ECE560
Computer and Information Security

Fall 2023

Intrusion Detection and Prevention

Tyler Bletsch
Duke University

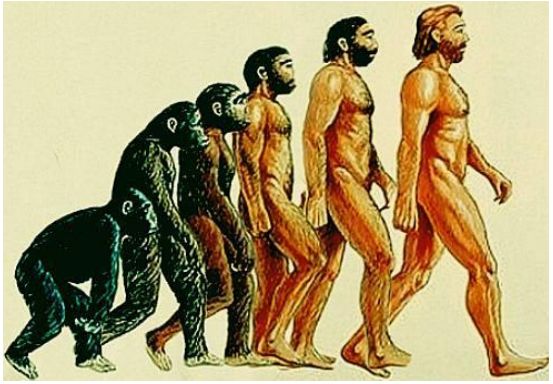
Outline

- Understanding intruders
- Intrusion detection system (IDS)
- Intrusion prevention systems (IPS)
- Detection theory
- Firewalls

Two ways to categorize intruders

- **Class of intruder:** What are they after?

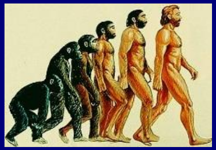
- **Intruder skill level:** How smart are they?



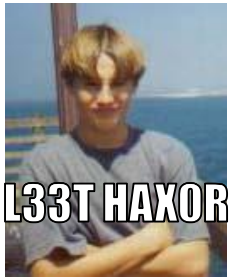
Classes of intruder

- **Class of intruder:** What are they after?
  - **Criminal** want to **monetize:** Turn attacks into money
    - Methods: Identity theft, corporate espionage, data theft, ransomware
    - Often Eastern European or southeast Asian (but *every* country has them)
    - Collaborate on dark web forums, conduct business on illicit sales sites
  - **Activists** want to **achieve political ends**
    - Methods: Deface websites, conduct DoS attacks, steal and leak data
  - **State-sponsored actors** want to **<u>really</u> achieve political ends**
    - Sponsored by governments. Also known as **Advanced Persistent Threats (APTs)** – covert, professional, long-term
    - Recent trends: Russia, China, and Iran attacking western powers; covert western counterattacks and overt western revelations
  - **Explorers:** motivated by learning or prestige
  - **Script kiddies**: using published tools to cause mischief

Two ways to categorize intruders



- **Intruder skill level:** How smart are they?



  - Apprentice
    - Minimal technical skills, use existing tools
    - Largest group, includes most criminals
    - Easiest to defend against

All-of-you.jpg

  - Journeyman
    - Can modify existing tools and exploit newly published vulnerabilities
    - Can discover some vulnerabilities

  - Master
    - Highly skilled, can discover new vulnerabilities broadly
    - Writes their own tools
    - Common in APT crews and at the top of criminal organizations
    - Hardest to defend against

Intruders will want you to misapprehend their skill and motivation!

- Criminals may want to seem like political activists to cover their true activities.
- Apprentices want to appear like Masters.
- Masters want to appear like Apprentices.
- Etc.

- During forensics, be hesitant to jump to conclusions…

Intruder Behavior

1. Target acquisition and information gathering
2. Initial access
3. Privilege escalation
4. Information gathering or system exploit
5. Maintaining access
6. Covering tracks

Table 8.1

Examples of Intruder Behavior

(a) Target Acquisition and Information Gathering

- Explore corporate website for information on corporate structure, personnel, key systems, as well as details of specific web server and OS used.
- Gather information on target network using DNS lookup tools such as dig, host, and others; and query WHOIS database.
- Map network for accessible services using tools such as NMAP.
- Send query email to customer service contact, review response for information on mail client, server, and OS used, and also details of person responding.
- Identify potentially vulnerable services, eg vulnerable web CMS.

(b) Initial Access

- Brute force (guess) a user's web content management system (CMS) password.
- Exploit vulnerability in web CMS plugin to gain system access.
- Send spear-phishing email with link to web browser exploit to key people.

(c) Privilege Escalation

- Scan system for applications with local exploit.
- Exploit any vulnerable application to gain elevated privileges.
- Install sniffers to capture administrator passwords.
- Use captured administrator password to access privileged information.

(d) Information Gathering or System Exploit

- Scan files for desired information.
- Transfer large numbers of documents to external repository.
- Use guessed or captured passwords to access other servers on network.

(e) Maintaining Access

- Install remote administration tool or rootkit with backdoor for later access.
- Use administrator password to later access network.
- Modify or disable anti-virus or IDS programs running on system.

(f) Covering Tracks

- Use rootkit to hide files installed on system.
- Edit logfiles to remove entries generated during the intrusion.

(Table can be found on pages 271-272 in textbook.)

Outline

- Understanding intruders
- Intrusion detection system (IDS)
- Intrusion prevention systems (IPS)
- Detection theory
- Firewalls

Intrusion Detection System (IDS)

- 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
  - 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

Comprises three logical components:

- **Sensors - collect data**
- **Analyzers - determine if intrusion has occurred**
- **User interface - view output or control system behavior**

Analysis Approaches

Anomaly detection

- Collect data relating to the behavior of legitimate users
- Current observed behavior is compared to baseline
- Detect:
  - Denial-of-service (DoS) attacks
  - Scanning
  - Worms

Signature/Heuristic detection

- Scan for known malicious data patterns via signature (e.g. antivirus) or rules (e.g. 'snort')
- Can only identify known attacks
- Detect:
  - Reconnaissance and attacks
  - Unexpected application services
  - Policy violations

Anomaly Detection

A variety of classification approaches are used:

Statistical	Knowledge based	Machine-learning
• Analysis of the observed behavior using univariate, multivariate, or time-series models of observed metrics	• Approaches use an expert system that classifies observed behavior according to a set of rules that model legitimate behavior	• Approaches automatically determine a suitable classification model from the training data using data mining techniques

Host-Based Intrusion Detection (HIDS)

- Primary purpose is to detect intrusions, log suspicious events, and send alerts
  - Can detect both external and internal intrusions

- Data sources:
  - System call traces
  - Audit (log file) records
  - File integrity checksums
  - Registry access

Distributed HIDS deployment

- Can put HIDS agents on many systems, manage centrally

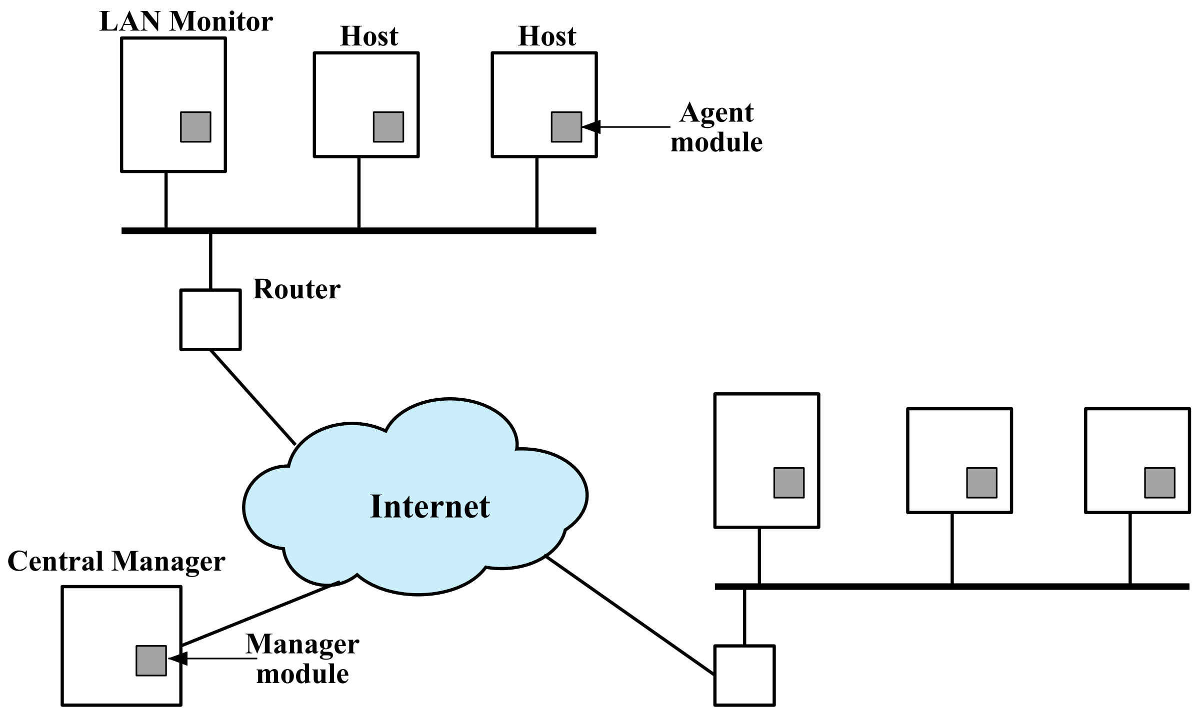
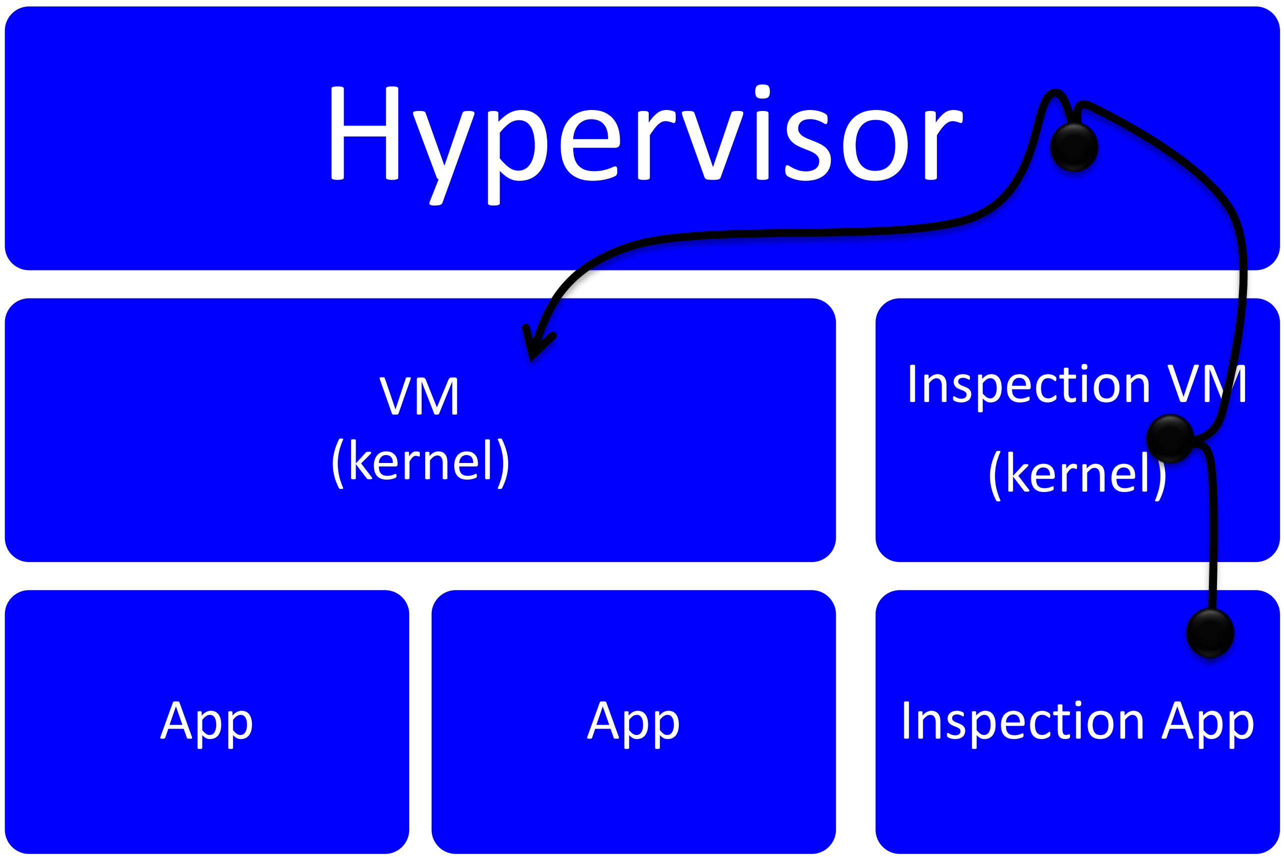


Figure 8.2 Architecture for Distributed Intrusion Detection

ALSO: Virtual Machine Introspection

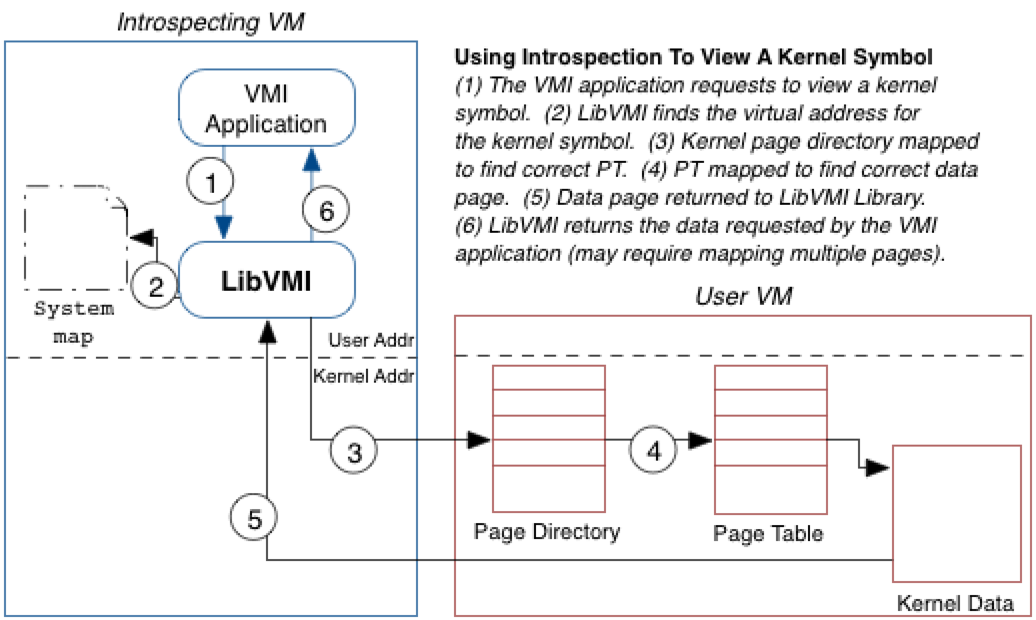
- Look at a VM from the outside

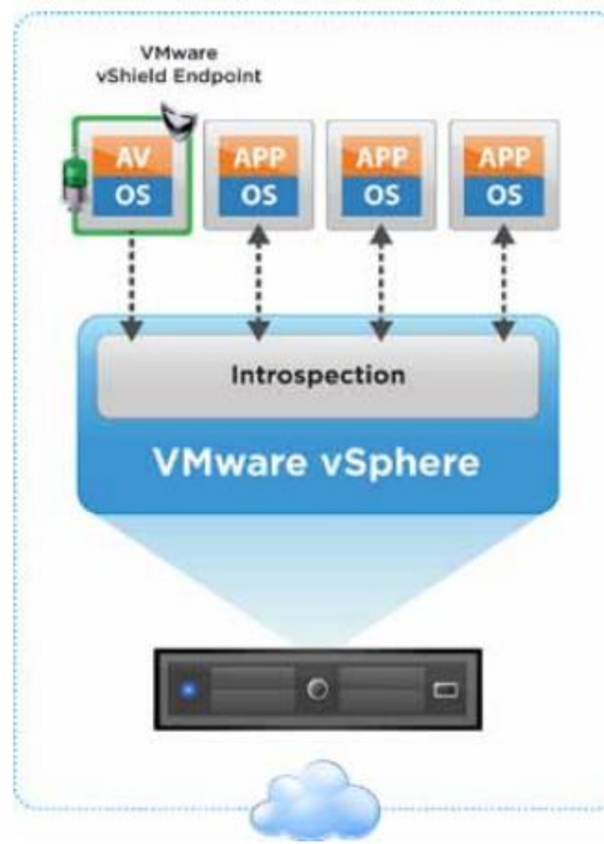


Challenge:
The Semantic
Gap

Virtual Machine Introspection

- Examples: libVMI, VMware vShield Endpoint, etc.





Network-Based IDS (NIDS)

- **Monitors traffic** at selected points on a network
- **Examines traffic** packet by packet in real time
  - May examine network, transport, and/or application-level protocol activity
- **Comprised of**:
  - A number of sensors
  - One or more management servers

- Analysis of traffic patterns may be done at the sensor, the management server or a combination of the two

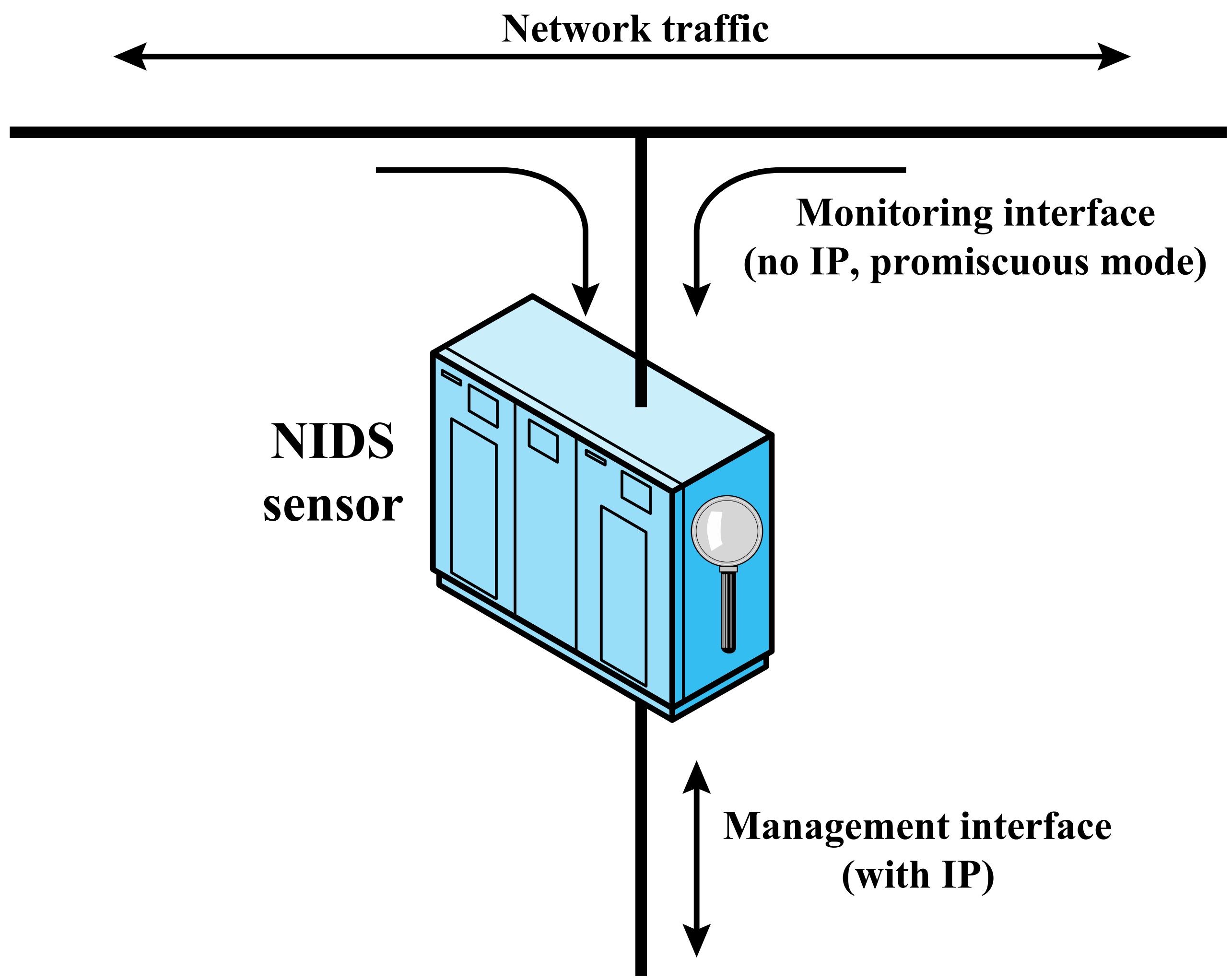


Figure 8.4 Passive NIDS Sensor

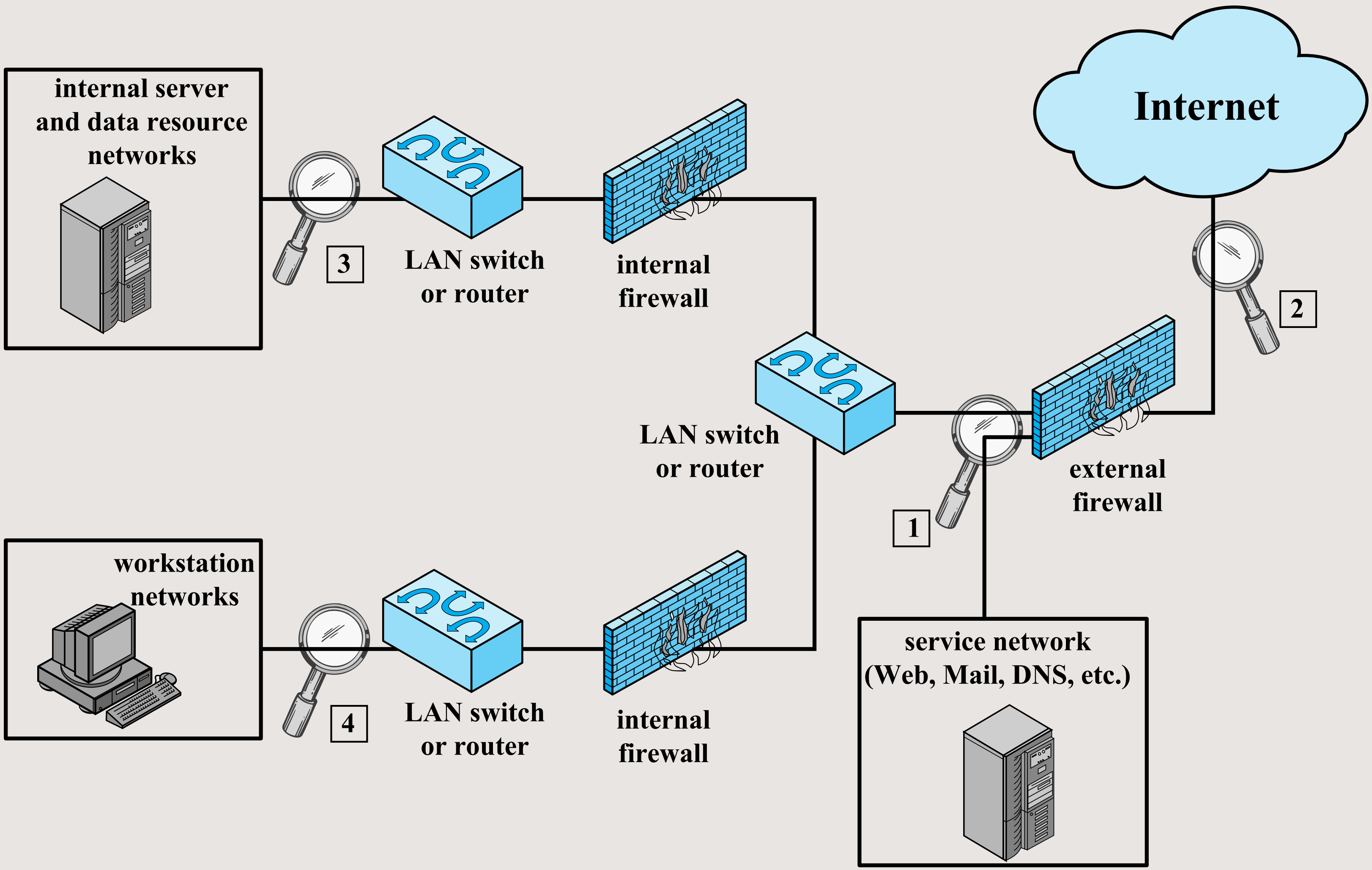


Figure 8.5 Example of NIDS Sensor Deployment

Stateful Protocol Analysis

- Understands and tracks network, transport, and application protocol **states** to ensure they progress as expected
- Higher resource use than stateless systems

Logging of Alerts

- Typical information logged by a NIDS sensor includes:
  - Timestamp
  - Connection or session ID
  - Event or alert type
  - Rating
  - Network, transport, and application layer protocols
  - Source and destination IP addresses
  - Source and destination TCP or UDP ports, or ICMP types and codes
  - Number of bytes transmitted over the connection
  - Decoded payload data, such as application requests and responses
  - State-related information

Flow records

- Modern IDS will often keep **flow records**: info on every TCP connection and UDP flow.
  - Data usually not kept (too big + privacy reasons)
  - Know the connect time, source IP+port, destination IP+port, duration
- Motivation: Historical tracking of suspicious activity
  - "I now know this malware talks to 24.1.2.3, so which of my machines have been talking to that IP?"
  - "I learned that someone at IP address 34.2.3.4 used stolen credentials, where have they been connecting, and have those machines been doing anything weird since then?"
  - "The server became infected at 2:23am, what connections were going on around then?"
  - "Let me scan the flow records and find stuff that looks like portscans so I can investigate!"

Honeypots

- Decoy systems designed to:
  - Lure a potential attacker away from critical systems
  - Collect information about the attacker's activity
  - Encourage the attacker to stay on the system long enough for administrators to respond
- Systems are filled with fabricated information that a legitimate user of the system wouldn't access
- Resources that have no production value
  - Therefore incoming communication is most likely a probe, scan, or attack
  - Initiated outbound communication suggests that the system has probably been compromised
- Classified as being either low or high interaction
  - Low interaction honeypot consists of a software package that emulates particular IT services or systems well enough to provide a realistic initial interaction, but does not execute a full version of those services or systems
  - High interaction honeypot is a real system, with a full operating system, services and applications, which are instrumented and deployed where they can be accessed by attackers

Outline

- Understanding intruders
- Intrusion detection system (IDS)
- Intrusion prevention systems (IPS)
- Detection theory
- Firewalls

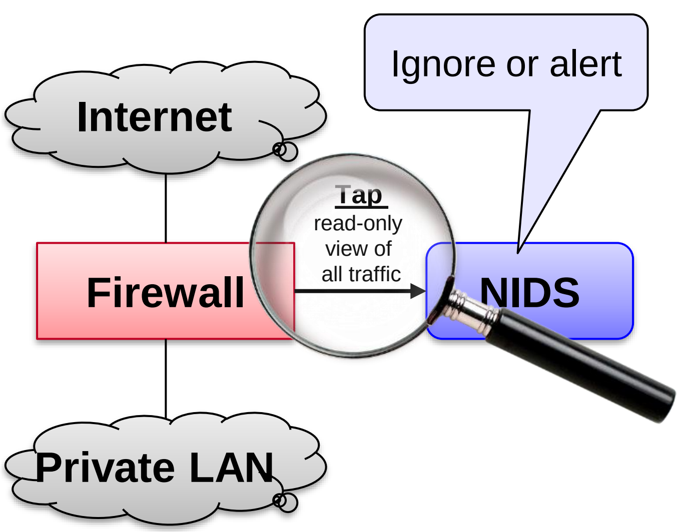
Single slide coverage of (almost) all IPS

Intrusion Prevention System (IPS):
It's IDS that can do something about stuff it sees

Example: NIDS vs NIPS

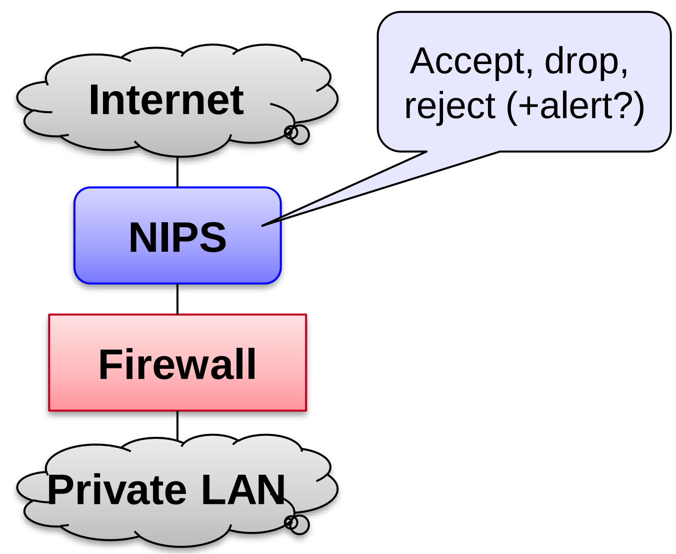
NIDS (mirrored port)

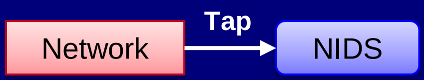
- Can only comment passively on traffic it sees
- False positive: Spurious alert



NIPS (inline)

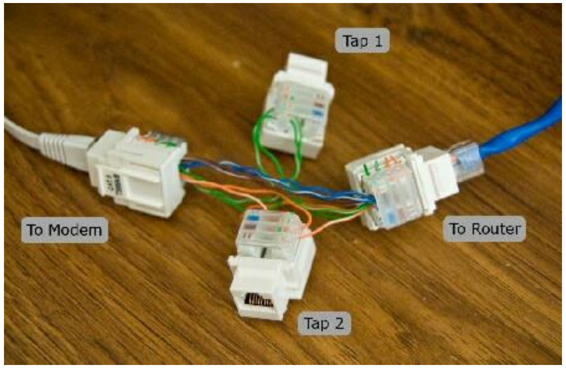
- Can **drop** (ignore) or **reject** (drop with ICMP notice sent to sender) any packet it doesn't like; can also alert.
- False positive: Breaks stuff



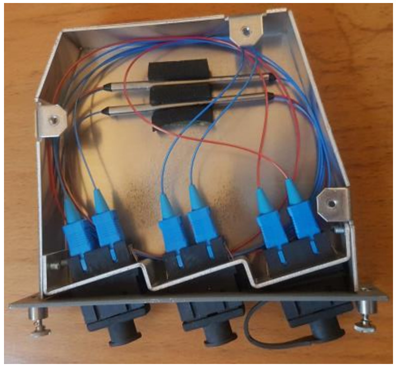


Wait, how do you get all the traffic like that?

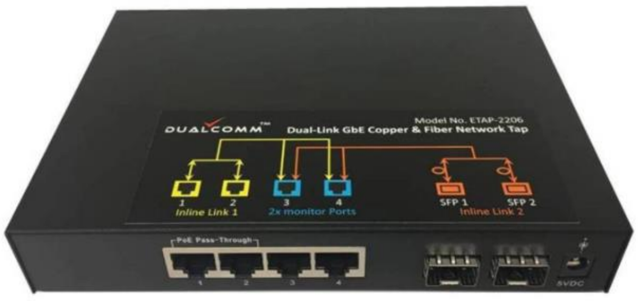
- Network **passive taps:**
  - Classic bidirectional copper (e.g. 100Gb Ethernet): **passive tap** has separate transmit and receive **wires** – literally splice them off
  - Modern optical fiber (e.g. fiber Ethernet): **passive tap again!** separate transmit and receive **fibers** – can use a passive light splitter!
- Network **active taps** (AKA "**port span**"):
  - Can always have hardware that replicates packets to another port
  - Can be done by dedicated hardware or by many modern network switches
    - When done on a switch, it's often called a **port span**



Passive tap for copper Ethernet



Passive tap for fiber Ethernet



Active tap for copper+fiber Ethernet

NIPS at Duke

- All the "Is this your student?" emails I've gotten from OIT were from Duke's IDS/IPS system, which is comprised of several components
- Examples:
  - Portscans are detected using a homespun python script that looks at flow data from a network logger and triggers if unique targets for a given service exceeds a threshold – threshold is configurable per service.
    - Example alert data:

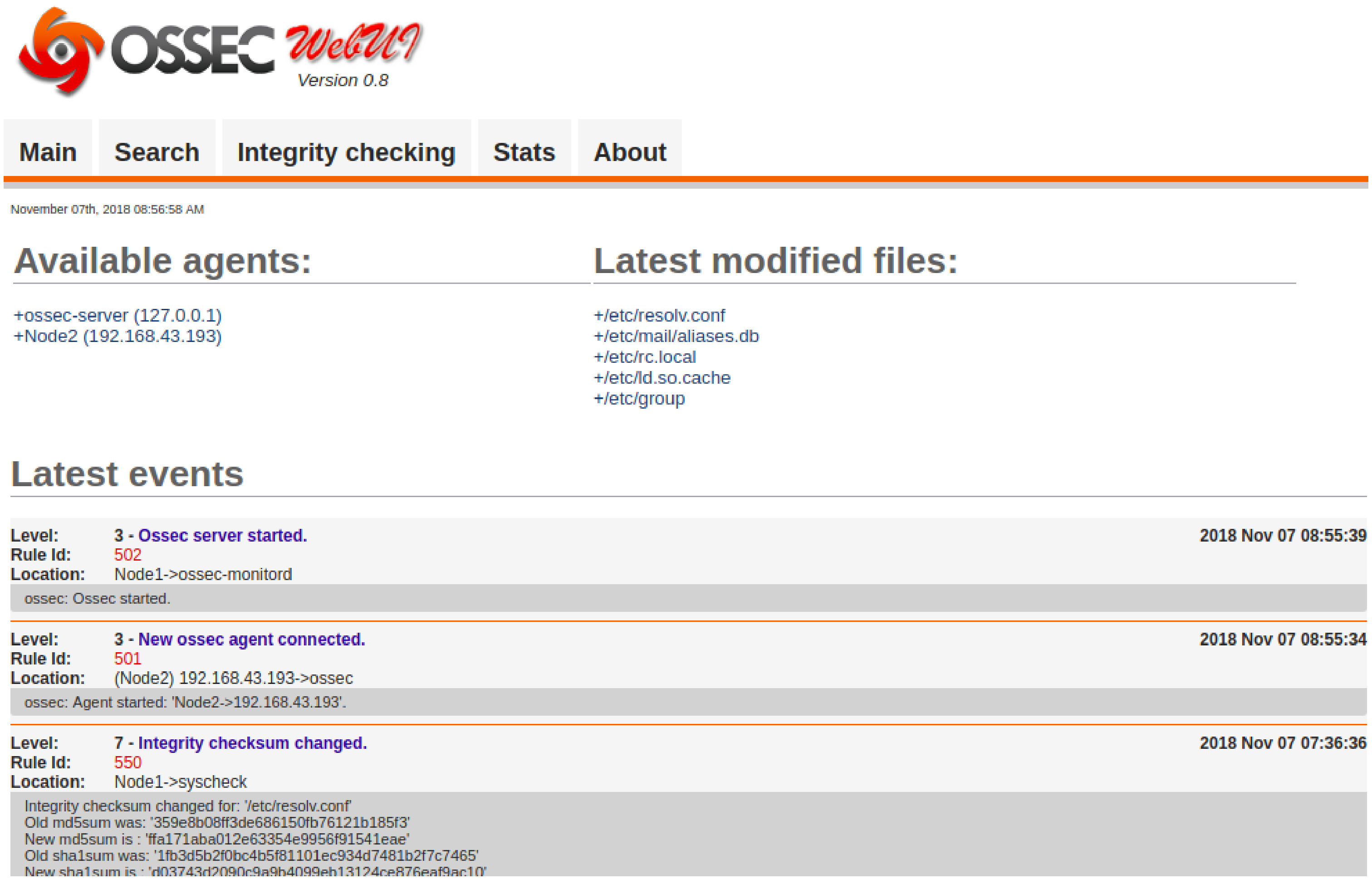
      The alert condition for 'Duke Scanners by IP' was triggered.

      This alert triggers when the argus scanner detect processes detects an IP on our networks that appears the be scanning. The behavior should be investigated to make sure that it was intentional and not malicious. If so and is likely to reoccur, we should see if the IP is static and possibly exclude it from this alert.

      ```
      -------------------------------------------------------------------
      ip,port,hosts_touched,threshold,firstseen,lastseen,host
      152.3.53.133,22,256,50,2018-10-25_20:30:20,2018-10-25_20:55:27,kali-vcm-28.vm.duke.edu
      ```
  - Auto-blocking of VictimCo incoming IP address: Caused because the unencrypted reverse shell content contained info reading an .htaccess and/or .htpasswd file (one of many rules that this flow would eventually violate)
    - "Solved" by whitelisting VictimCo with OIT's IDS/IPS systems

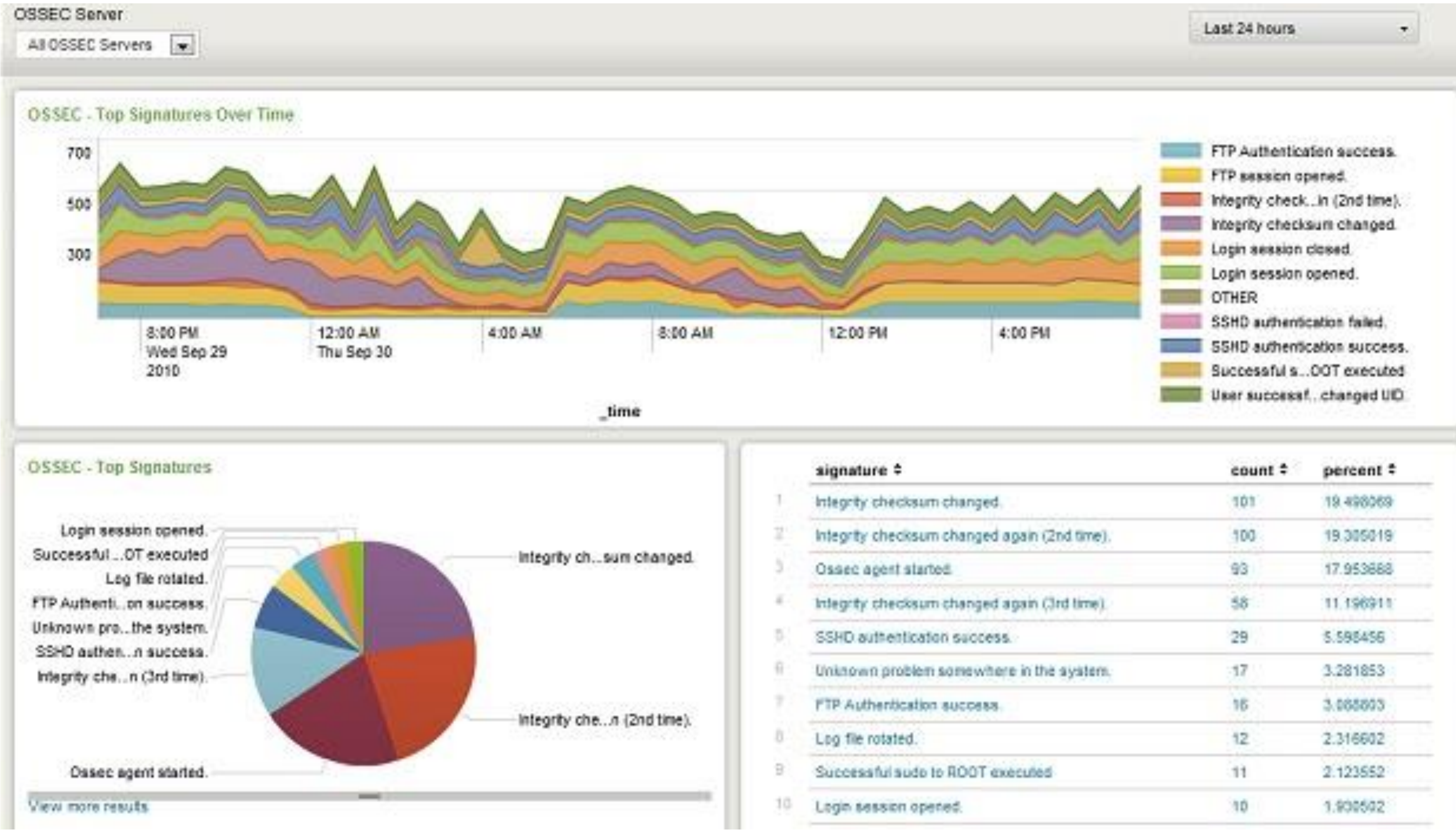
Examples of free modern IDS/IPS

- **OSSEC**: Open source, cross platform HIDS



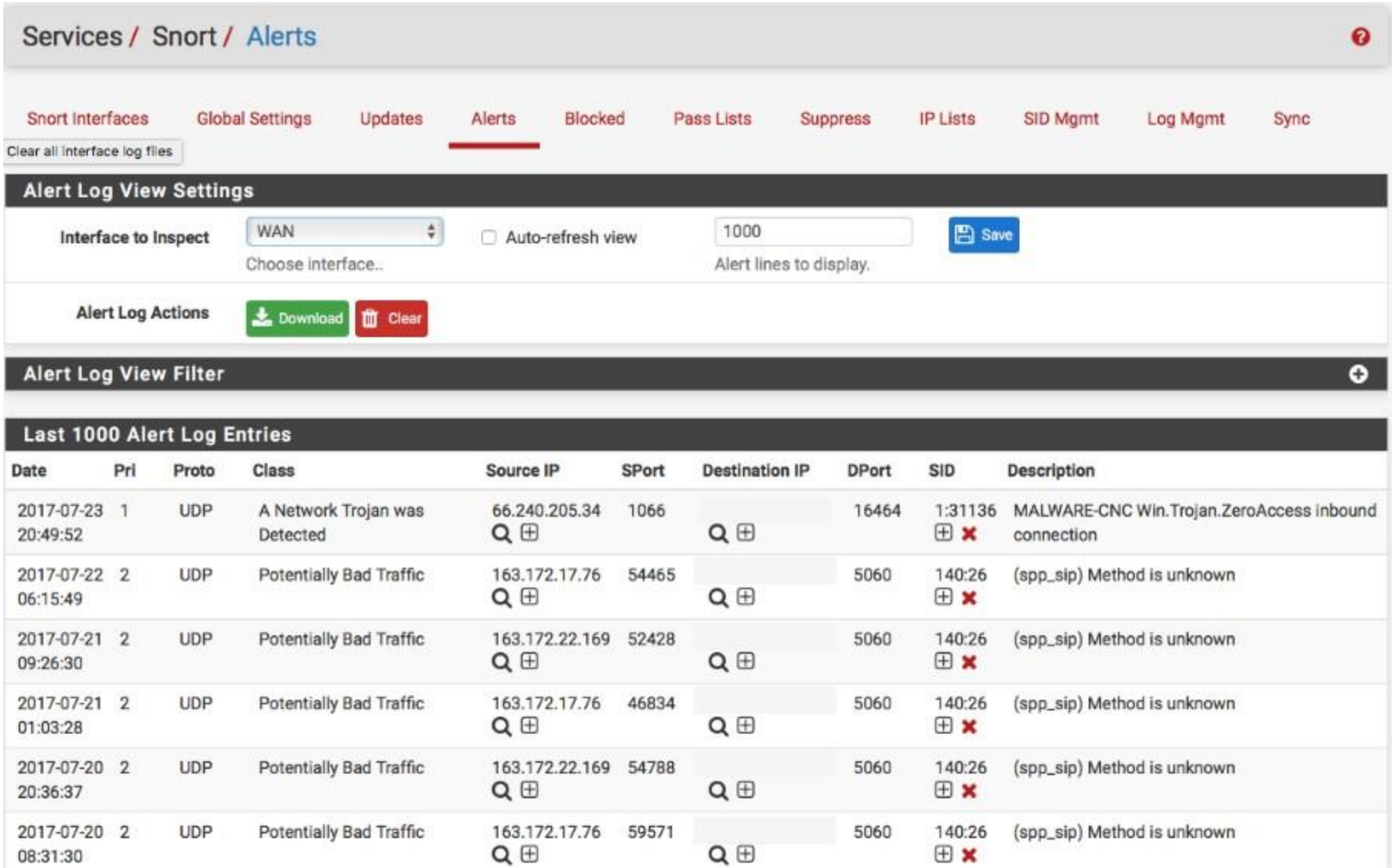
Examples of free modern IDS/IPS

- **Splunk**: Free and premium versions available; covers HIDS+NIDS



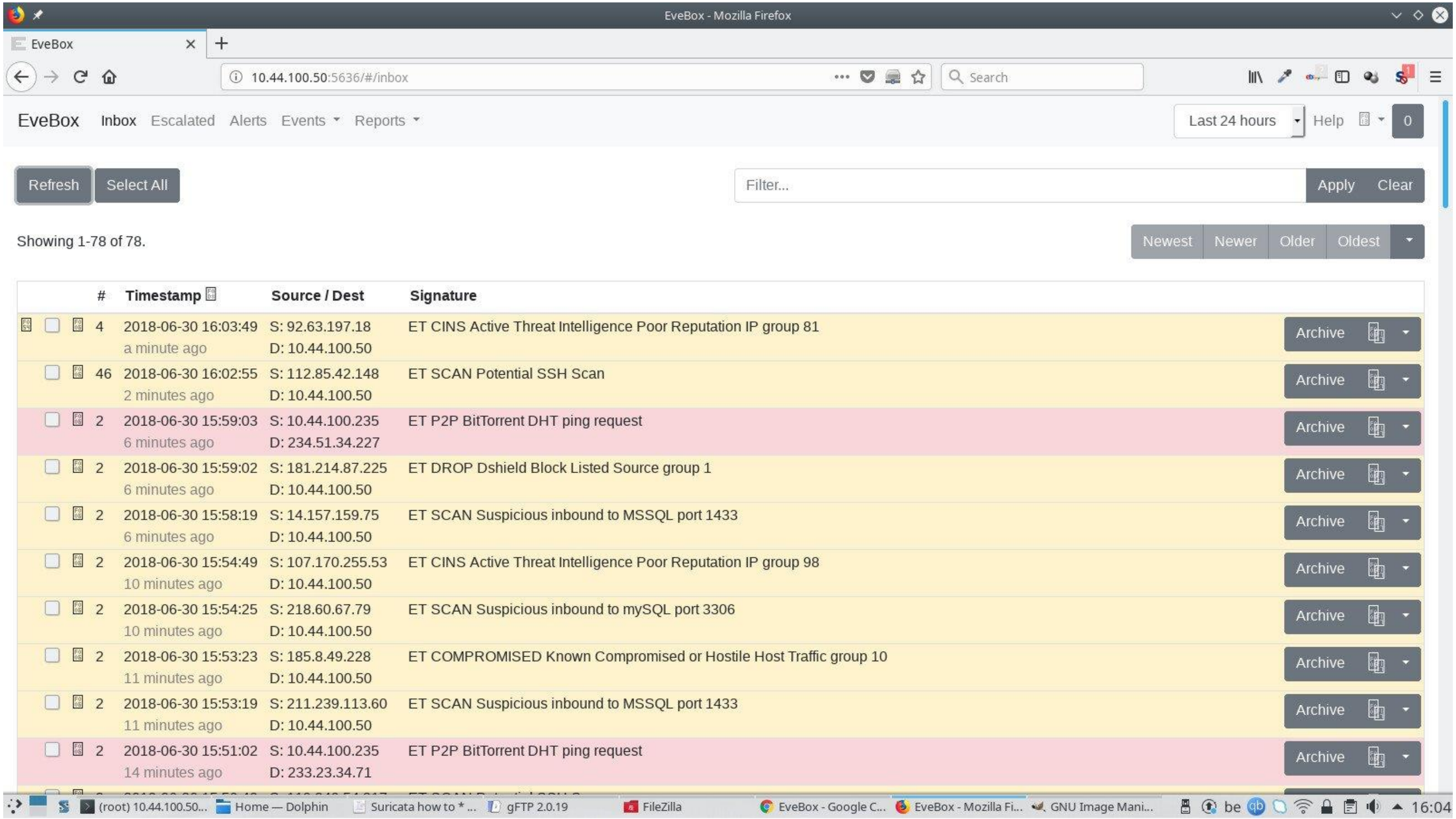
Examples of free modern IDS/IPS

- **Snort**: Open-source NIDS, old and common, single-threaded



Examples of free modern IDS/IPS

- **Suricata**: Open-source NIDS, multi-threaded, bit fancier



Outline

- Understanding intruders
- Intrusion detection system (IDS)
- Intrusion prevention systems (IPS)
- Detection theory
- Firewalls

Problem: We're not sure

- We might say it's **malicious** and we're **right** (**True positive**)

  We detected bad stuff and did something about it! Yay! ☺

- We might say it's **malicious** but we're **wrong** (**False positive**)

  We blocked legitimate stuff! People are mad at us! ☹

- We might say it's **benign** and we're **right** (**True negative**)

  That traffic is cool and good, let it through! Yeah! ☺

- We might say it's **benign** and we're **wrong** (**False negative**)

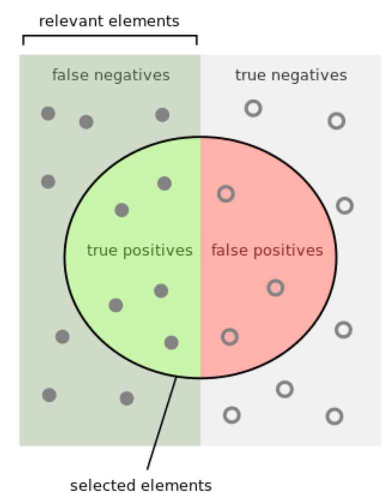
  We missed an attack! Oh no, danger! ☹

Confusion Matrix

- A Confusion matrix is a table describing the performance of some detection algorithm
  - True positives (TP): number of correct classifications of malware
  - True negatives (TN): number of correct classifications of non-malware
  - False positives (FP): number of incorrect classifications of non-malware as malware
  - False negatives (FN): number of incorrect classifications of malware as non-malware

Detection Result

Reality	T	F
T	True Positive	False Negative
F	False Positive	True Negative



https://en.wikipedia.org/wiki/Precision_and_recall

Adapted from material by Patrick McDaniel, Penn State

Metrics

(from perspective of detector)

- **False positive rate**: $FPR = \frac{FP}{FP + TN} = \frac{\text{\# benign marked as malicious}}{\text{total benign}}$
- **True negative rate**: $TNR = 1 - FPR = \frac{TN}{FP + TN} = \frac{\text{\# benign unmarked}}{\text{total benign}}$
- **False negative rate:** $FNR = \frac{FN}{FN + TP} = \frac{\text{\# malicious not marked}}{\text{total malicious}}$
- **True positive rate:** $TPR = 1 - FNR = \frac{TP}{FN + TP} = \frac{\text{\# malicious correctly marked}}{\text{total malicious}}$

<u>Shorthand for this part</u>
"**Alert**" = mark as malicious
"**Malware**" = packet is malicious

Detection Result

Reality	T	F
T	True Positive	False Negative
F	False Positive	True Negative

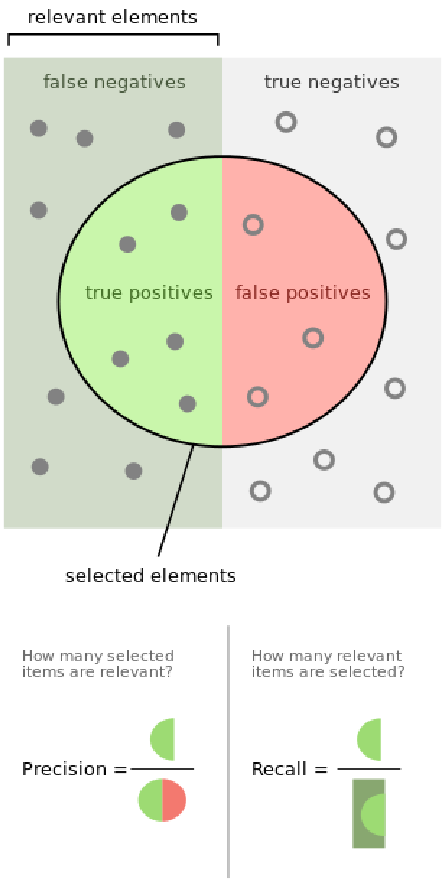
Adapted from material by Patrick McDaniel, Penn State

Precision and Recall

- **Recall** (also known as sensitivity)
  - fraction of correct instances among all instances that actually are positive (malware)
  - TP / (TP + FN)

    ^ Note: This is also the TPR
- **Precision**
  - fraction of correct instances (malware) that algorithm believes are positive (malware)
  - TP / (TP + FP)

Recall: percent of malware you alert on
Precision: percent alerts that are right



https://en.wikipedia.org/wiki/Precision_and_recall

Adapted from material by Patrick McDaniel, Penn State

Bayes Rule

- Pr(x) function, probability of event x
  - Pr(sunny) = 0.8 (80% of sunny day)
- Conditional probability
  - Pr(x|y), probability of x *given* y
  - Pr(cavity|toothache) = 0.6
  - 60% chance of cavity given you have a toothache
- Bayes' Rule (of conditional probability)

Bayes rule of conditional probability

$$Pr(B|A) = \frac{Pr(A|B) \cdot Pr(B)}{Pr(A)}$$

Example:

- Assume: Pr(cavity) = 0.5, Pr(toothache) = 0.1
- What is Pr(toothache|cavity)?
  - = Pr(cavity|toothache)*Pr(toothache)/Pr(cavity)
    = 0.6 * 0.1 / 0.5
    = 0.12

Base Rate Fallacy

- Occurs when assessing P(X|Y) without considering probability of X and the total probability of Y

  Example:

  - Base rate of malware is 1 packet in a 10,000 → Pr(Malware) = 0.00001
  - Intrusion detection system is 99% accurate → Pr(Alert | Malware) = 0.99
  - 1% false positive rate (alert on benign) → Pr (Alert | !Malware) = 0.01
  - 1% false negative rate (fail to alert on malicious) → Pr (!Alert | Malware) = 0.01

- A packet is marked by the NIDS as malware.
  What is the probability that packet X actually is malware?
  → Pr(Malware | Alert)

- This is the **precision**: the rate at which an alert is actually true.

  ("How often was alerting someone actually justified?")

Base Rate Fallacy

- Our goal is to find the true alert rate (i.e., Pr(Malware|Alert)) using Bayes rule:

$$\Pr(Malware|Alert) = \frac{\Pr(Alert|Malware) * \Pr(Malware)}{\Pr(Alert)}$$

- We know:
  - 1% false positive rate (benign marked as malicious 1% of the time); TNR= 99%
  - 1% false negative rate (malicious marked as benign 1% of the time); TPR= 99%
  - Base rate of malware is 1 packet in 10,000
- Let's figure the ingredients to this equation…
  - Pr(Alert|Malware) = ? **TPR = 0.99**
  - Pr(Malware) = ? **Base rate = 0.0001**
  - Pr(Alert) = ? **0.01**

$$\Pr(Alert|Malware) = \frac{\#\, malicious\ correctly\ marked}{total\ malicious} = \frac{TP}{FN + TP} = TPR$$

$$\Pr(Alert) = \Pr(Alert|Malware) * \Pr(Malware) + \Pr(Alert|!Malware) * \Pr(!Malware) = (0.99 * 0.0001) + (0.01 * 0.9999) = 0.01$$

Adapted from material by Patrick McDaniel, Penn State

Base rate fallacy …

- Now let's plug into the Bayes rule formula:

$$\Pr(Malware|Alert) = \frac{\Pr(Alert|Malware) * \Pr(Malware)}{\Pr(Alert)}$$

- Using these ingredients:
  - Pr(Alert|Malware) = **0.99**
  - Pr(Malware) = **0.0001**
  - Pr(Alert) = **0.01**

$$= \frac{0.99 \cdot 0.0001}{0.01} = 0.0099$$

OH NO!

- A little less than 1% of alarms are actually malware!
- What does this mean for network administrators?

Almost all the stupid alerts are LIES!!!!

Adapted from material by Patrick McDaniel, Penn State

All the math in one place

Name:	Base rate			Recall		Precision		
Prob:	P(M)	P(A|!M)	P(!A|M)	P(A|M)	P(!A|!M)	P(M|A)	P(A)	P(!M)
Rate:	BR	FPR	FNR	TPR	TNR			
Eqn:		FP/(FP+TN)	FN/(FN+TP)	1-FNR = TP/(FN+TP)	1-FPR = TN/(FP+TN)	TP/(TP+FP) = P(A|M)*P(M)/P(A)	P(A|M)*P(M) + P(A|!M)*P(!M)	1-BR
A	0.0001	0.01	0.01	0.9900	0.9900	0.0098	0.0101	0.9999
B	0.0001	0.01	0.0001	0.9999	0.9900	0.0099	0.0101	0.9999
C	0.0001	0.0001	0.01	0.9900	0.9999	0.4975	0.0002	0.9999
D	0.0001	0.0001	0.0001	0.9999	0.9999	0.5000	0.0002	0.9999

Four possible situations

Note: You can access this spreadsheet – it's here.

Which variable matters most? (1)

Name:	Base rate			Recall		Precision		
Prob:	P(M)	P(A|!M)	P(!A|M)	P(A|M)	P(!A|!M)	P(M|A)	P(A)	P(!M)
Rate:	BR	FPR	FNR	TPR	TNR			
Eqn:		FP/(FP+TN)	FN/(FN+TP)	1-FNR = TP/(FN+TP)	1-FPR = TN/(FP+TN)	TP/(TP+FP) = P(A|M)*P(M)/P(A)	P(A|M)*P(M) + P(A|!M)*P(!M)	1-BR
A	0.0001	0.01	0.01	0.9900	0.9900	0.0098	0.0101	0.9999
B	0.0001	0.01	0.0001	0.9999	0.9900	0.0099	0.0101	0.9999
C	0.0001	0.0001	0.01	0.9900	0.9999	0.4975	0.0002	0.9999
D	0.0001	0.0001	0.0001	0.9999	0.9999	0.5000	0.0002	0.9999

Making this better… …didn't help much ☹

Which variable matters most? (2)

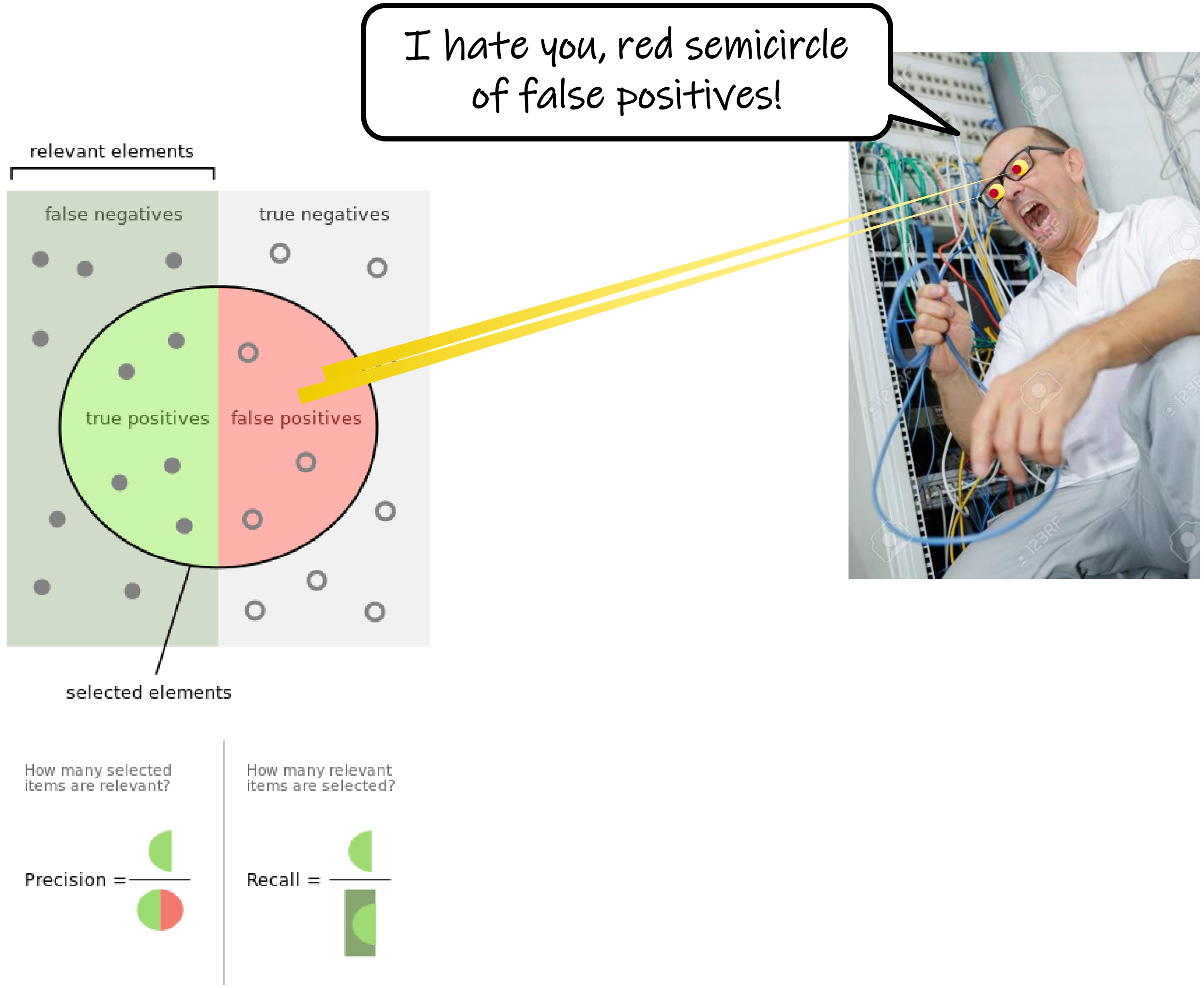
Name:	Base rate			Recall		Precision		
Prob:	P(M)	P(A|!M)	P(!A|M)	P(A|M)	P(!A|!M)	P(M|A)	P(A)	P(!M)
Rate:	BR	FPR	FNR	TPR	TNR			
Eqn:		FP/(FP+TN)	FN/(FN+TP)	1-FNR = TP/(FN+TP)	1-FPR = TN/(FP+TN)	TP/(TP+FP) = P(A|M)*P(M)/P(A)	P(A|M)*P(M) + P(A|!M)*P(!M)	1-BR
A	0.0001	0.01	0.01	0.9900	0.9900	0.0098	0.0101	0.9999
B	0.0001	0.01	0.0001	0.9999	0.9900	0.0099	0.0101	0.9999
C	0.0001	0.0001	0.01	0.9900	0.9999	0.4975	0.0002	0.9999
D	0.0001	0.0001	0.0001	0.9999	0.9999	0.5000	0.0002	0.9999

But making this better…

…helped a lot!! ☺

Base Rate Fallacy conclusion

- In any detection system, you **need a false positive rate as low or lower than the base rate, otherwise most alarms are incorrect!**



Outline

- Understanding intruders
- Intrusion detection system (IDS)
- Intrusion prevention systems (IPS)
- Detection theory
- Firewalls

Firewall Characteristics

Design goals

All traffic from inside to outside, and vice versa, must pass through the firewall

Only authorized traffic as defined by the local security policy will be allowed to pass

The firewall itself is immune to penetration

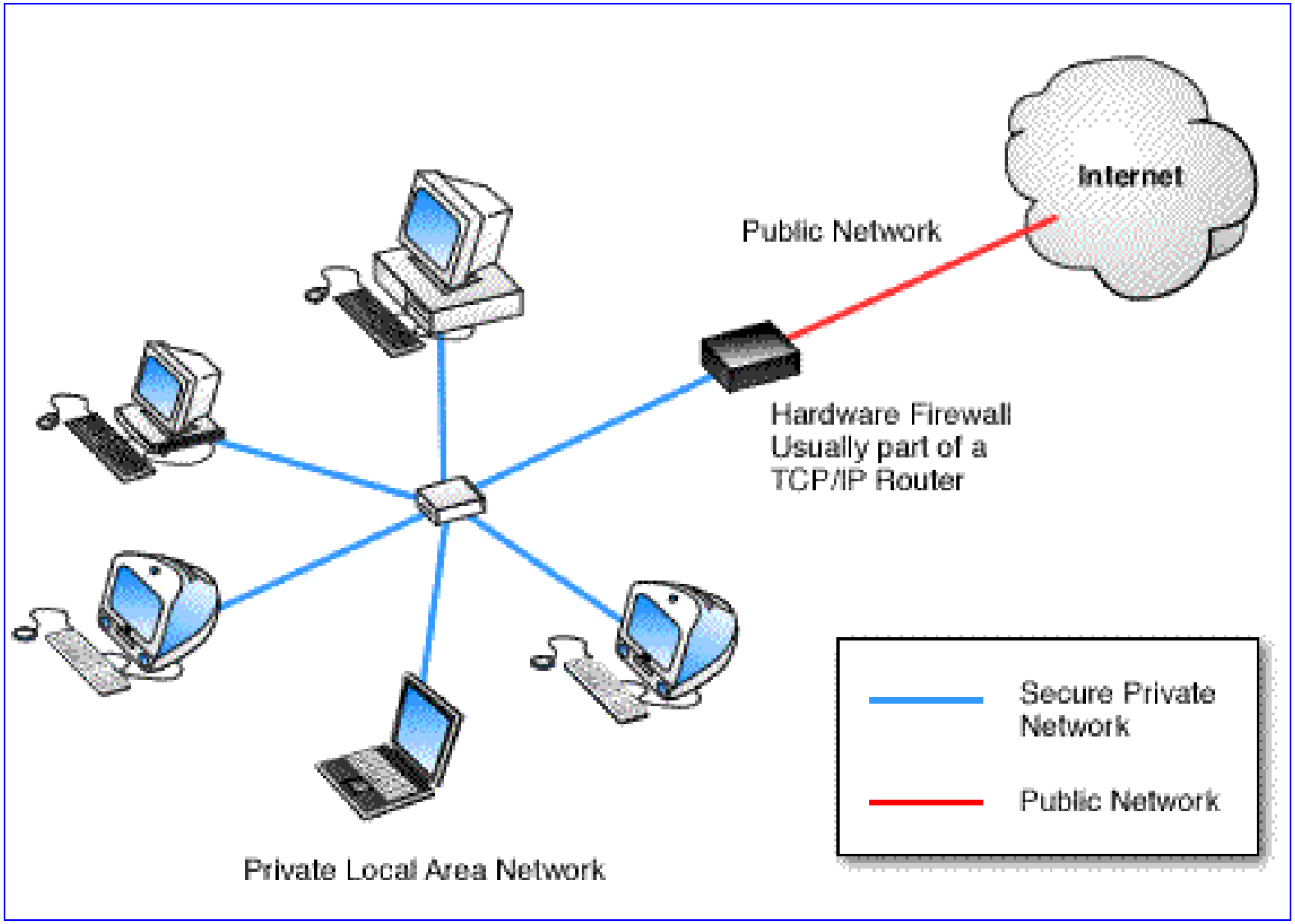
Types of firewalls

Simpler, less expressive, less resource-intensive

Type	Logic	Pros	Cons
Packet filter	Decide on per-packet basis	• Simple • Fast • Easy to configure	• Dumb • Not very expressive
Stateful packet inspection	Decide on stream or higher level basis	• More expressive	• More resource intensive • More configuration
Application-level proxy	Understands app-level traffic	• Can enforce app-relevant restrictions	• Need one customized for each app

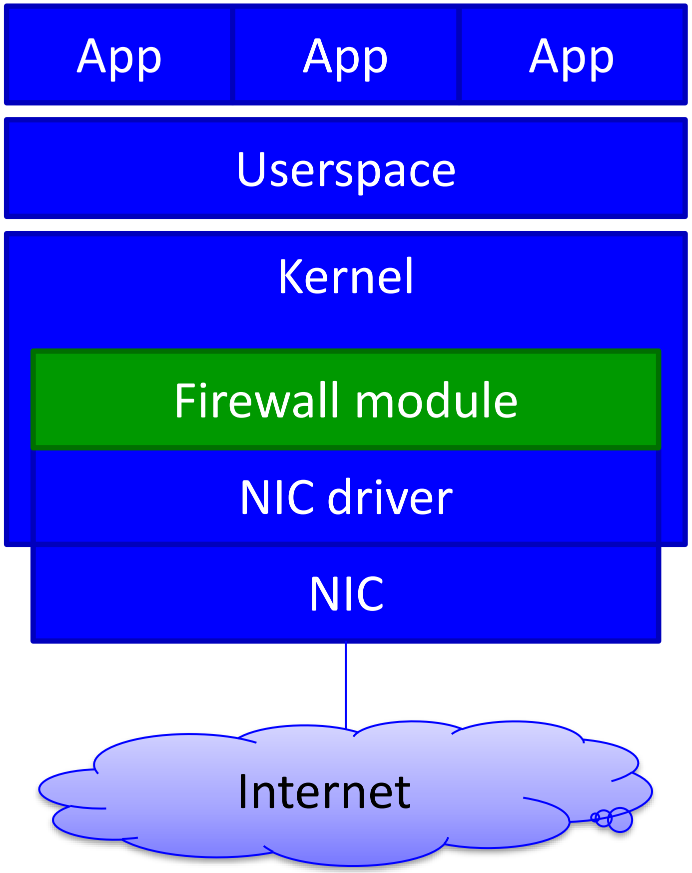
More complex, more expressive, more resource-intensive

Placement of firewalls (1)



LAN firewall

Placement of firewalls (2)



Host-based firewall

Firewall Filter Characteristics

- Characteristics that a firewall access policy could use to filter traffic include:

IP address and protocol values

This type of filtering is used by packet filter and stateful inspection firewalls

Typically used to limit access to specific services

Application protocol

This type of filtering is used by an application-level gateway that relays and monitors the exchange of information for specific application protocols

User identity

Typically for inside users who identify themselves using some form of secure authentication technology

Network activity

Controls access based on considerations such as the time or request, rate of requests, or other activity patterns

Limitations of firewalls

- Book spends a long time on this, but it's simple:
  firewalls have human-built rules *and*
  can only deal with packets that go through them.
- Two scenarios they don't help:
  - HTTP service has a vulnerability and firewall allows HTTP
    (Firewall set to allow the bad thing)
  - Firewall is at ISP uplink but rogue cell phone gets inside of LAN via WiFi
    (Firewall not traversed to do the bad thing)

Packet Filtering Firewall

- Applies rules to each incoming and outgoing IP packet
  - Typically a list of rules based on matches in the IP or TCP header:
    - Source IP address
    - Destination IP address
    - Source and destination transport-level address
    - IP protocol field
    - Interface

- Two default policies:
  - DROP - prohibit unless expressly permitted
    - More conservative, controlled, visible to users
  - ACCEPT - permit unless expressly prohibited
    - Easier to manage and use but less secure

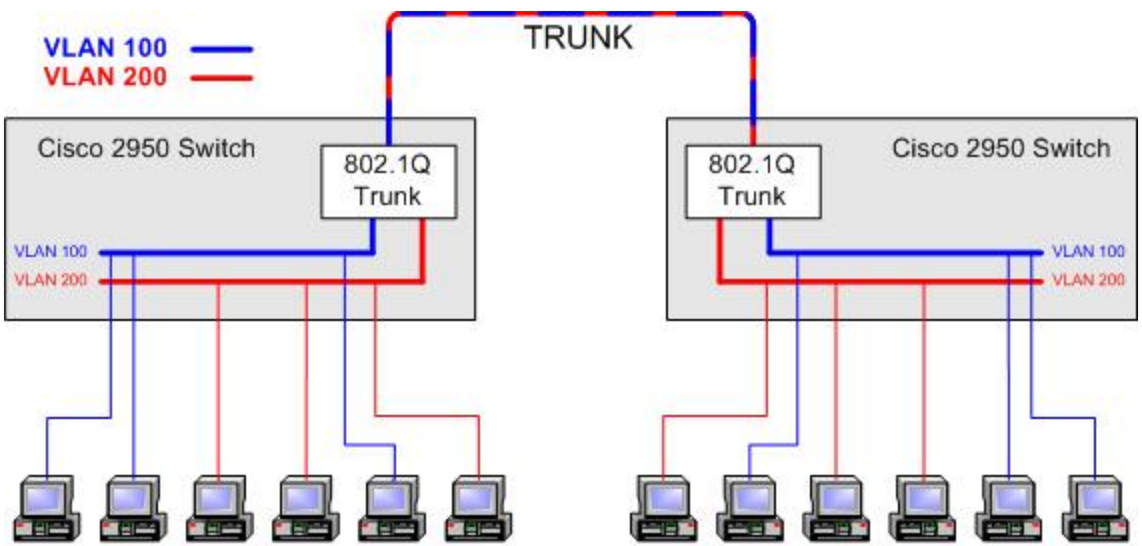
Table 9.1
Packet-Filtering Examples

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

Reminder: VLANs exist

This slide is from way back in the network lecture

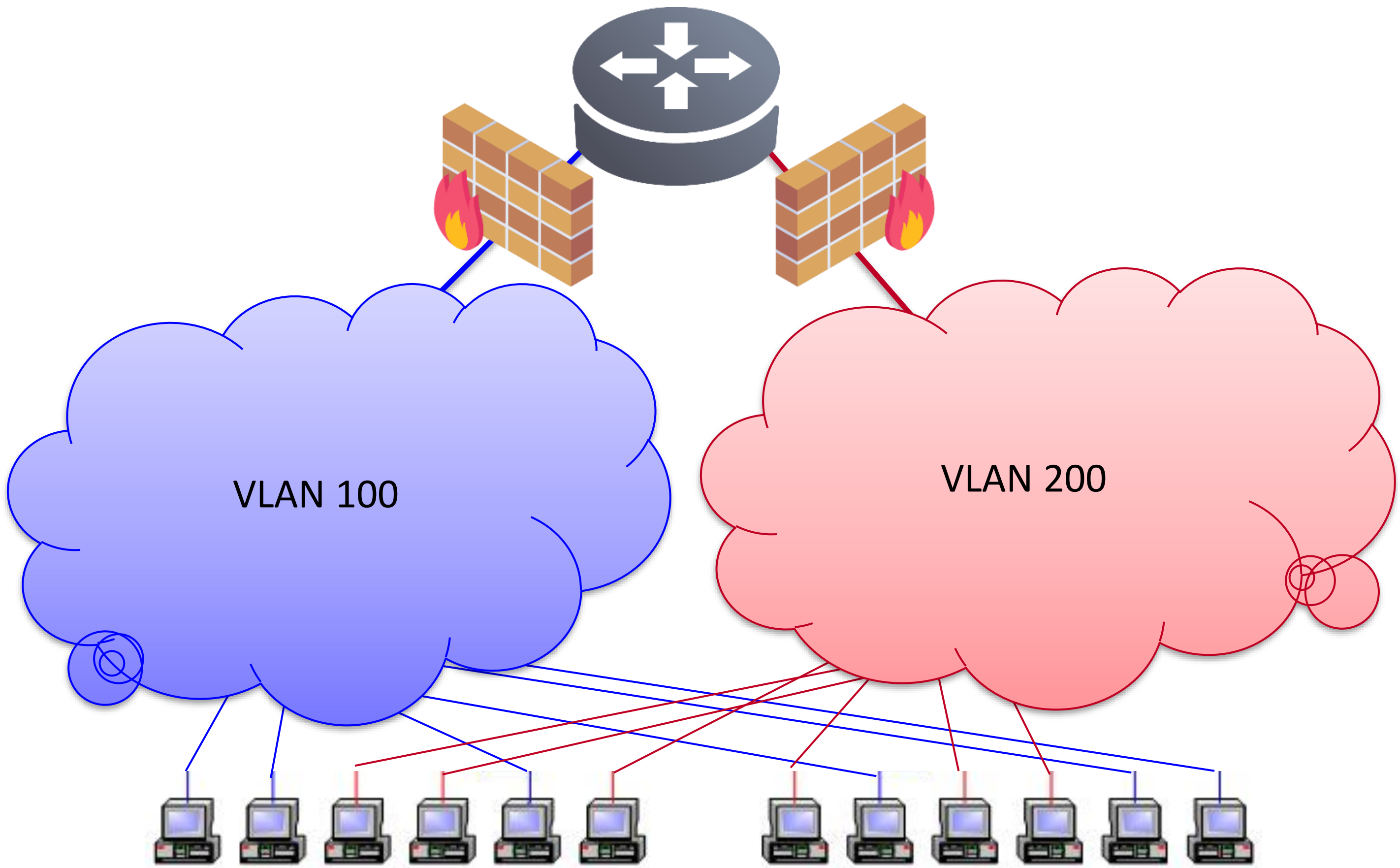
- *Logically* separate layer 2 networks
- Switch ports can be:
  - **Access ports:** can only see one VLAN, aren't aware of VLAN concept
  - **Trunk ports:** end point includes a VLAN tag in packet header to indicate which VLAN it wants to talk to; interprets such headers on incoming packets



http://www.examcollection.com/certification-training/ccnp-configure-and-verify-vlans-and-trunking.html

VLANs make it convenient to have firewall/NIDS/NIPS boundaries

- If two VLANs want to talk, it's via a router; that's a great place to put a firewall!



http://www.examcollection.com/certification-training/ccnp-configure-and-verify-vlans-and-trunking.html

Conclusion

Understanding intruders

- Criminal/activist/state/other
- Skill level

Intrusion detection systems (IDS)

- Look for anomalies or signatures, log/alert accordingly
- Either host-based or network-based

Intrusion prevention system (IPS)

- It’s an IDS but it takes action

Detection theory

- Need false positive rate <= base rate, otherwise most alerts are wrong

Firewalls

- Block traffic based on rules