport Layer ✓ End-to-end error control	**TCP**, UDP, SCTP, …
Network Layer ✓ Network addressing; routing or switching	**IP**, ICMP, IPsec, …

TCP/IP Protocol Fields



Stateless Packet Filtering

Stateless by default

- They filter packets individually

Rules are applied sequentially to individual incoming and outgoing packets

Each rule is associated with a policy or action (what to do with matching packets)

The default policy is applied to non-matching packets

Commonly Encountered Actions

Accept – Allow the connection.

Drop – Drop the connection, act like it never happened. This is best if you don't want the source to realize your system exists.

Reject – Don't allow the connection, but send back an error. This is best if you don't want a particular source to connect to your system, but you want them to know that your firewall blocked them.

Log – Log the packet for later inspection

Common Default Policies

Discard - prohibit unless expressly permitted

- More conservative, controlled, visible to users

Forward - permit unless expressly prohibited

- Easier to manage and use but less secure

Packet-filtering Examples

1. Inbound mail from an external source is allowed (port 25 is for SMTP incoming).
2. This rule is intended to allow a response to an inbound SMTP connection.
3. Outbound mail to an external source is allowed.
4. This rule is intended to allow a response to an inbound SMTP connection.
5. This is an explicit statement of the default policy. All rule sets include this rule implicitly as the last rule.

Rule	Direction	Src address	Dest addresss	Protocol	Dest port	Action
1	In	External	Internal	TCP	25	Permit
2	Out	Internal	External	TCP	>1023	Permit
3	Out	Internal	External	TCP	25	Permit
4	In	External	Internal	TCP	>1023	Permit
5	Either	Any	Any	Any	Any	Deny

Limitations of Stateless Firewalls

Advantages

Simplicity

Typically transparent to users

Very fast

Weaknesses

Cannot handle dynamically negotiated ports

IP fragmentation cannot be handled

Rule sets can get too complex to understand

Very strict

Stateful Filtering

Also keeps track of protocol state

- For example, keeps track of TCP sequence numbers to prevent attacks that depend on the sequence number
- Or filter packets that do not belong to an established connection

Most common firewall type

More flexible policies

- Can differentiate connections initiated internally

Requires more storage to save connection state

- The more connection the more storage used

Common Configuration

Block incoming connections, but allow outgoing

- Incoming packets are permitted, if they belong to established connections
- Other incoming packets are blocked

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

Application Firewalls

Operate on the application layer

Similar to proxies

HTTP, BGP, DHCP, DNS, SPDY, SMTP, FTP, SMTP, IMAP, SSH, SSL/TLS, LDAP, NTP, RTP, SNMP, TFTP, …

Filters/transforms potentially harmful application messages while proxying traffic

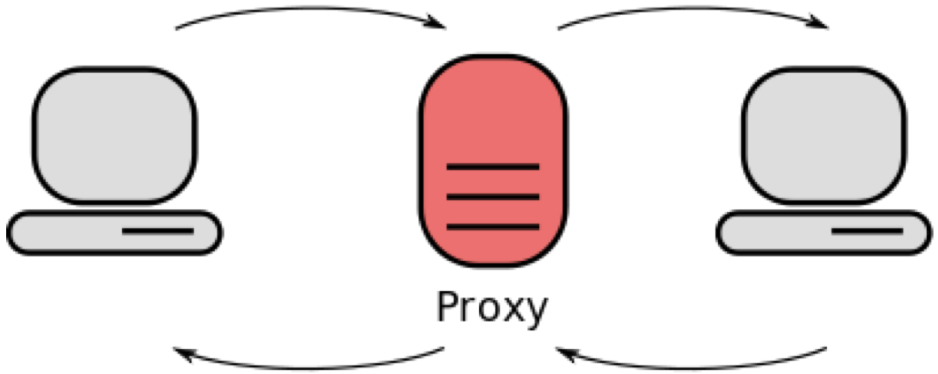
- For example, protect web applications from SQL injection, XSS, etc. (aka Web application firewalls)

Proxies

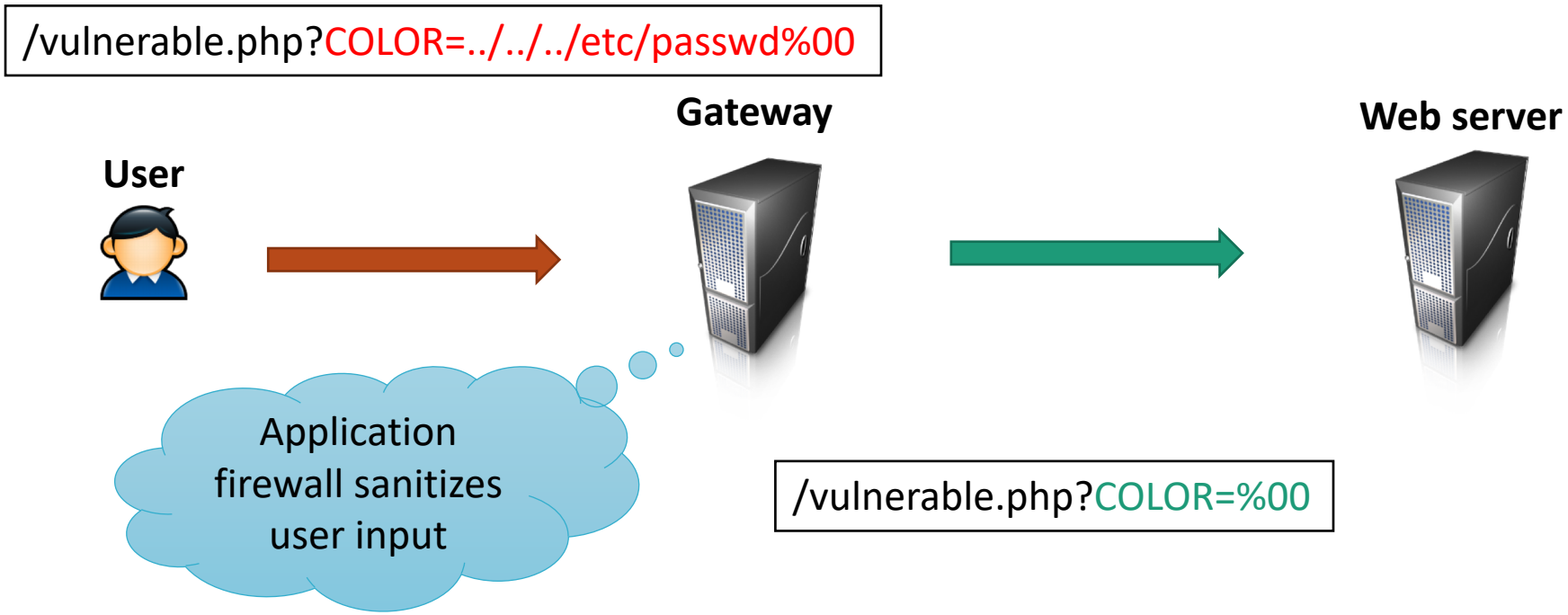
Essentially intermediate stepping stones

Many non-security uses as well

- For example, caching



Example: Web Application Gateway



Circuit-level Proxy Firewalls

A circuit-level gateway or proxy firewall hides the internal network by proxying TCP streams

Essentially a **transparent**, **transport-layer** proxy

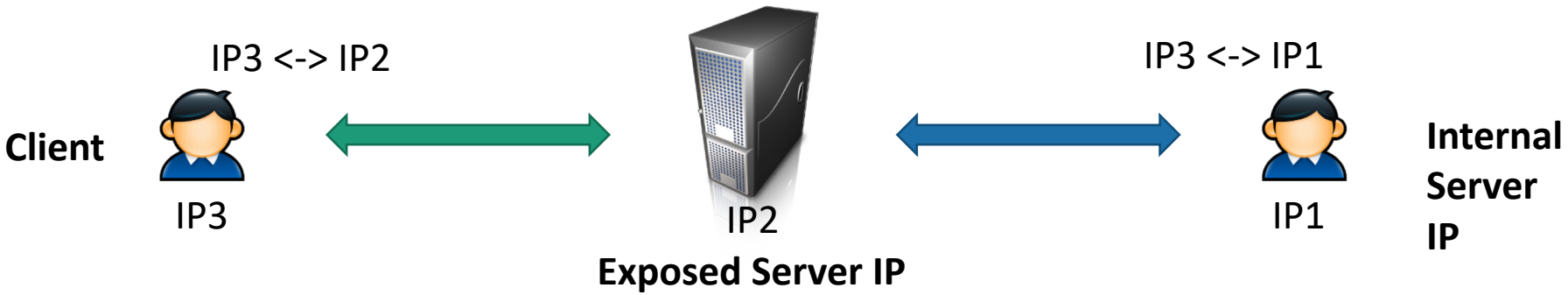
Circuit-level Gateway

Sets up two TCP connections, one between itself and a TCP user on an inner host and one on an outside host

- Hides inner network details
- Protects client protocol stack

Relays TCP segments from one connection to the other without examining contents

- Security function consists of determining which connections will be allowed



Application-level Gateway (ALG)

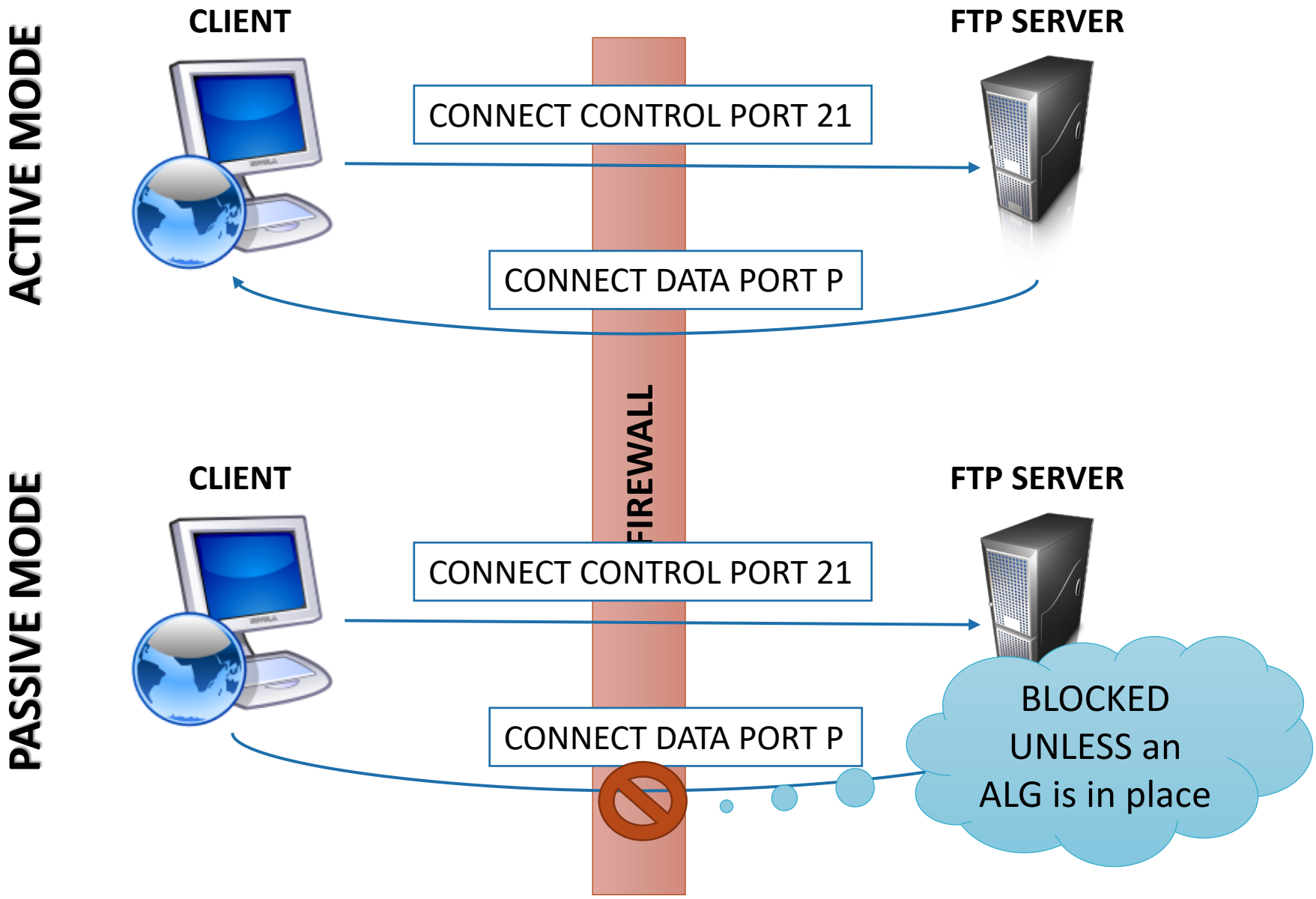
Packets are inspected and application protocols processed

Can implement parts of an application

- For example, authenticate users before forwarding

Can customize the firewall rules based on service needs

- For example, FTP



Allow FTP SERVER -> TCP DESTPORT 80 ACCEPT

Host-based Firewalls

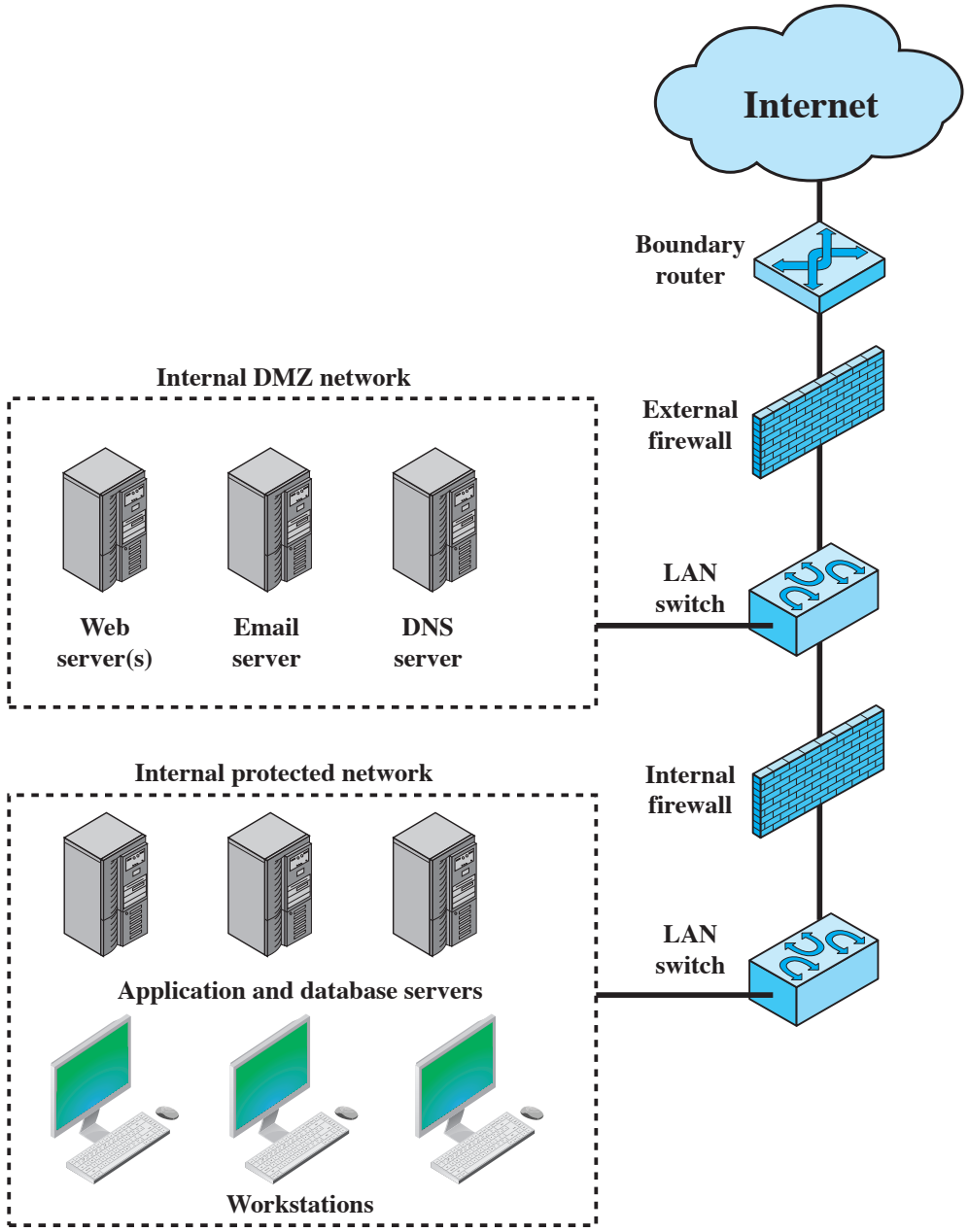
Used to secure an individual host

Available in operating systems or can be provided as an add-on package

Advantages

- Filtering rules can be tailored to the host environment
- Protection is provided independent of topology
- Provides an additional layer of protection
- Can ask the user

Internal Firewalls



Virtual Private Networks (VPN)

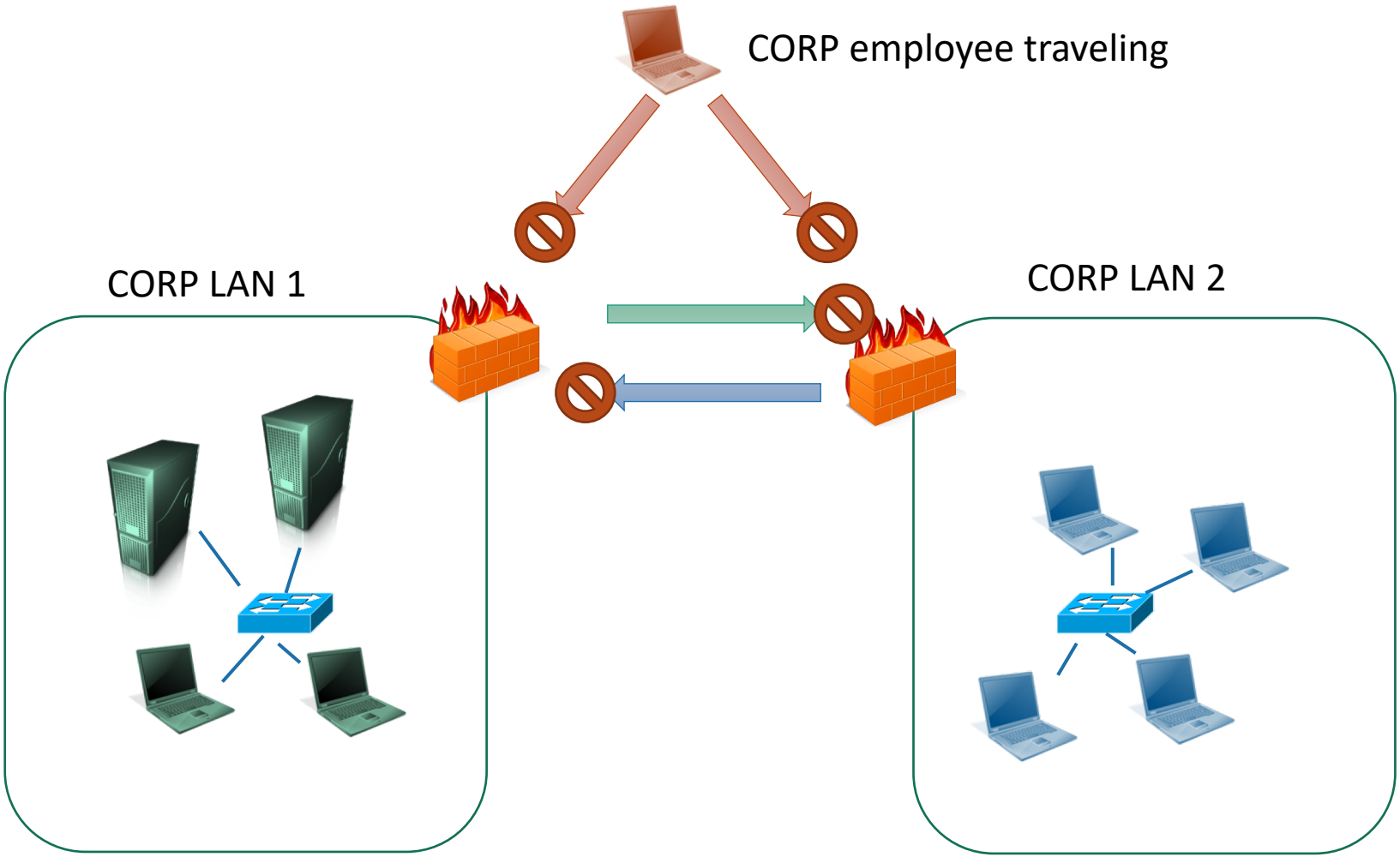
Users may not always be behind the firewall but need to access the internal network

Users of the same organization may be located to different local area networks (LANs)

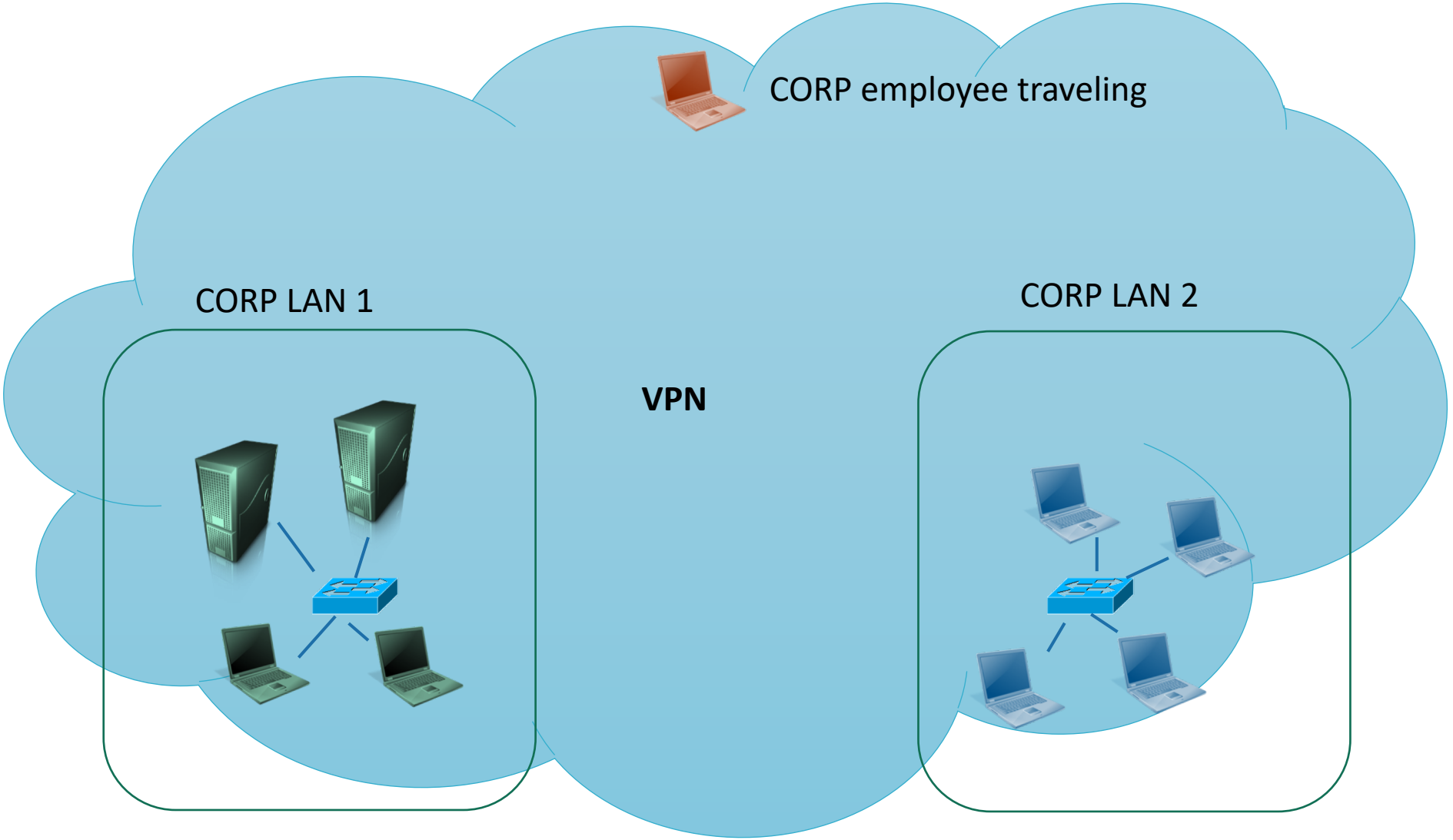
VPNs create a secure bridge between LANs or a LAN and the user

- Everyone is virtually on the same network

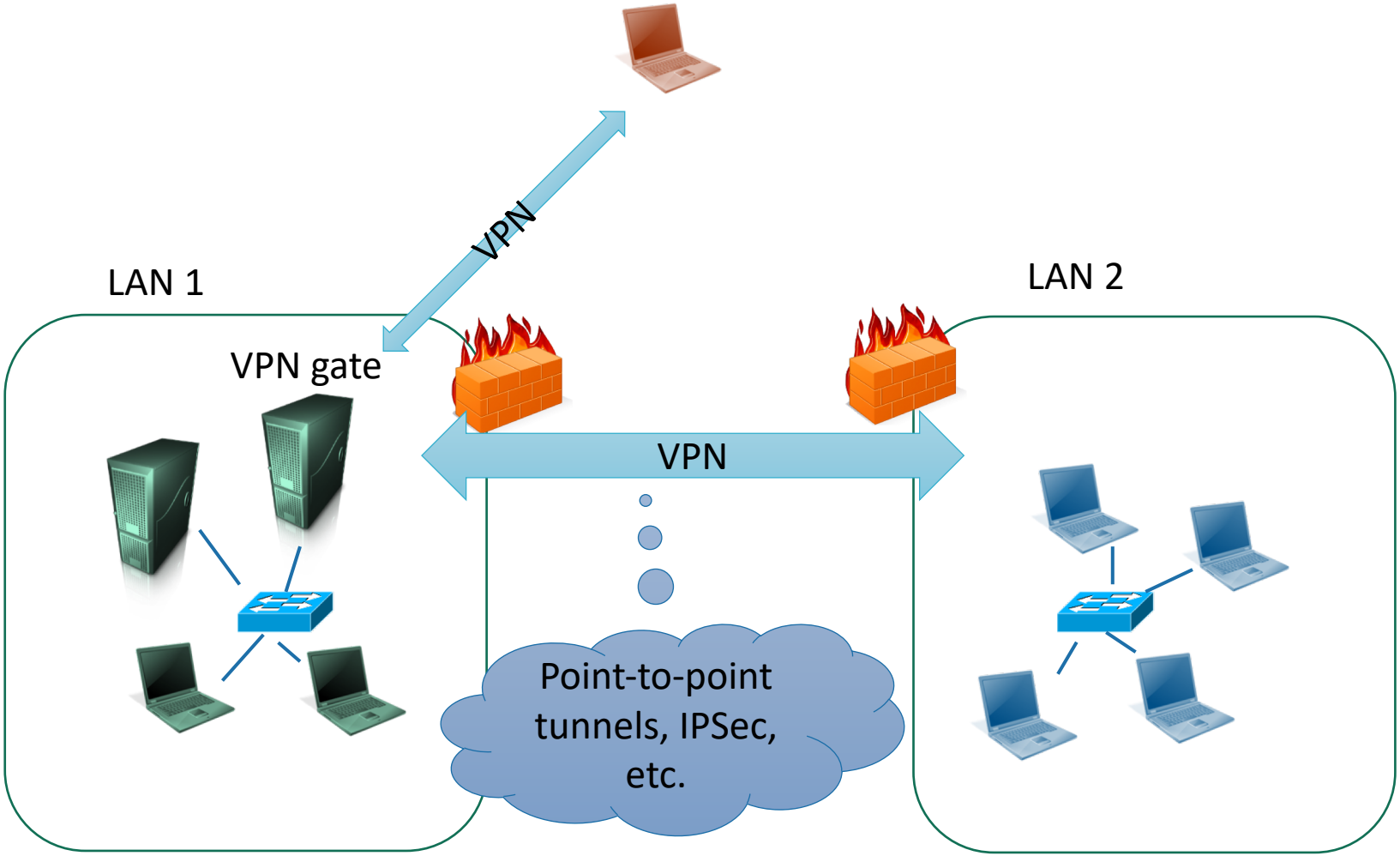
Virtual Private Networks



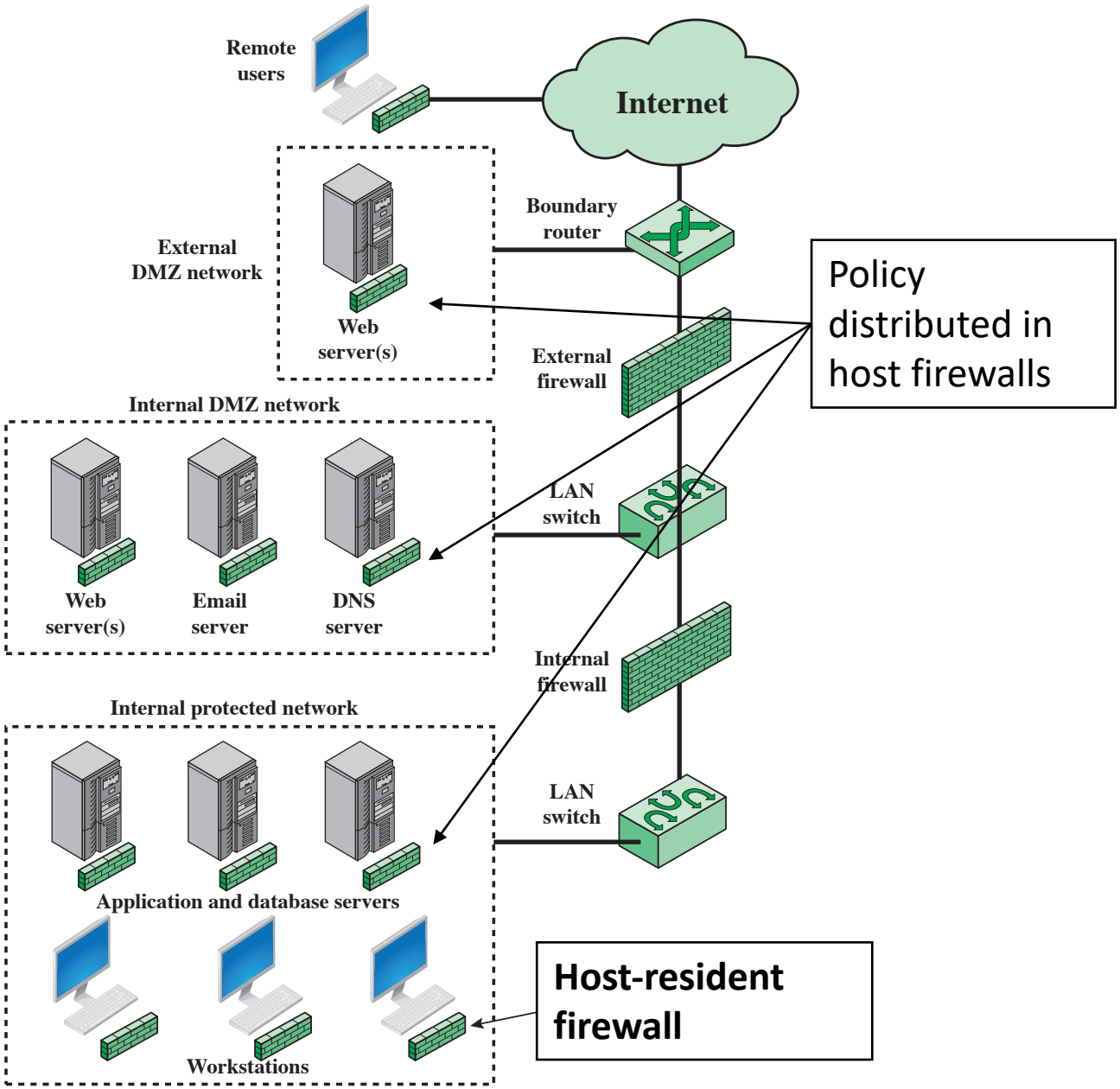
Virtual Private Networks



Virtual Private Networks

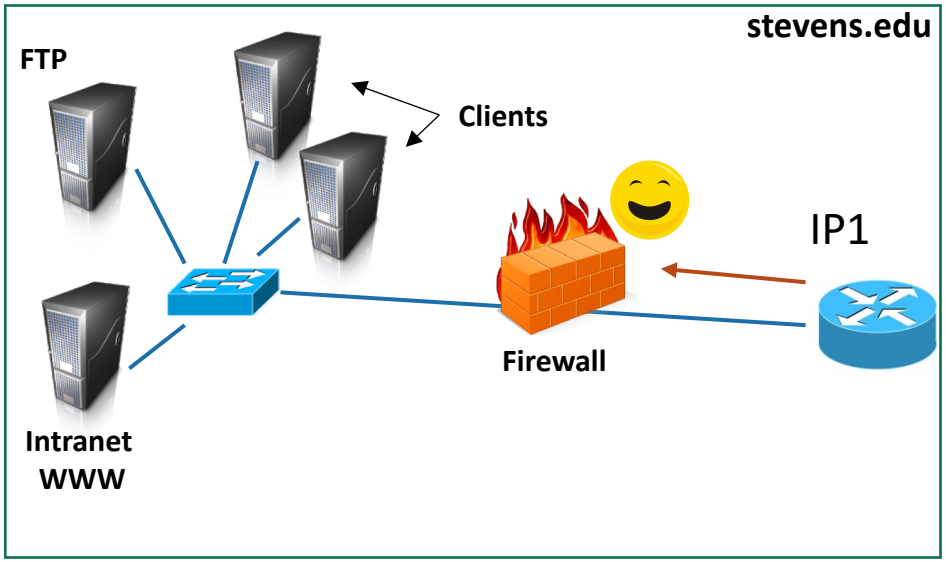


Distributed Firewalls



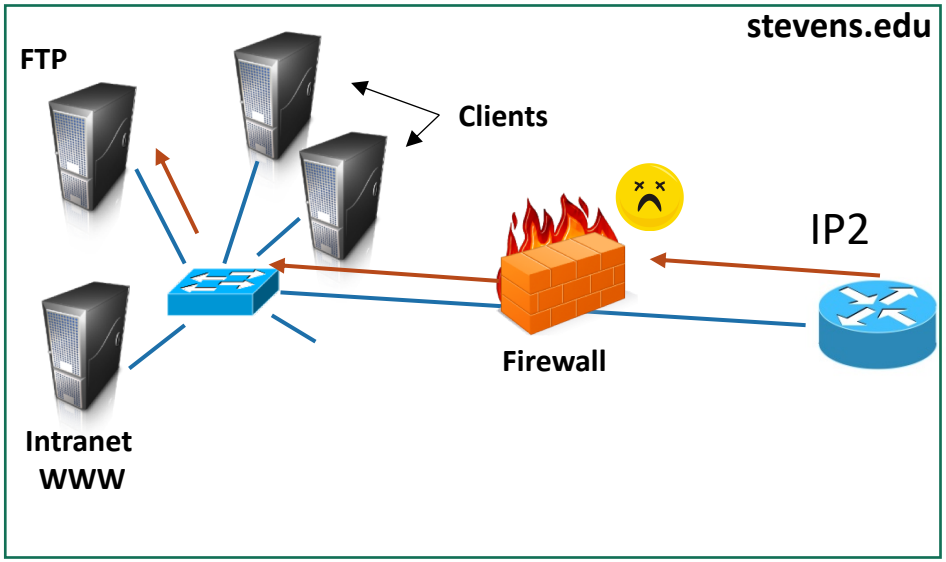
Firewall Limitations

Improper configuration could allow bypassing the firewall



Firewall Limitations

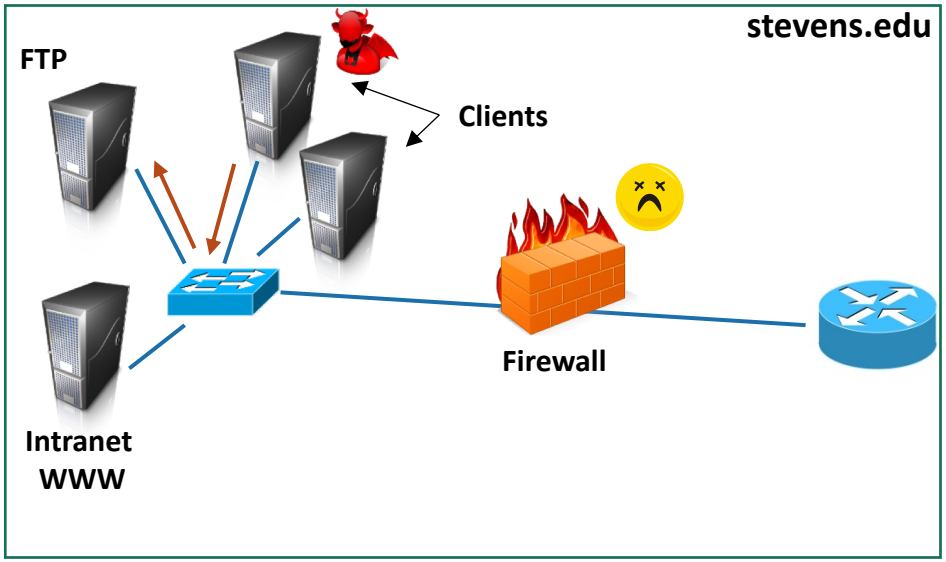
Improper configuration could allow bypassing the firewall



Firewall Limitations

Improper configuration could allow bypassing the firewall

Cannot protect from internal threats

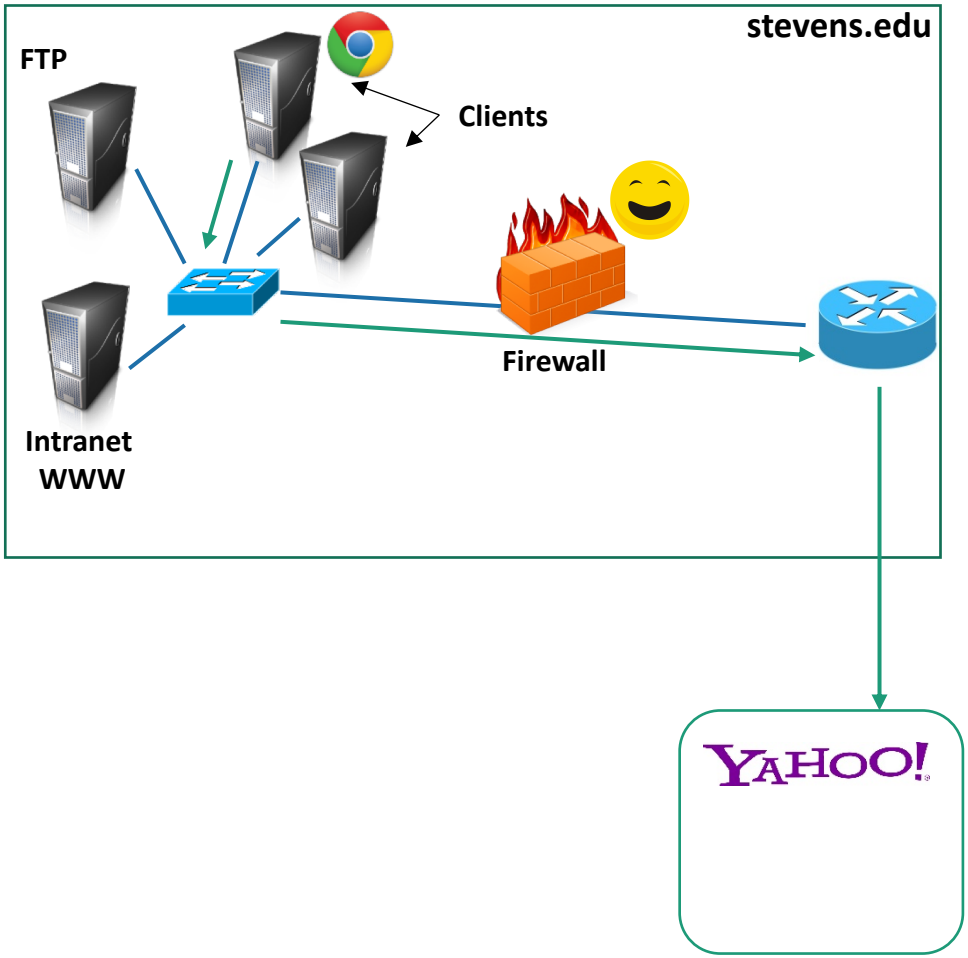


Firewall Limitations

Improper configuration could allow bypassing the firewall

Cannot protect from internal threats

Cannot protect from client-side attacks

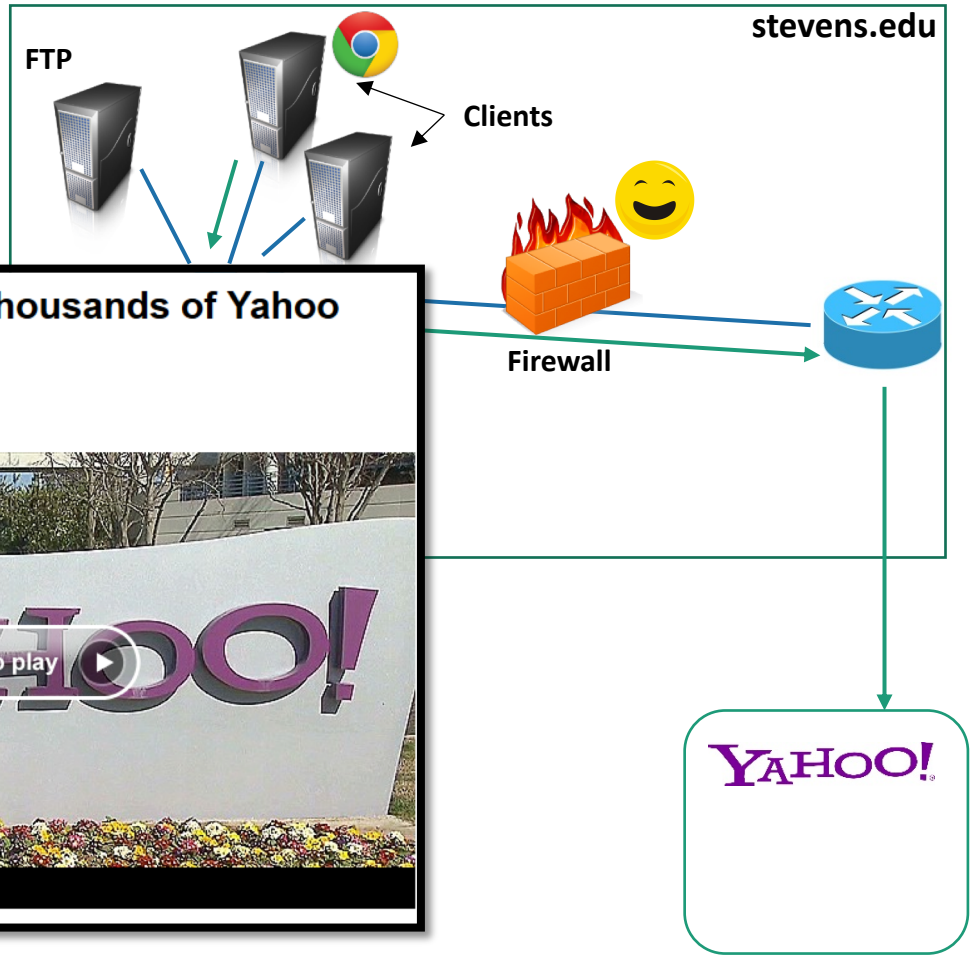


Firewall Limitations

Improper configuration could allow bypassing the firewall

Cannot protect f
internal threats

Cannot protect side attacks

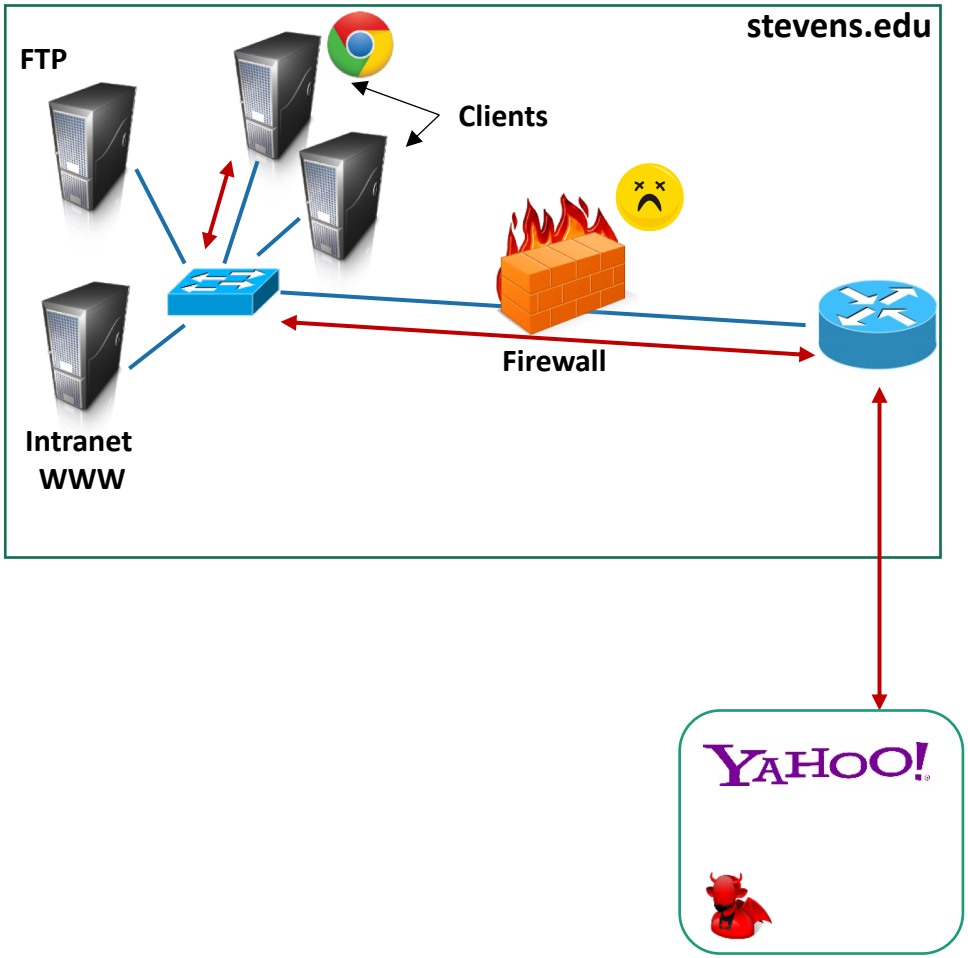


Firewall Limitations

Improper configuration could allow bypassing the firewall

Cannot protect from internal threats

Cannot protect from client-side attacks



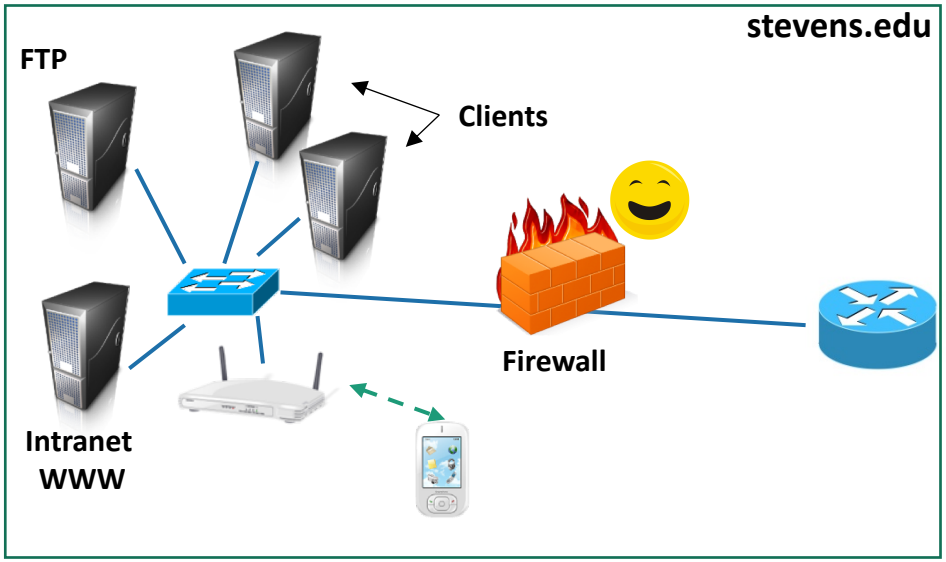
Firewall Limitations

Improper configuration could allow bypassing the firewall

Cannot protect from internal threats

Cannot protect from client-side attacks

Bring-your-own-device type of problems



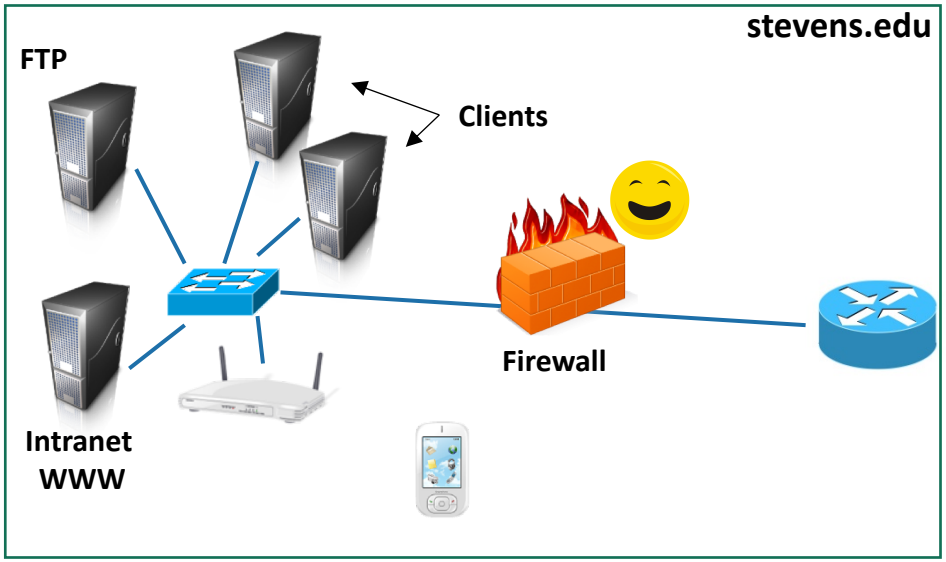
Firewall Limitations

Improper configuration could allow bypassing the firewall

Cannot protect from internal threats

Cannot protect from client-side attacks

Bring-your-own-device type of problems



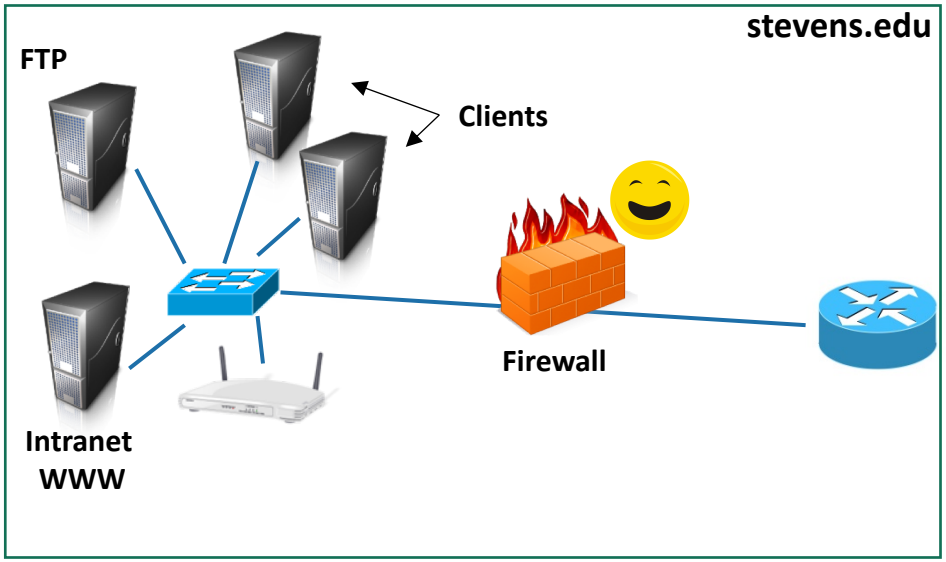
Firewall Limitations

Improper configuration could allow bypassing the firewall

Cannot protect from internal threats

Cannot protect from client-side attacks

Bring-your-own-device type of problems



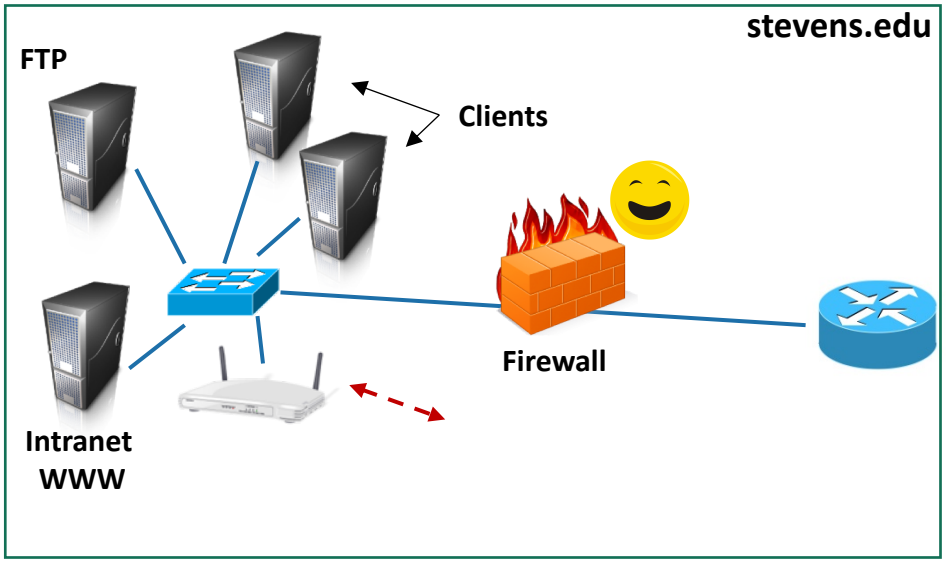
Firewall Limitations

Improper configuration could allow bypassing the firewall

Cannot protect from internal threats

Cannot protect from client-side attacks

Bring-your-own-device type of problems



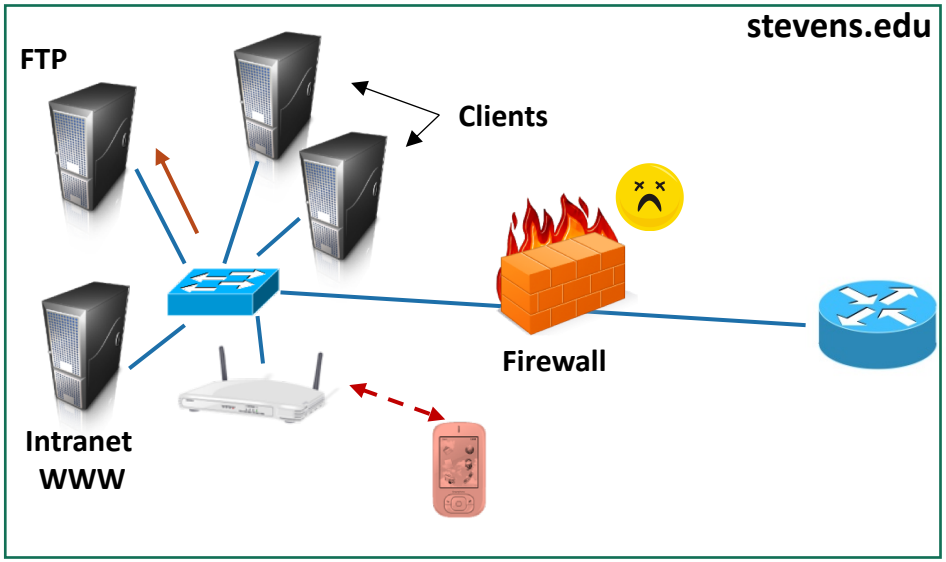
Firewall Limitations

Improper configuration could allow bypassing the firewall

Cannot protect from internal threats

Cannot protect from client-side attacks

Bring-your-own-device type of problems



Reading

Firewalls and Internet Security: Repelling the Wily Hacker

http://www.wilyhacker.com/

Walls and gates
https://www.cs.columbia.edu/~smb/papers/Walls_and_Gates.pdf

Packets found on the Internet

https://www.cs.columbia.edu/~smb/papers/packets.pdf

Intrusion Detection

Intruder Behavior

Target acquisition and information gathering

Intrusion / initial access

Privilege escalation (if required)

Malicious Activity

Maintaining access

Covering tracks

Intrusion Detection Systems

Intrusion detection systems monitor networks and hosts for malicious activities and policy violations

Intrusion detection systems (**IDS**) generate alerts and log identified events

Intrusion prevention systems (**IPS**) react by blocking the detected activity

IDS Types

Host-based IDS (HIDS)

- Monitors the characteristics of a single host for suspicious activity

Network-based IDS (NIDS)

- Monitors network traffic and analyzes network, transport, and application protocols to identify suspicious activity

Distributed or hybrid IDS (HIDS)

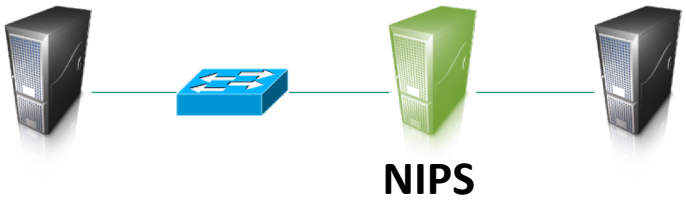
- Combines information from a number of sensors, often both host and network based, in a central analyzer that is able to better identify and respond to intrusion activity

NIDS vs NIPS

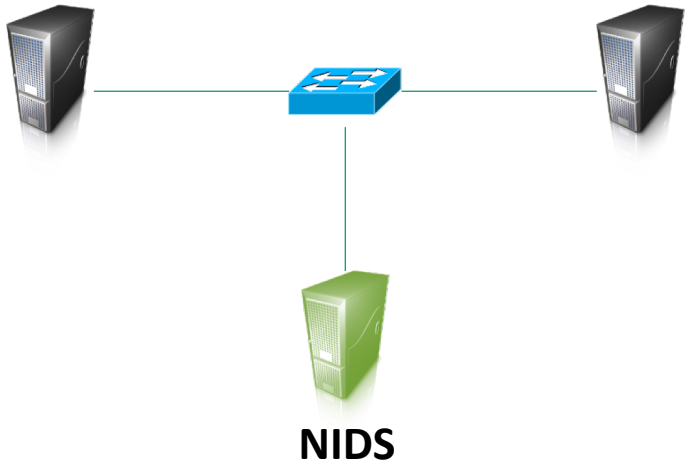
Similar designs

Deployment location is the main difference

Inline deployment to filter/block offending traffic



Parallel deployment to passively monitor traffic



IDS Components

Sensors - collect data

Analyzers - determine if intrusion has occurred

User interface - view output or control system behavior

Analysis Approaches

Anomaly detection

First establish the behavior of legitimate users

Detection: Observed behavior is analyzed to determine if it matches expected legitimate behaviors

Signature/Heuristic detection

Detection: Uses a set of known malicious data patterns or attack rules that are compared with current behavior

- Also known as misuse detection

Can only identify known attacks for which it has patterns or rules

Anomaly Detection

Statistical

- Analysis of the observed behavior using univariate, multivariate, or time-series models of observed metrics

Machine learning

- Approaches automatically determine a suitable classification model from the training data using data mining techniques

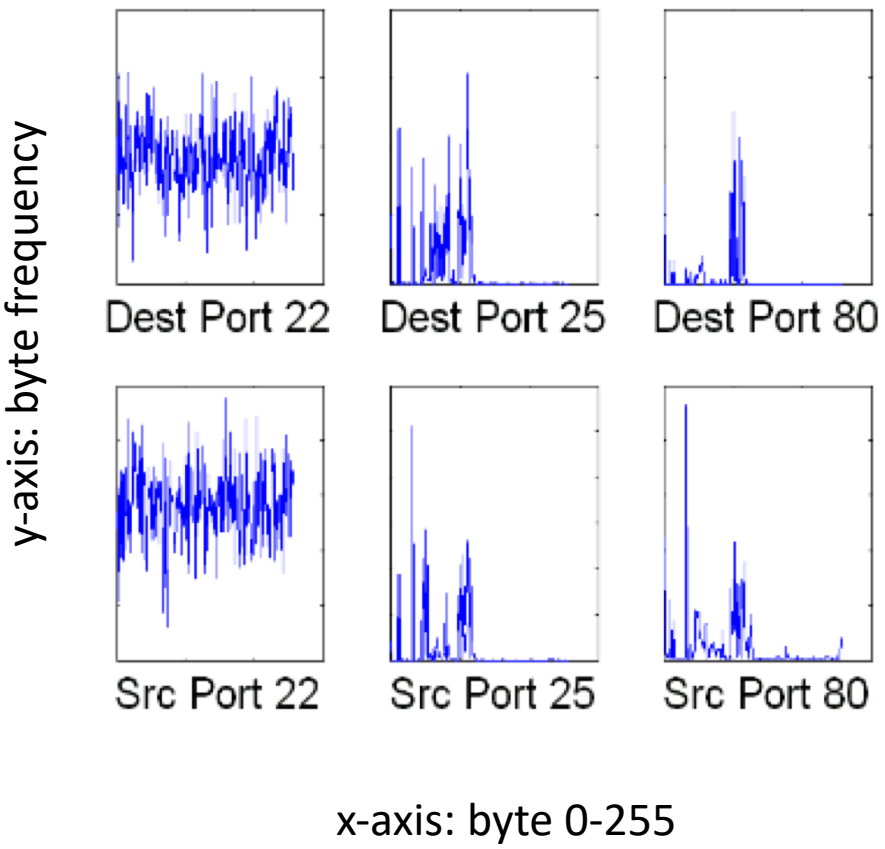
Example of Statistical AD

Byte frequency-based anomaly detection

Can the frequency of the bytes commonly used in a protocol be used to detect anomalies?

Reading: *Anomalous Payload-Based Network Intrusion Detection, RAID 2015*

http://link.springer.com/chapter/10.1007/978-3-540-30143-1_11



Example: HTTP

HTTP is predominantly a text protocol

Different request sizes may exhibit different byte frequencies

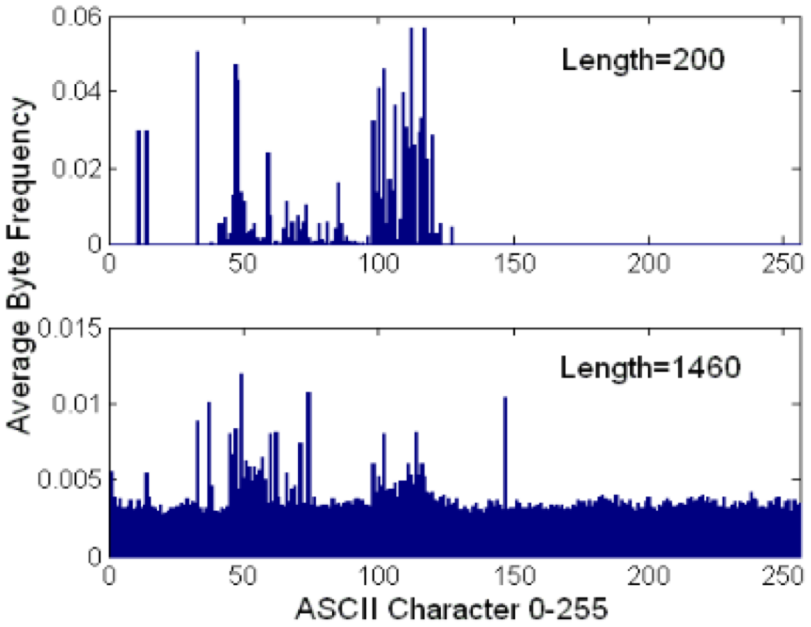


Fig. 2. Example byte distribution for different payload lengths for port 80 on the same host server

Example: HTTP

HTTP is predominantly a text protocol

Different request sizes may exhibit different byte frequencies

How can we use this information to detect anomalies?

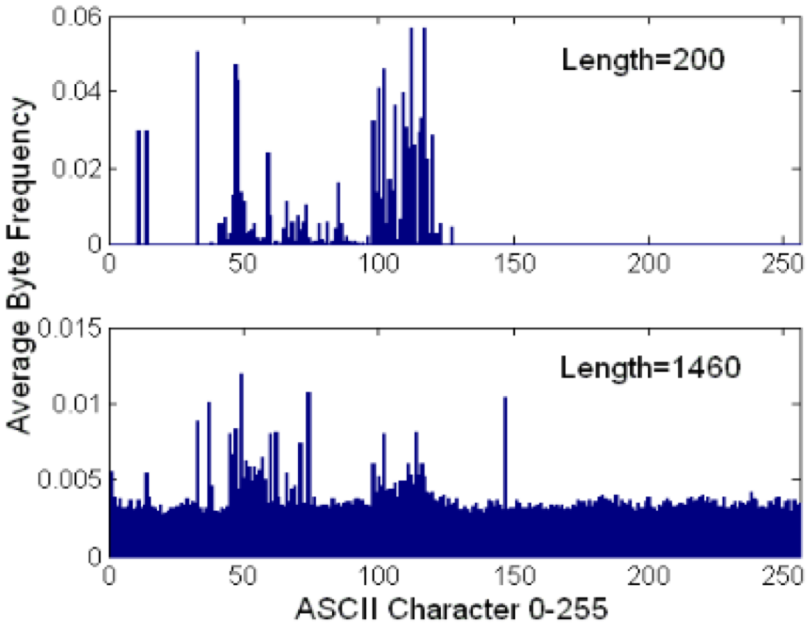


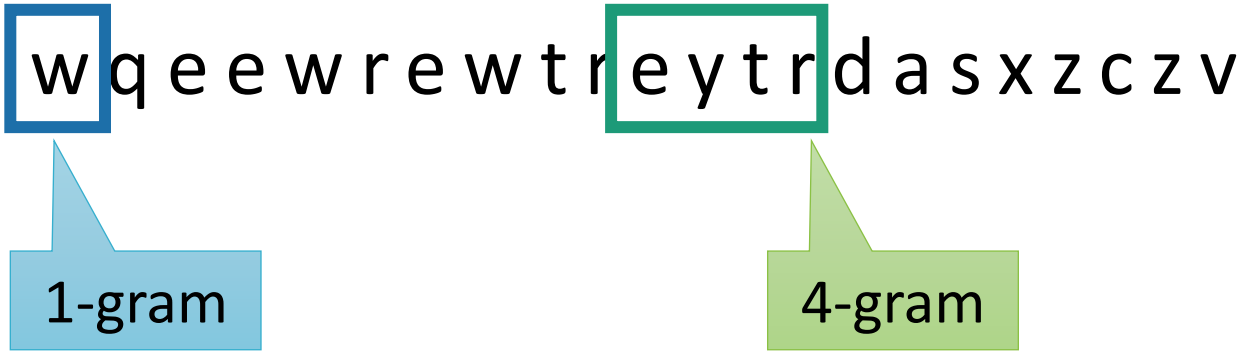
Fig. 2. Example byte distribution for different payload lengths for port 80 on the same host server

N-grams

Single byte frequency may not be a good feature of the protocol for identifying anomalies

N-grams are a contiguous sequence of n items from a given sequence of bytes

- N-gram frequency also commonly used for AD



Signature-based Detection

Relies on a large database of signatures → patterns of known malicious data

Collected data are compared with signatures

Signatures needs to be specific enough to minimize error, while still detecting a sufficiently large fraction of malicious data

Widely used in anti-virus products, network traffic scanning proxies, and in NIDS

Rule-based Intrusion Detection

Same as signature-based

A rule could be considered as more expressive

- Including multiple patterns
- Including heuristics

Widely used in network intrusion detection systems

All intrusion detection systems make mistakes

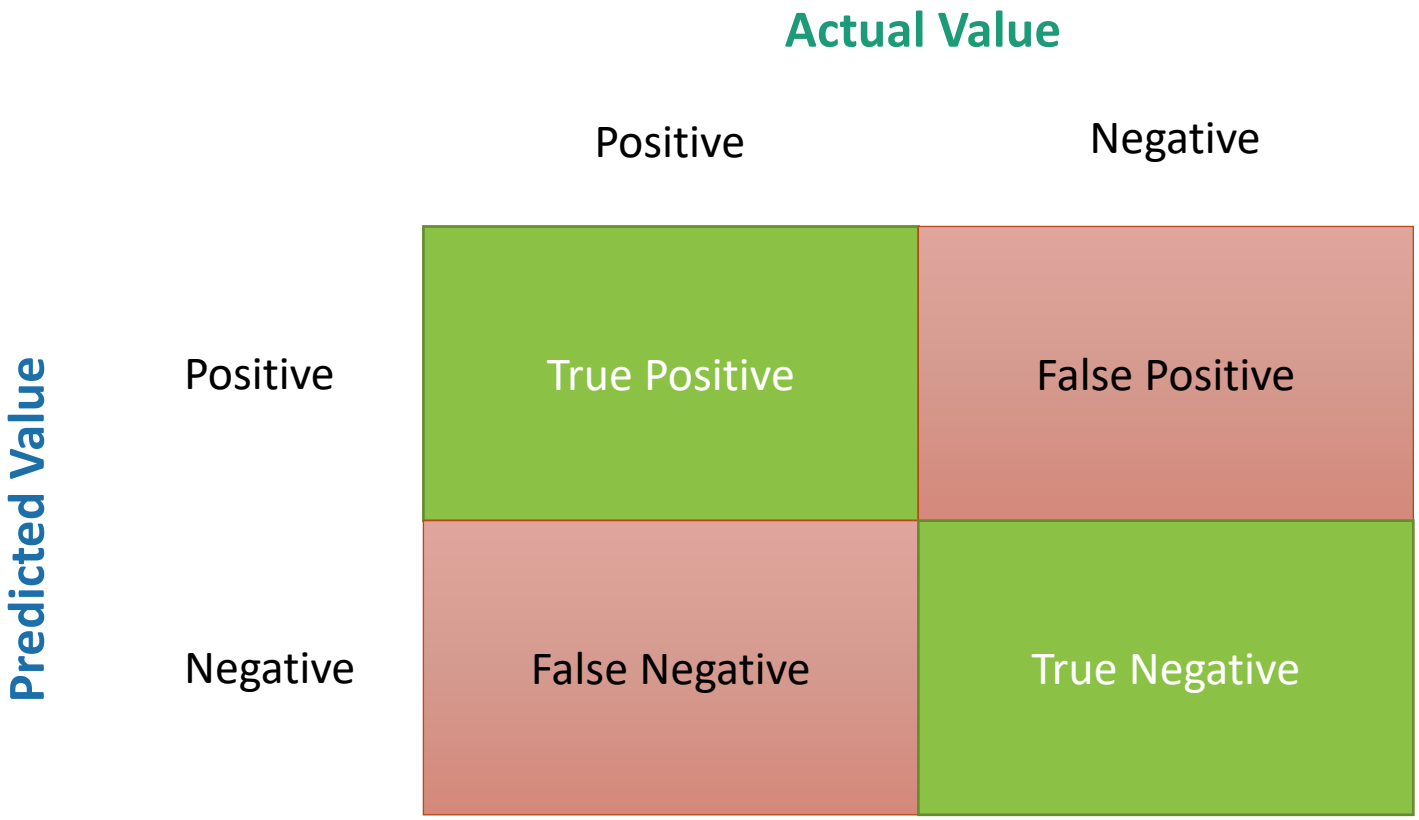
Confusion Matrix

It assists us in classifying IDS behavior

Predicted Value	Actual Value: Positive	Actual Value: Negative
Positive	True Positive	False Positive
Negative	False Negative	True Negative

Confusion Matrix

There are 2 kind of correct decisions and 2 kind of incorrect



Measuring IDS

True positive rate (TPR)

- or Sensitivity
- or **Recall** (R)

TPR = TP / **P** = TP / (TP + FN)

False positive rate (FPR)

- or Fall-out

FPR = FP / **N** = FP / (FP + TN)

Precision

- or positive predictive value (PPV)

PPV = TP / (TP + FP)

Accuracy (ACC)

ACC = (TP + TN) / (P + N)

P and **N** is the actual number of positives and negatives respectively

Measuring IDS

True positive rate (TPR)

- or Sensitivity
- or **Recall** (R)

TPR = TP / **P** = TP / (TP + FN)

Precision

- or positive predictive value (PPV)

PPV = TP / (TP + FP)

Accuracy (ACC)

ACC = (TP + TN) / (P + N)

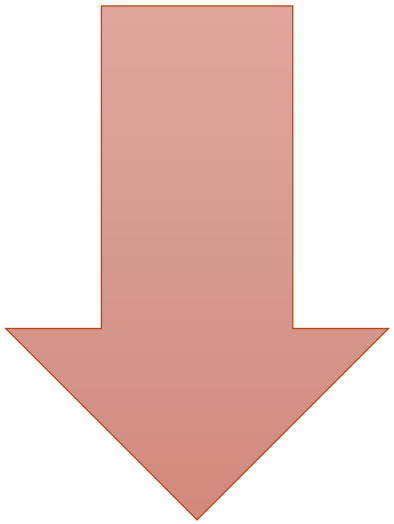
P and **N** is the actual number of positives and negatives respectively

Measuring IDS

False positive rate (FPR)

- or Fall-out

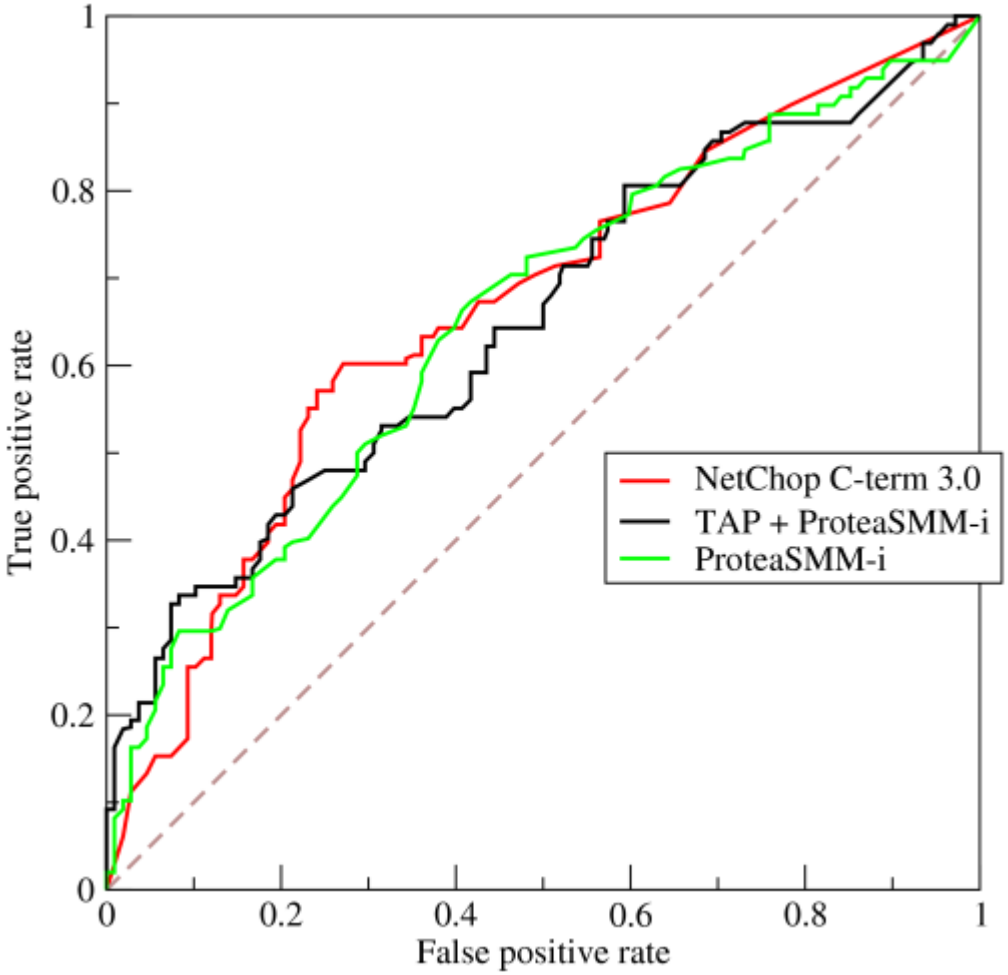
FPR = FP / **N** = FP / (FP + TN)



Lower is better

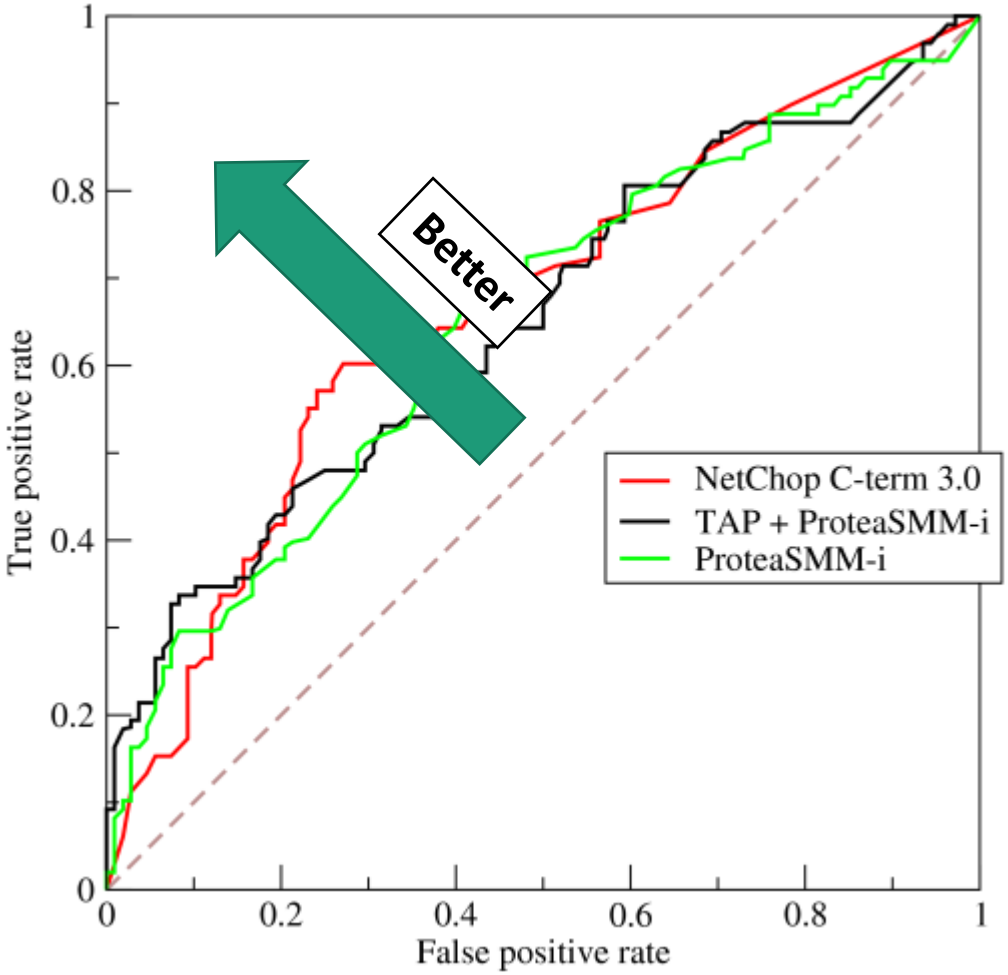
P and **N** is the actual number of positives and negatives respectively

Receiver Operating Characteristic (ROC)



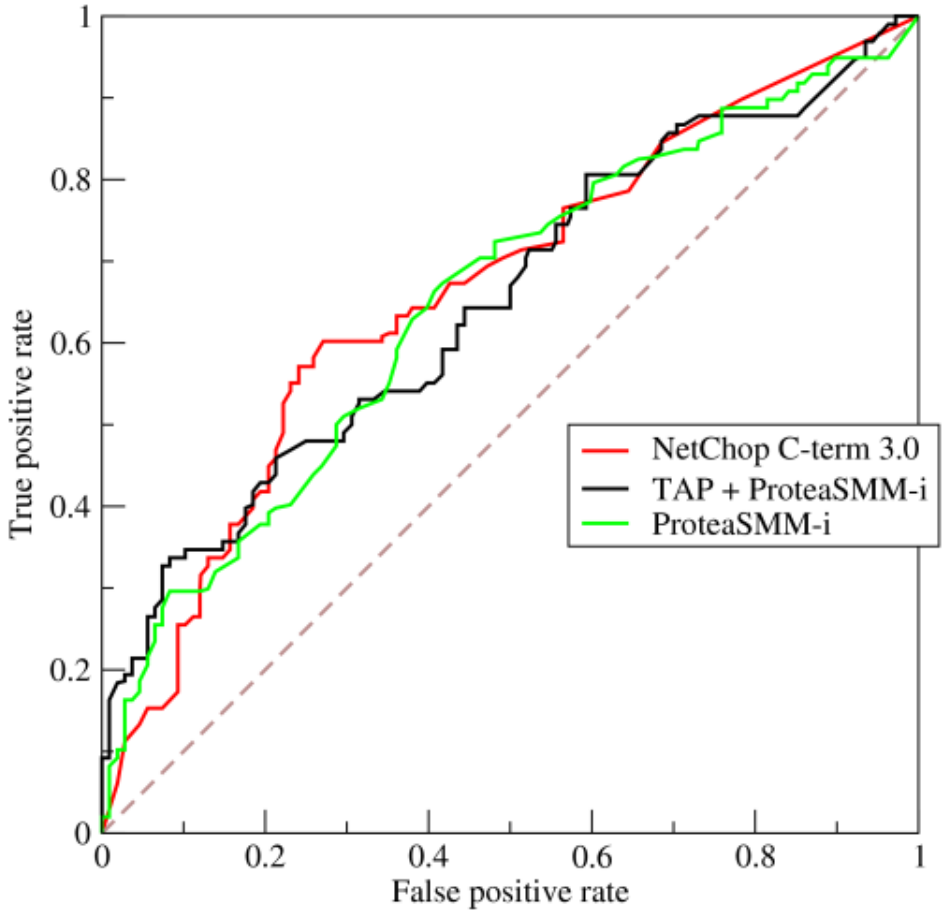
From Wikipedia

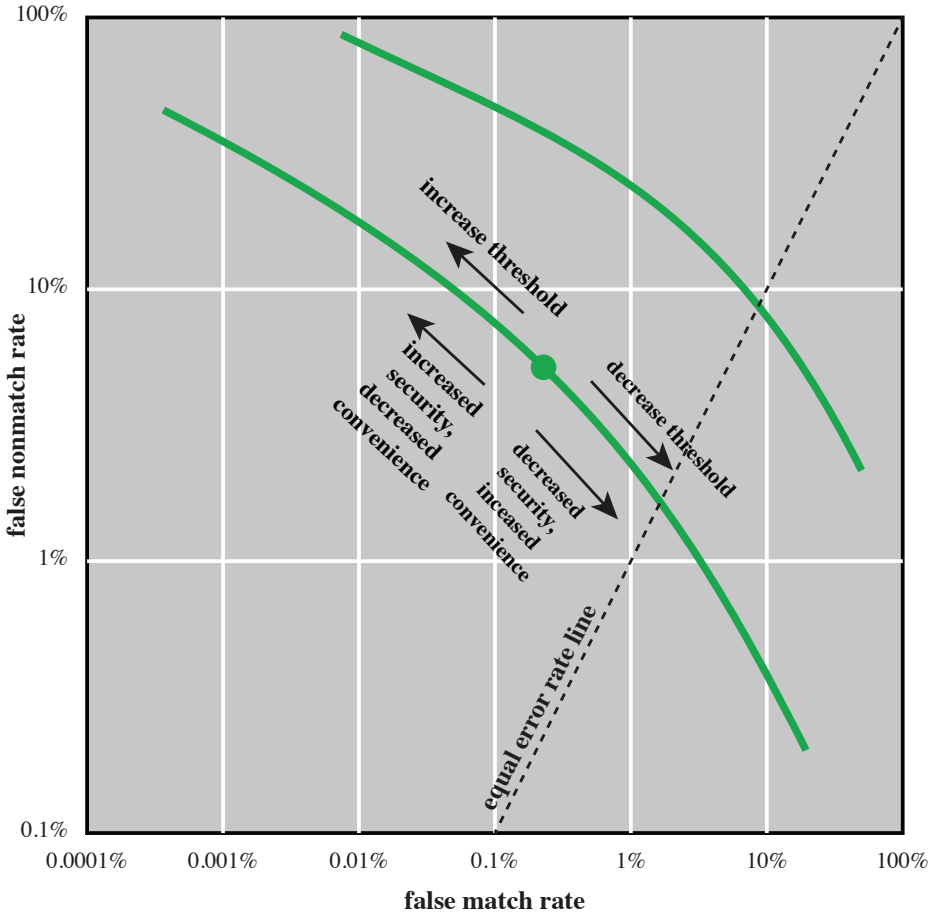
Receiver Operating Characteristic (ROC)



From Wikipedia

Receiver Operating Characteristic (ROC)





False-negative Rate

What is the formula for FNR?

False-negative Rate

What is the formula for FNR?

FNR = FN / **P** = FN / (FN + TP)

Host-Based IDS (HIDS)

Very broad

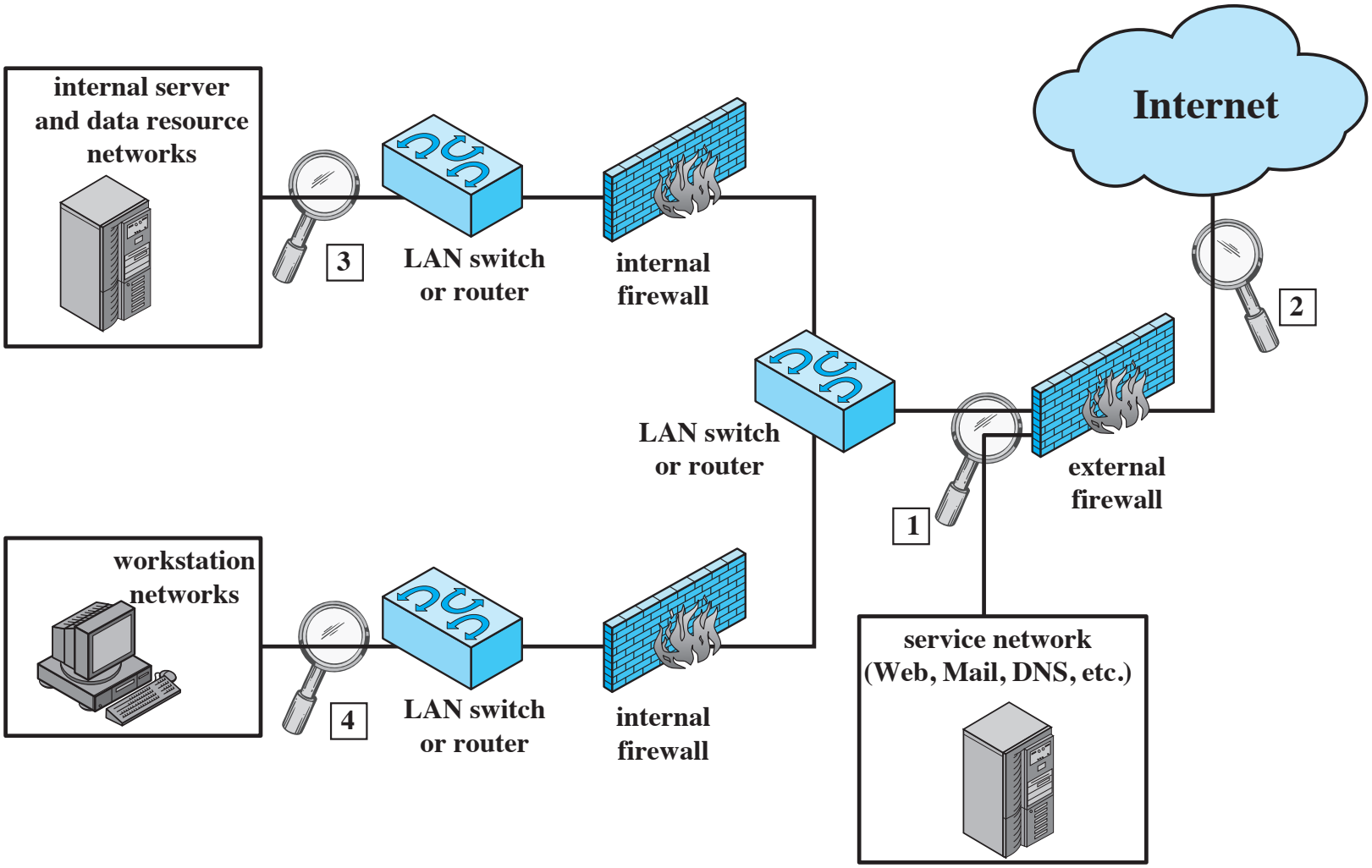
Adds a specialized layer of security software to vulnerable or sensitive systems

Can use either anomaly or signature and heuristic approaches

Collected data can be any OS artifact

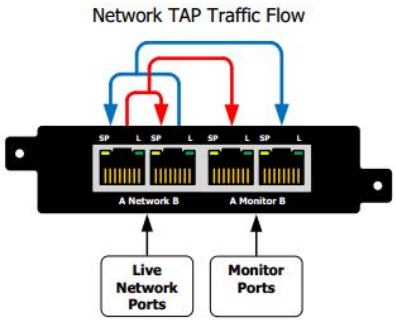
- Files, system and function calls, memory and CPU consumption, etc.

Network-Based Intrusion Detection System (NIDS)

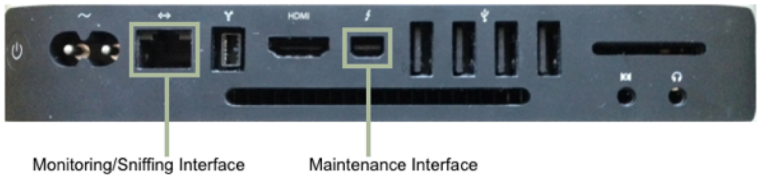


Obtaining Data

Ethernet taps clone data to the NIDS box



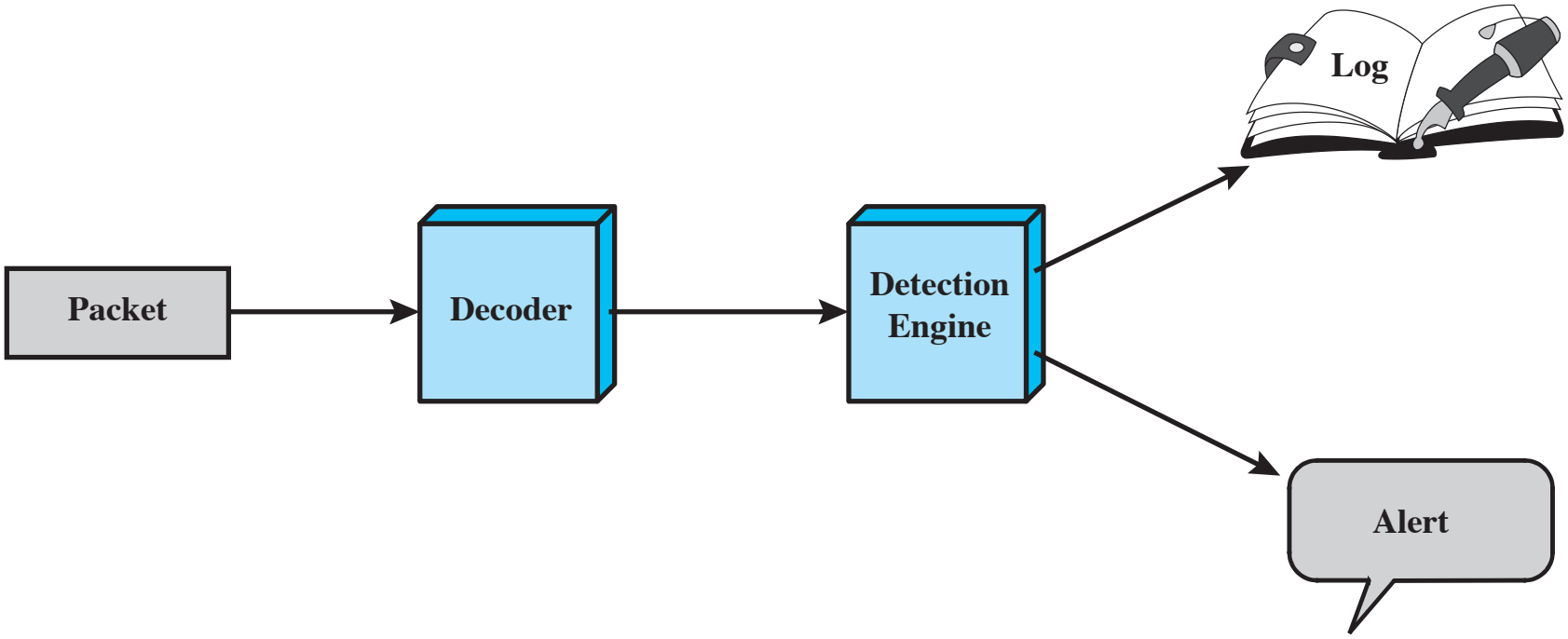
Network elements with built-in NIDS



A common server for lower-bandwidth networks

Snort IDS

SNORT is one of the most well known rule-based NIDS (https://www.snort.org/)



Example Snort Rule: TCP

```
alert tcp any any -> 192.168.1.0/24 111
(content:"|00 01 86 a5|"; msg: "mountd access";)
```

Match

- **TCP** packets
- Source is
  - **any** IP address
  - **any** port
- Destination is
  - an IP address on the **192.168.1.0/24** network
  - TCP port **111**
- Packet content includes the hex values **00 01 86 a5**

Example Snort Rule: TCP

```
alert tcp any any -> 192.168.1.0/24 111
(content:"|00 01 86 a5|"; msg: "mountd access";)
```

Match

- **TCP** packets
- Source is
  - **any** IP address
  - **any** port
- Destination is
  - an IP address on the **192.168.1.0/24** network
  - TCP port **111**
- Packet content includes the hex values **00 01 86 a5**

Issues an alert with message **"mountd access"**

Example Snort Rule: Slammer Worm

```
alert udp $EXTERNAL_NET any -> $HOME_NET 1434
(msg:"MS-SQL Slammer Worm Activity"; content:"|04 01
01 01 01 01 01 01|"; classtype:bad-unknown; sid:9998;
rev:1;)
```

Which packets will this rule match?

Example Snort Rule: Slammer Worm

```
alert udp $EXTERNAL_NET any -> $HOME_NET 1434
(msg:"MS-SQL Slammer Worm Activity"; content:"|04 01
01 01 01 01 01 01|"; classtype:bad-unknown; sid:9998;
rev:1;)
```

This rule matches packets destined for UDP port number 1434 (MS SQL) containing a sequence of bytes that characterize slammer

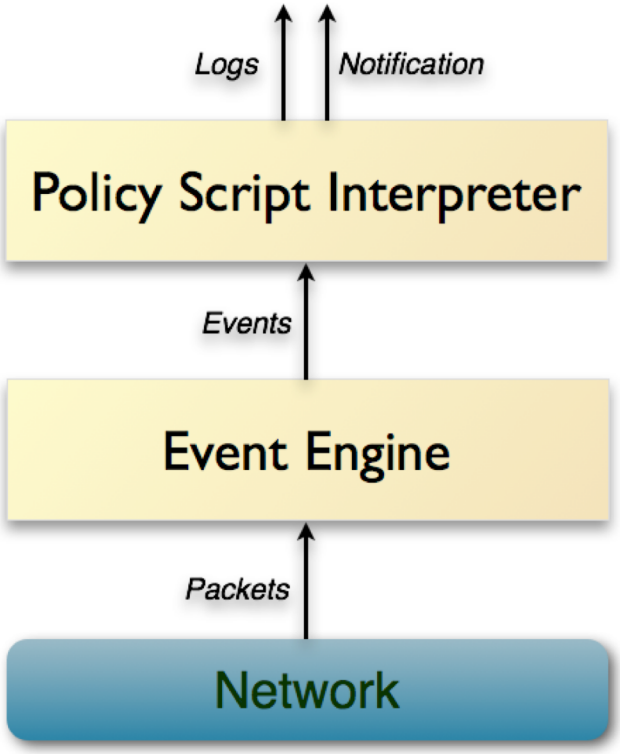
0x04 is the first byte of the packets sent to propagate the worm, followed by a string of 0x01 bytes and finally, the payload of the worm code itself follows

Example Snort Alerts

```
[**] [1:718:6] TELNET login incorrect [**]
[Classification: Potentially Bad Traffic] [Priority: 2]
09/08-15:40:18.391368 94.200.10.71:23 -> 102.60.21.3:1552
TCP TTL:63 TOS:0x10 ID:6313 IpLen:20 DgmLen:85 DF
***AP*** Seq: 0x38DB3A4  Ack: 0x6A4825DA  Win: 0x8218  TcpLen: 32
TCP Options (3) => NOP NOP TS: 460280 739040
[Xref => http://www.whitehats.com/info/IDS127]


[**] [1:498:3] ATTACK RESPONSES id check returned root [**]
[Classification: Potentially Bad Traffic] [Priority: 2]
09/08-16:11:05.410761 102.60.21.3:2323 -> 94.178.4.82:3502
TCP TTL:64 TOS:0x0 ID:24781 IpLen:20 DgmLen:108 DF
***AP*** Seq: 0xDBC0D386  Ack: 0x22CA5EEE  Win: 0x7FF8  TcpLen:
20
```

BRO: Network Security Monitor



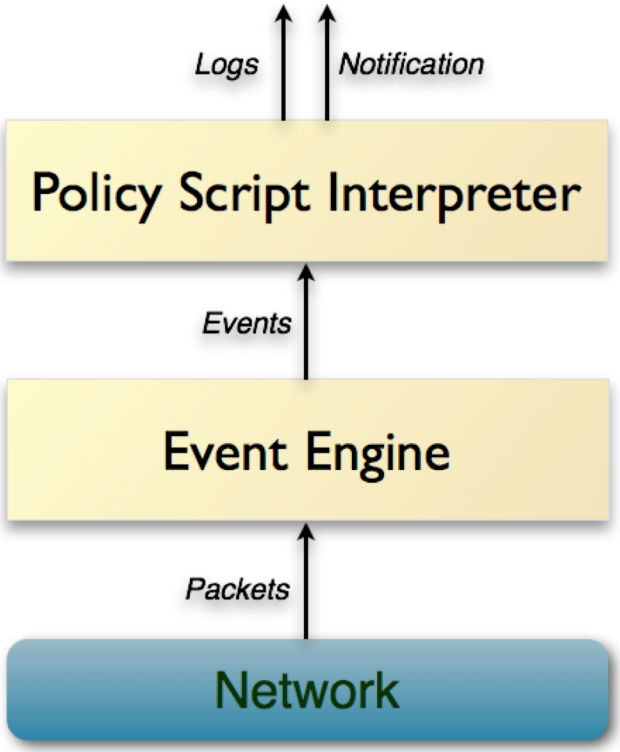
https://www.bro.org/

BRO: Network Security Monitor

Policy is enforced through an event processing loop

```
event signature_match(state:
signature_state, msg: string, data: string)
```

```
signature my-first-sig {
ip-proto == tcp
dst-port == 80
payload /.*root/
event "Found root!"
}
```



Reading

Bro: A System for Detecting Network Intruders in Real-Time

https://www.usenix.org/legacy/publications/library/proceedings/sec98/full_papers/paxson/paxson.pdf

Insertion, Evasion, and Denial of Service: Eluding Network Intrusion Detection

http://cs.unc.edu/~fabian/course_papers/PtacekNewsham98.pdf

Network Intrusion Detection: Evasion, Traffic Normalization, and End-to-End Protocol Semantics

https://www.usenix.org/legacy/events/sec01/full_papers/handley/handley.pdf

Honeypots

Computing Honeypot

A decoy host or network

No production purpose

Aims to attract attackers

Heavily monitored

Honey = Something Worth Getting

Data

User data

Credit cards

SSN

Passwords

Corporate data

…

Compromised system

Sends SPAM

Performs DDoS

A stepping stone

Distributes malware

…

Also for Defense

Honeypots or tarpits can be also used for defensive purposes

- Keep attackers preoccupied with dummy systems

Delay network connections

- Slowdown computer worms

Automatically launch counter attacks?

- Unethical, illegal, and dangerous

Honeypot Types based on Interaction

High

Real OS and services

Virtual or physical

Harder to detect

More expensive to maintain

Low

A program

Simulates OS and services

Scripts interact with the attacker instead

Can simulate entire networks (see honeyd)

Targeted Honeypots

Spam honeypots

- Pretend to be an open (misconfigured) email relay server

E-mail traps

- Funnel all emails to non-existing accounts to a monitoring account

Detectable honeypots

- Deter attackers

Monitoring

Depends on the type of honeypot

Typical monitored interfaces

- Network
  - Tcpdump
  - Actual Ethernet taps
- System calls
- Service requests
- Downloaded/uploaded files

Honeynets

A set of honeypots deployed in one or more networks doing collaborative monitoring

Examples:

- https://www.honeynet.org/
- http://www.leurrecom.org/
- http://www.honeyathome.org/
- SweetBait: Zero-hour worm detection and containment using low- and high-interaction honeypots