Practical Network Defense

Master's degree in Cybersecurity 2024-25

Intrusion Detection Systems

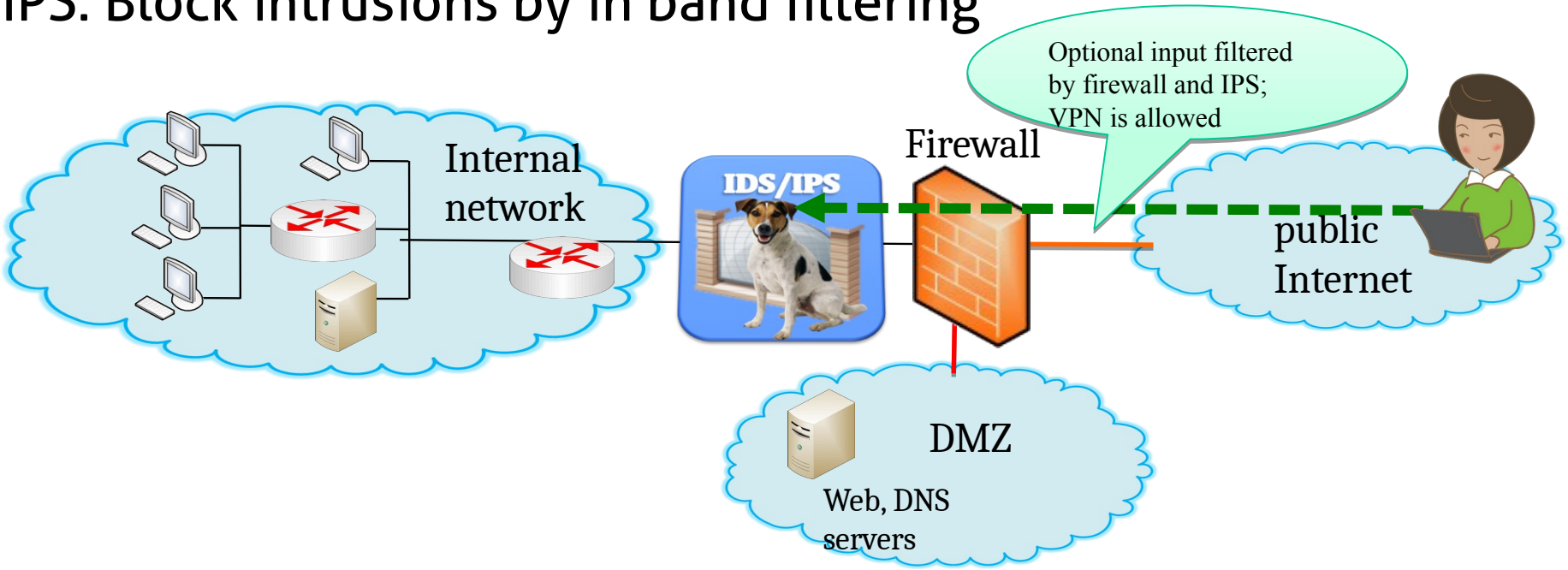
Angelo Spognardi
spognardi@di.uniroma1.it
Dipartimento di Informatica
Sapienza Università di Roma

Intrusion detection/prevention systems

- An Intrusion Detection System (IDS) aims at detecting the presence of intruders before serious damage is done.
- The ultimate purpose of an intruder could be to:
  - Prevent the legitimate users from using the system
  - Reveal confidential information
  - Use the system as a stepping stone to attack other systems
- Second generation IDS are IPS, Intrusion Prevention Systems, also produce **responses** to suspicious activity, for example, by modifying firewall rules or blocking switches ports

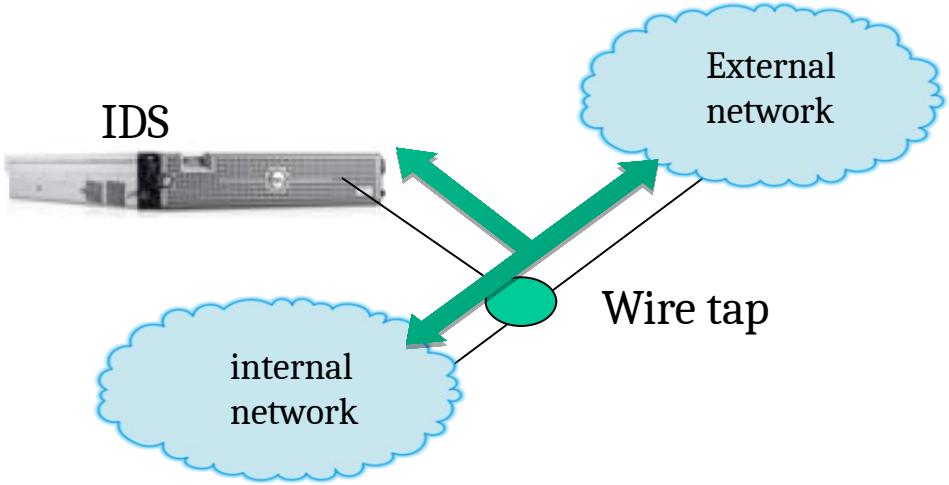
Intrusion Detection/Prevention system (IDS/IPS)

- Deep packet inspection (payload)
- IDS: report intrusions by out of band detection
- IPS: Block intrusions by in band filtering

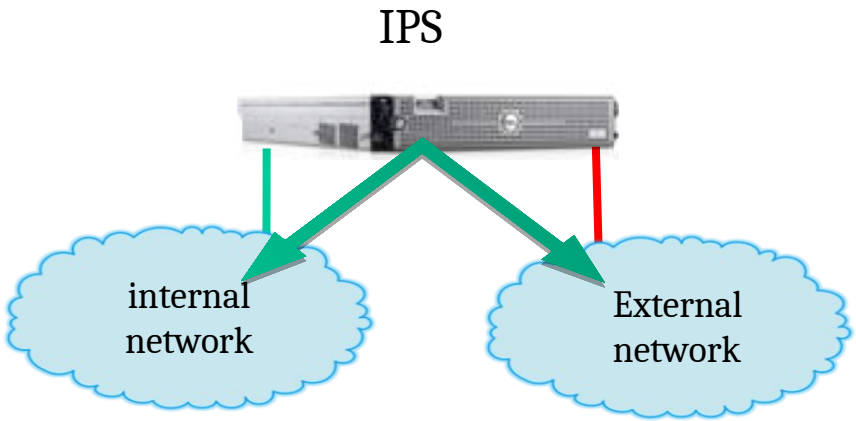


IDS vs. IPS

IDS: out of band



IPS: in line

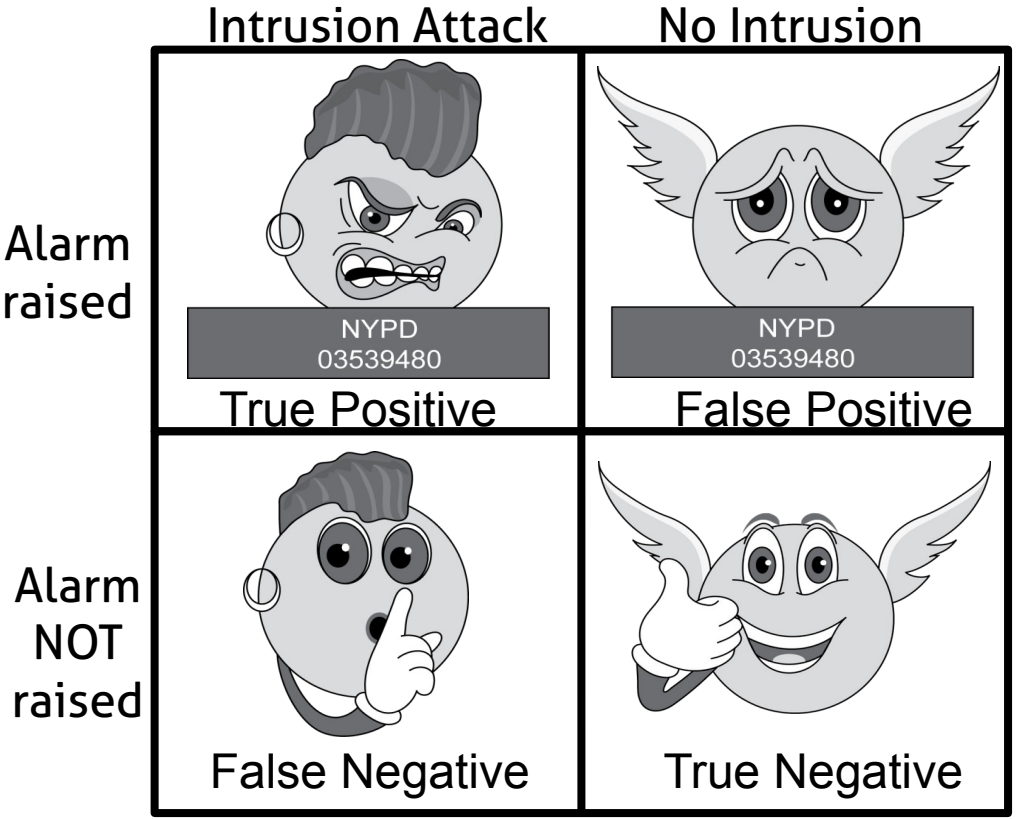


IDS and IPS

- IDS is passive: detects and raises alarms
- IPS is active: detects and reacts
- Typically IPS is placed in-line, able to actively prevent/block intrusions in real time
  - However, functionalities of both have blurred: NIST (National Institute of Standards and Technology) uses the term IDP

Beware to the alarms

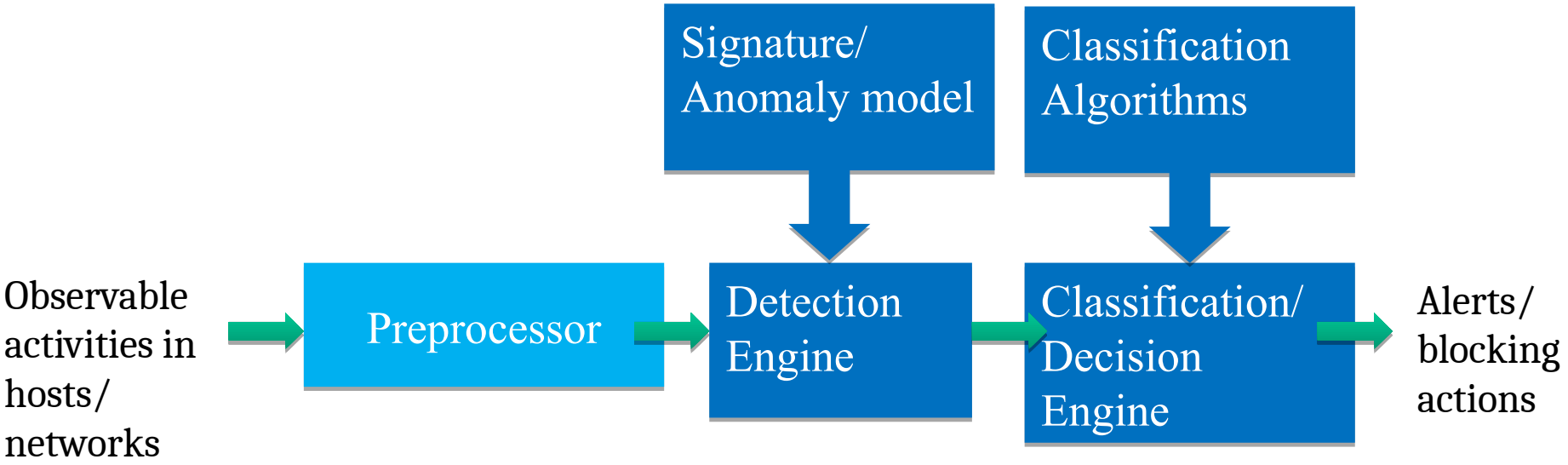
- Alarms can be raised (positive) or not (negative)
- IDS needs to detect a substantial percentage of intrusions with few false alarms
- If too few intrusions detected → no security
- If too many false alarms → ignore



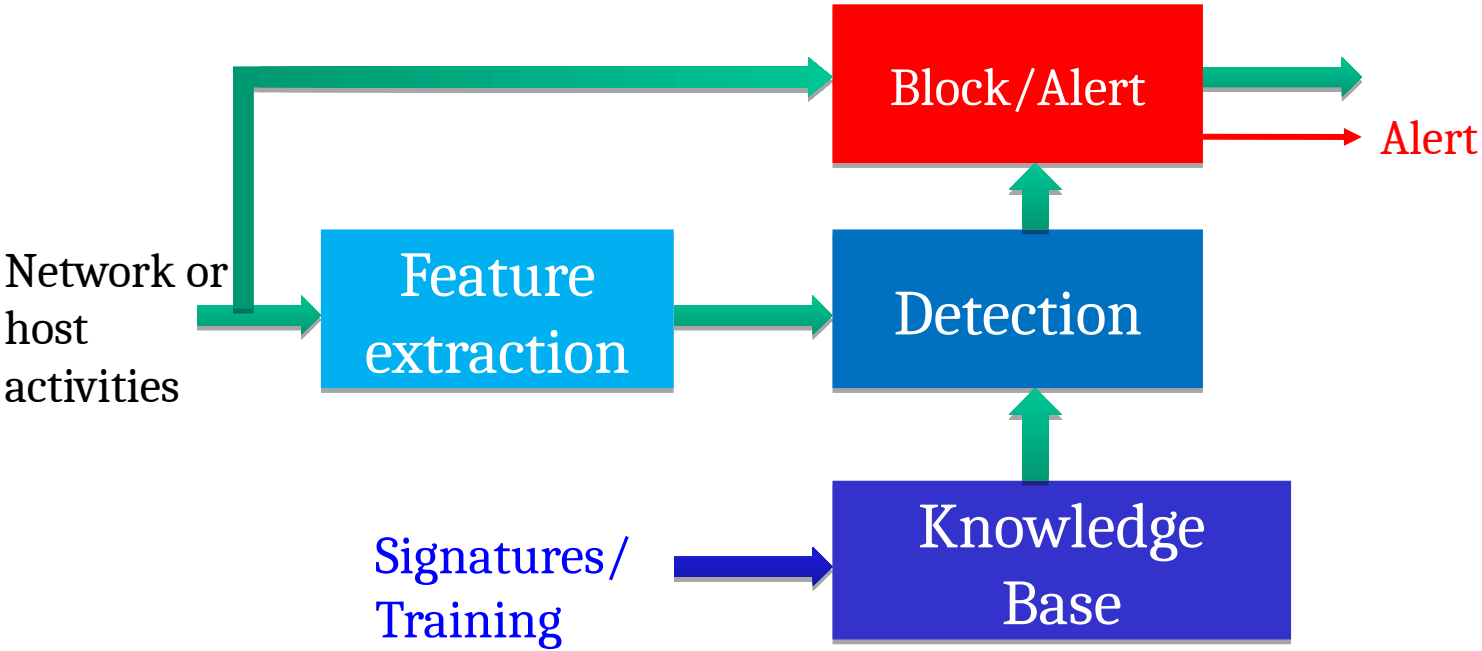
Other functionalities

- Recording information related to observed events.
- Notifying (alert) security administrators of important observed events.
- Producing reports
  - Reports summarize the monitored events or provide details on particular events of interest.
- In case of IPS also changing the network activity
  - Drop connections, block accesses, change configurations of other devices, change of the content of packets (normalization of the requests) and so on

IDS/IPS block diagram



IDS/IPS Function Blocks



Types of IDS

- Host-based (HIDS):
  - Monitors events in a single host to detect suspicious activity.
  - Typically deployed on critical hosts offering public services
  - Advantage: better visibility into behavior of individual applications running on the host
- Network-based (NIDS):
  - Analyses network, transport and application protocol activity
  - Often placed behind a router or firewall that is the entrance of a critical asset
  - Advantage: single NIDS/IPS can protect many hosts and detect global patterns
- Wireless (WIDS):
  - Analyses wireless networking protocol activity (not T- or A-layers)
  - Typically deployed in or near an organization's wireless network

HIDS (Host-based)

- Only monitors traffic on one specific system
  - No promiscuous mode
- It looks for unusual events or patterns that may indicate problems
  - Unauthorized access and activity
  - Unexpected activity
  - Changes in configurations
  - Software changes
    - Ex. Tripwire

NIDS (Network-based)

- Usually operates in promiscuous mode → sniffers
  - Can have multiple NICs to monitor multiple network segments
- Usually connected to switches with ports mirrored (or in SPAN mode, Switch Port ANalizer):
  - All the traffic generated within ALL the ports of the switches are replicated on the mirrored port where the NIDS is placed
- Often it has a series of **sensors** placed in different networks (DMZ, internal net, specific nodes…)
  - Distributed detection system

Other types of IDS

- File integrity monitors (e.g. Tripwire, AFICK)
  - Monitor changes to key system configuration files
- Flow-based IDS (NetFlow)
  - Tracks network connections
  - Establishes patterns of normal traffic
  - Alert when unusual services/patterns/protocols/behaviors seen
  - Can give a good overall situational view on large network(s)
- Hybrid detection capabilities
  - Augment or replace signature-based detection
  - Usually anomaly/behavior-based (pseudo-artificial intelligence)
  - Often require "training" periods to establish a baseline

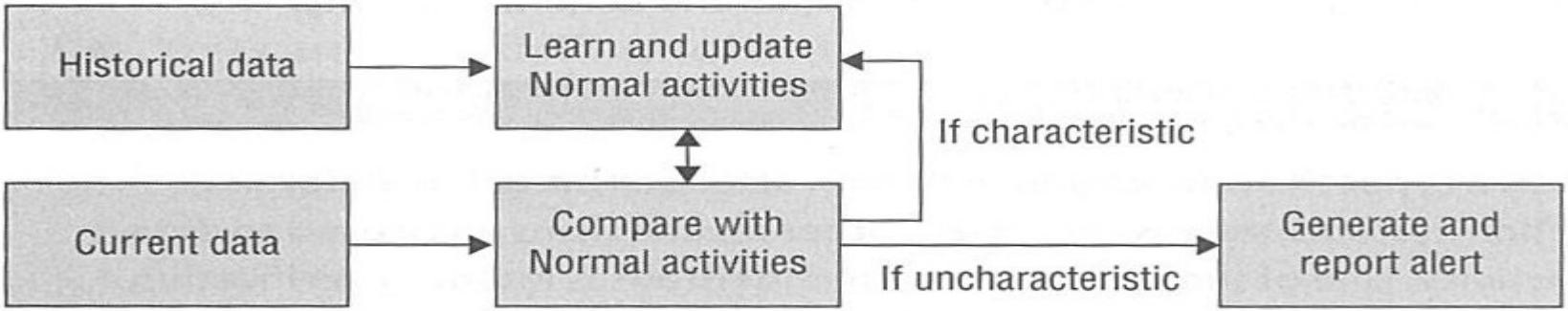
Activities Monitored by IDS/IPS (1)

- Any activity sensitive to occurrences of any events deemed to be **security concerns**
- Attempted and successful breach
  - Reconnaissance
  - Patterns of specific commands in application sessions
    - e.g., a successful remote login session should contain authentication commands
    - Login and location frequency
  - Content types with different fields of application protocols
    - e.g., the password for an application must be 7-bit ASCII of 8 to 64 allowed characters to avoid buffer overflow and SQL injection
  - Network packet patterns between protected servers and the outside world
    - Client application, protocol and port, volume, and duration
    - Rate and burst length distributions for traffic
  - Privilege escalation

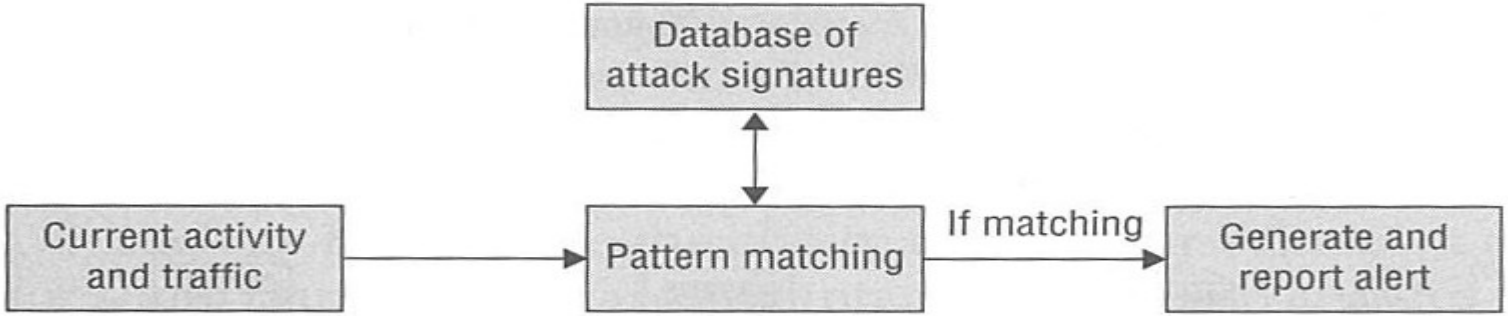
Activities Monitored by IDS/IPS (2)

- Attacks by legitimate users/insiders
  - Illegitimate use of root privileges
  - Unauthorized access to resources and data
  - Command and program execution
    - Mouse, keyboard, CPU, disks, I/O patterns
    - Programs/system calls/processes execution frequencies, resource access (exhaustion), denied executions
    - File/database access activity
    - Read/write/create/delete frequency; records read/written; failed reads, writes, creates, deletes; resource exhaustion
- Denial of service attacks
  - Rate and burst length distributions for all types of traffic
- Malware:
  - Rootkits/Trojans/Spyware
  - Viruses, zombie and worms
  - Scripts
  - Hard to handle mutations
    - Polymorphic and metamorphic viruses: each copy has a different body

IDS approaches



Behavior-based (anomaly detection)



Signature-based (misuse detection)

How to recognize an intrusion

- Behavior-based (anomaly detection):
  - Define behavioral characteristics of normal behavior
  - Compare actual behavior with these. If there are significant differences, **raise an alarm**
  - Difficult to define all possible normal behavior. New activities often give "false positives" (i.e. normal behavior classified as intrusion)
- Signature based (misuse detection):
  - Define characteristics of various types of abnormal activities
  - Compare actual behavior with these. If any of them match, **raise alarm**!
  - Difficult to produce a complete catalog of abnormal activities.
    - If any are missing, there will be "false negatives" (i.e. undetected intrusions)

Learn and classify anomalies

- Behavior is typically described in terms of a set of features
- The feature set should describe all relevant aspects of the behavior to be recognized
- Anomaly detection requires some form of learning (or "training"), usually based on **data mining** in actual observations
- A too large feature set means that both training and classification will take unnecessarily long time
  - Require more observations in order to deal with more features

Example: Lee & Stolfo feature set

- A classic set of features for NIDS anomaly detection
- Constructed by data mining on identified attack patterns in network traffic (collected via tcpdump)
- Features fall into three classes:
  - Intrinsic features: Data about a particular connection or network flow
    - e.g. connection lifetime, amount of data, illegal fragments
  - Traffic features: Statistical information about connections
    - e.g. % of connections to same host with "SYN" errors
  - Content features: Application-related statistics
    - e.g. No. of file creations, failed login attempts.
- Rather out of date by now (published in 2000), but still useful inspiration

Other feature sets

Particular technologies need their own feature sets, e.g.:

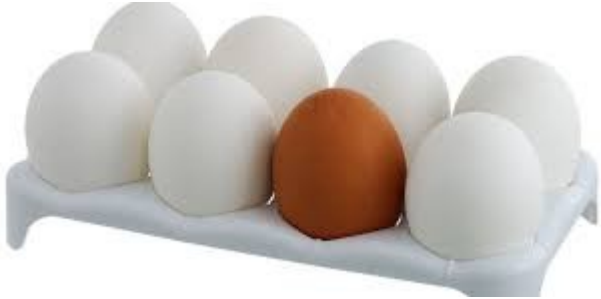
- Wireless networks:
  - Signal power
  - Sequence number "jumps"
  - Round-trip time (RTT)
- Grid/cloud systems:
  - GridFTP connections
  - GridFTP mode of operation
  - Number of GridFTP clients (evidence of "Flash crowds")
  - Traffic entropy
  - Type of LDAP operations

Behavior-based IDS

- Intruders may behave in a different manner from ordinary users or ordinary programs:
  - Many types of attack are characterized by abnormal patterns of OS use or network use.
- Recognizing abnormal behavior enables us to detect attacks as they take place.
- Big questions:
  - How to recognize normal and abnormal behavior patterns?
  - How quickly can recognition take place?
  - How do we deal with abnormally behaving systems?

Behavioral w.r.t. anomalies

- Take sequence of observed behavioral elements (system calls, network packets or others)
- Derive feature values from the behavioral elements
- Derive the "normal" behavior, generated using statistics or with a set of rules (like parameters or procedures) or with a Machine Learning approach
- Compare the traffic against the normal behavior:
  - 1 Distance measures (statistics and thresholds):
    - Hamming distance: How many elements have to be changed?
    - Mahalanobis distance: Generalized distance in n dimensions.
    - Kolmogorov complexity: Difference in information density?
  - 2 Probability measures ("how likely is this sequence?"), e.g.:
    - Markov model
    - Neural Networks
    - Any other ML mechanism
  - 3 Rule sets ("does the sequence follow a set of pre-defined rules?")



Adaptive and self-learning profile

- Adaptive profiles can account for normal network changes to avoid raising false alarms
- Self-learning is critical to ensure wide and successful deployment of anomaly-based detection mechanisms
  - Manually set the profiles is difficult because of the complexity of dynamically changing traffic statistics
- Need to apply anomaly-based detection at various levels of traffic aggregation to achieve the most accurate protection
  - A single server, a server farm, a business division, an enterprise

Signature-based IDS

- Starts from the idea that intruders/attacks may have a characteristic appearance which makes it possible to identify them
- The idea is to screen the PAYLOADS of the packets looking for specific patterns → signatures
- Suppliers of IDSs maintain huge databases of signatures (code or data fragments) which characterize various classes of intruder.
- Rapid recognition involves searching for matches for one or more of the known signatures from a collection of many thousands of signatures

Signature-based (Rule-based) IDS

- Dominant technology in commercial systems
- Rules express actions on given conditions, possibly with complex predicates, including timing, payload content etc etc.

```
Act.   proto src…      dest…                    options…
alert tcp any any -> 10.1.1.0/24 80 (content: "/cgi-bin/phf";)
alert tcp any any -> 10.1.1.0/24 6000:6010 (msg:"X traffic";)
alert tcp any any -> any 21 (msg:"FTP ROOT";content:"USER root";)
```

- Typically supplied ready-made by manufacturer of IDS
- Requires considerable effort by manufacturer to find right rules and distribute them to subscribers as new attack forms are analyses
- A job for real experts! (But easy for the user. . .)
- Rules can in fact be derived from more automatic IDS technologies (Markov, ANN, clustering, etc.)! This is an active area for research.

Signature-based IDS principles

- A packet sniffer "on steroids"
- Captures the packets in a LAN and applies some fairly complex logic to decide whether an intrusion has taken place
  - SNORT is one of the best known intrusion detectors
    - Easy-to-learn and easy-to-use rule language for intrusion detection.
    - The rules are stored in **/etc/snort/rules** directory
- Con: can not inspect encrypted traffic (VPNs, SSL)
- Con: not all attacks arrive from the network
- Con: record and process huge amount of traffic

IDS detection capability

- Decode packets, namely DPI: deep packet inspection
- Decode application and protocol headers to look at high-layer activity → the payload containing the application protocol
- Protocol decoding to detect anomalies

Misuse detection

- Set of rules defining a behavioral signature likely to be associated with attack of a certain type
- Example: buffer overflow
  - A setuid program spawns a shell with certain arguments
  - A network packet has lots of NOPs in it
  - A very long argument to a string function
- Example: SYN flooding (denial of service)
  - Large number of SYN packets without ACKs coming back
  - ...or is this simply a poor network connection?
- Attack signatures are usually very specific and may miss variants of known attacks
  - Why not make signatures more general?

Extract misuse signatures

- Use invariant characteristics of known attacks
  - Bodies of known viruses and worms, port numbers of applications with known buffer overflows, RET addresses of stack overflow exploits
  - Hard to handle malware mutations
    - Metamorphic viruses: each copy has a different body
- Challenge: fast, automatic extraction of signatures of new attacks
- **Honeypots** are useful for signature extraction
  - Try to attract malicious activity, be an early target

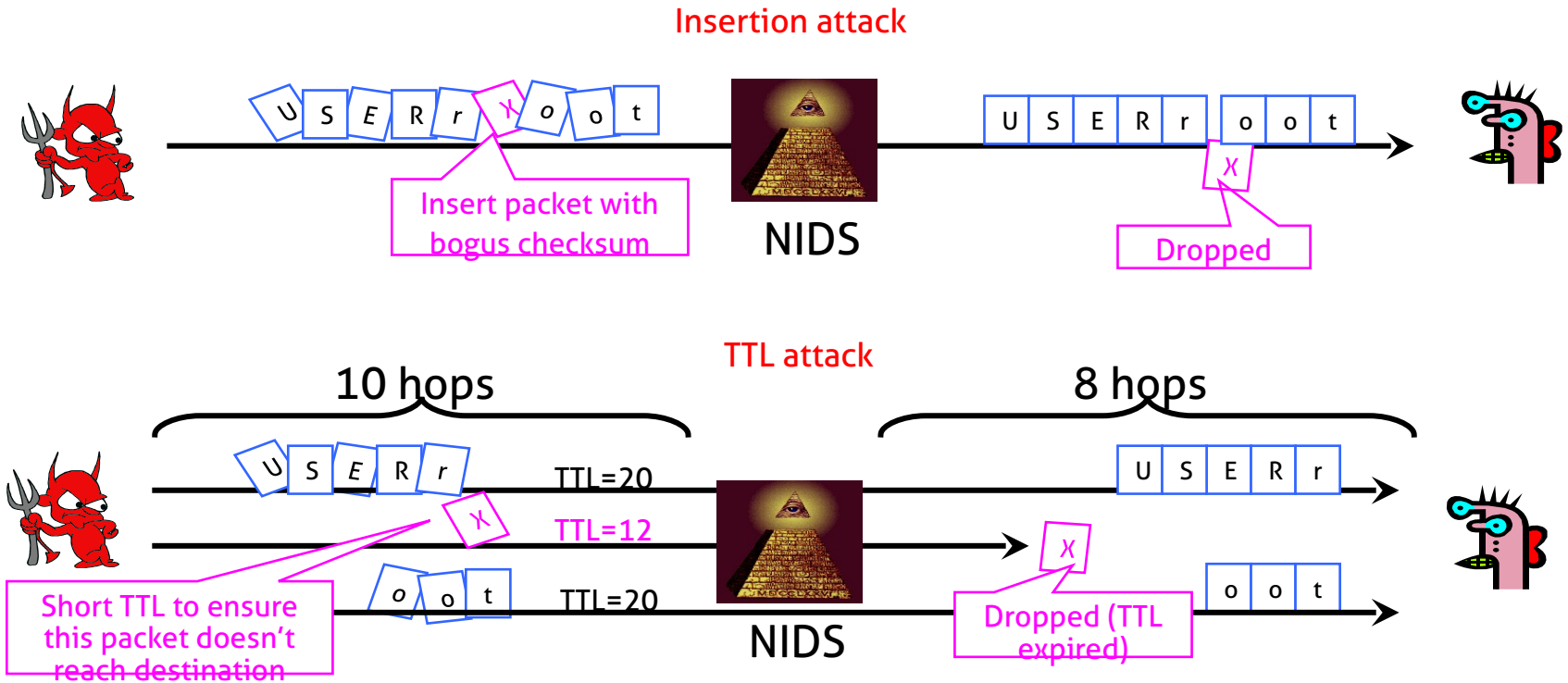
Honeypot

- Definition:
  - A security resource whose value lies in it being attacked, probed or compromised.
- A **honeypot** is (usually) a single computer, whereas
- A **honeynet** is a network of computers, usually protected by a firewall to regulate traffic.
- The idea is to attract the attackers

Example of IDS rule evasion

- Want to detect "USER root" in packet stream
- Scanning for it in every packet is not enough
  - Attacker can split attack string into several packets; this will defeat stateless NIDS
- Recording previous packet's text is not enough
  - Attacker can send packets out of order
- Full reassembly of TCP state is not enough
  - Attacker can use TCP tricks so that certain packets are seen by NIDS but dropped by the receiving application
    - Manipulate checksums, TTL (time-to-live), fragmentation

TCP Attacks on NIDS



Behavior-based detection techniques: protocol anomaly

- Layers 2-7 inspection by rules
- A protocol or service for a non-standard purpose or on a non-standard port
  - e.g., modified protocols for tunneling through firewalls (e.g., P2P on port 80)
  - Port scan
- Full IP defragmentation, and TCP reassembly

- Deep packet inspection
  - IP fragmentation overlaps and suspicious IP options
  - Unusual TCP segmentation overlaps and illegal TCP options and usage
  - Deep application protocol parsing/decoding
    - Illegal field values and combinations
    - Illegal command usage
    - Unusually long or short field lengths, which can indicate buffer overflow
    - Very long argument to a string function
    - Unusual number of occurrences of particular fields/commands
  - Application semantics
    - Type of encoding is legal for a given field
    - Applications can be embedded within it
    - Application level anomaly: shellcode in unexpected fields

Behavior-based detection techniques: statistical anomaly

- Statistical measures are used to capture network traffic behavior
  - A stable balance among different types of TCP packets in the absence of attacks
  - 3-way handshake, 4-way close, and data transfer
- This balance can be learned and compared against short-term observations that will be affected by attack events
  - Profiles based on statistical measures of time-of-the-day, day-of-the-week variations in traffic volume
  - Profiles for traffic rate distributions on a multi-week scale normal network environments
  - Profiles based on statistical measures could raise DDoS anomalies based on rare events of the difference between the long- and short-term distributions or based on a rare occurrence of long bursts of high-rate traffic
- Sensitivity level: can be adjusted to different levels of profile deviation
  - e.g., low sensitivity: high traffic profile deviation can trigger a DDoS alarm

Signature vs. Behavior

- Signature-based detection
  - Clearly indicates the detected attack method
- Behavioral-based alerts
  - Indicate:
    - The attack type
    - The behavioral rule that was violated, such as port scan
    - The statistical profile that was violated
- Behavioral protection can not identify the specific attack or exploit that was blocked
  - Require security administrator to investigate clues given by the behavioral rule to determine which attack was actually attempted
  - Acceptable for new and unknown attacks
  - But established threats and known exploits should be easily identifiable
- Deciphering information about each attack reported by a behavioral-only system becomes unmanageable for a large number of hosts

Combine behavior and signature

- False negatives: attack is not detected
  - Problem in signature-based misuse detection
- False positives: harmless behavior is classified as an attack
  - Problem in statistical anomaly detection
- Signature "can not" detect:
  - DoS/DDoS
  - Zero-day exploits
  - Protocol/application anomaly
- Best solution is to combine both signatures and behavioral rules
  - No false positives: Do not ever block legitimate traffic under any circumstances
  - No false negatives: Do not miss attacks, even when the attacker intentionally tries to evade detection

Combined Behavior and Signature Detection

- Detection Correlation:
  - Signature, Anomaly, and Denial of Service detection functionality
  - Interdependence and cross-checking of suspicious traffic
  - Behavioral protection can block zero-day attacks without updates to the system
- Once an exploit has been recognized using behavior-based technique, a stateful signature can be created to provide accurate detection and save manpower
  - Lower the false positive rates
  - Reduce the response time to attacks
- Most anti-virus, anti-spyware, firewall products are integrated with both Behavior and Signature intrusion detection

IDS vs. IPS

- IDS: out of band
  - An IDS false positive is an alert that did not result in an intrusion
  - The system under attack was not vulnerable to the attack
  - The detection mechanism may be faulty
  - IDS detected an anomaly that turned out to be benign
- An IDS false positive causes a security analyst to expend unnecessary effort
  - Minimize false positives
- No interference with traffic

- IPS: in band
  - When an IPS has a false positive, legitimate traffic will be blocked
  - IPS cannot have false positives
  - Better development for filters and thorough tests
- To match the line speed, IPS hardware requirement is higher
  - ASIC or FPGA

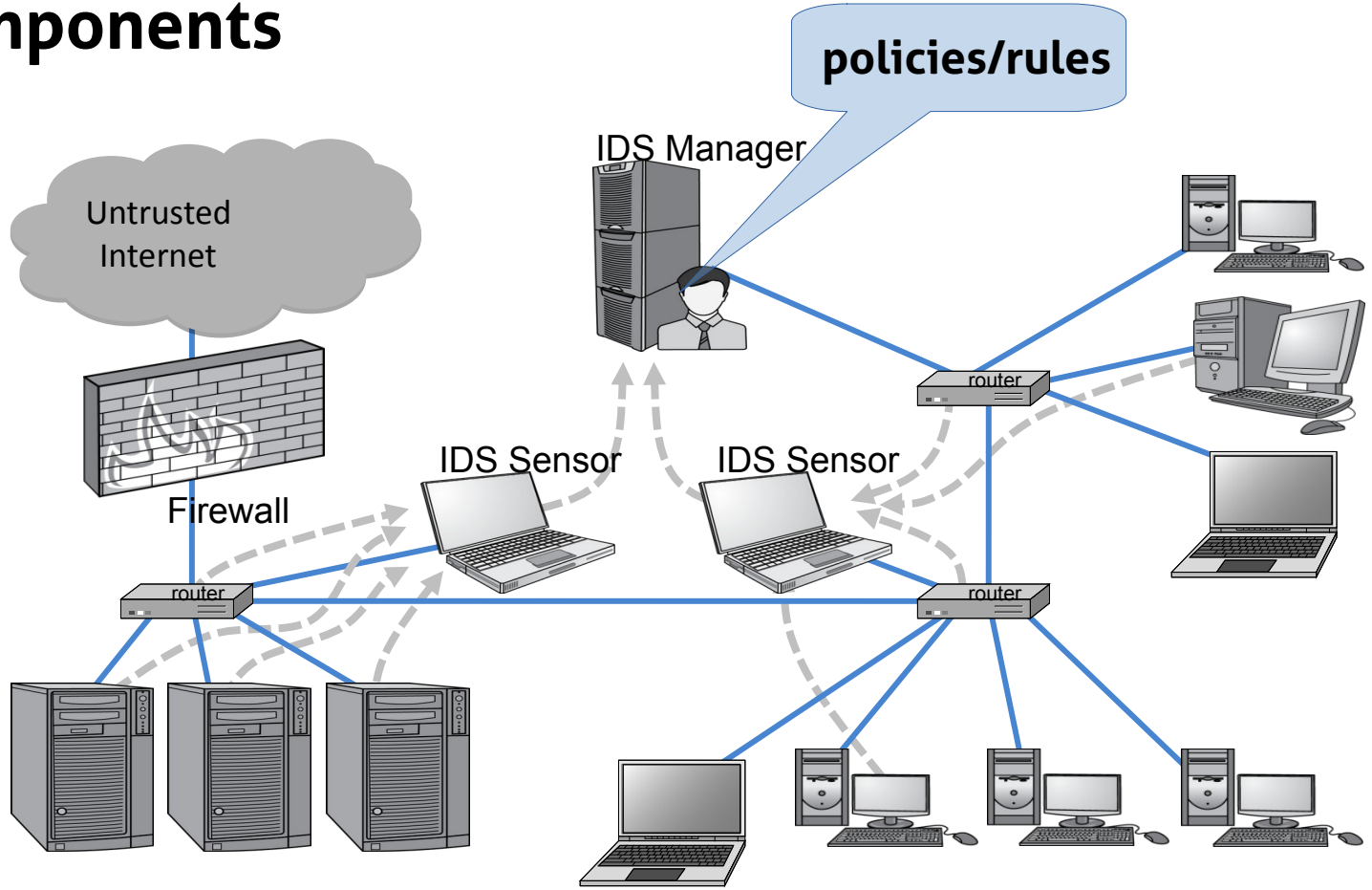
Observations

- Legitimate traffic in real networks contains anomalies
  - Protocol anomalies come from custom applications that use off-the-shelf protocol libraries, but use them in unexpected ways
  - Behavioral anomalies come from exceptional, but often critical, business processes
- IDS filters create leads on suspicious activity intended for an expert to follow
- IPS filters are used for automatic action such as blocking traffic or quarantining an endpoint
- Anomaly-based detection mechanisms (both protocol and statistical) are useful for IDS, but inappropriate for IPS
  - Anomaly filters can not be used for blocking, only for alerting

IDS architecture

- Usually several parts:
  - Network sensors: detect and send data to the system
  - Central monitoring system: a server that processes and analyzes data sent from sensors
  - Database and storage component: repository for event information

IDS Components



Host IDS/IPS

- Many host security products have integrated HIPS, anti-malware and firewall
  - Protects mobile hosts from attack when attached outside the protected network
  - Protects against local attacks from a user, and codes/scripts from removable devices
  - Protects against attacks from the same subnet/VLAN
  - Protects against encrypted attacks where the encrypted data stream terminates at the host being protected
  - Inspect packet content after decrypting received VPN or SSL packets
  - Inspect packet together with anti-malware software by decrypting or emulating malware
- Con: if an attacker takes over a host, then one can tamper with IDS/agent binaries and modify audit logs
- Con: only local view of the attack
- Con: Host-based anomaly detection has high false alarm rate

Host IDS/IPS evolved in EDR

- EDR: Endpoint Detection and Response
- Integrated toolkit of software (HIPS, anti-malware, firewall…) that continuously monitors a device to detect and respond to cyber threats
  - Inspect packet content after decrypting received VPN or SSL packets
  - Inspect packet together with anti-malware software by decrypting or emulating malware
  - Inspect any resource used by the endpoint (user accounts, changes to ASP keys, executables and administrative tool usage, process executions, process-level network activity, including DNS requests, connections, and open ports, archive file creation, removable media usage, and so on)
  - Exploits threat detection/response capabilities of the vendor's global threat intelligence database, which is further enhanced with machine learning capabilities
    - Example: Microsoft Defender Advanced Threat Protection (ATP) which is powered by the Microsoft Cloud

Network IDS/IPS

- Deploying sensors at strategic locations with a central monitor
  - Inspecting network traffic
    - Watch for violations of protocols and unusual connection patterns
  - Protect network equipment, such as printers that do not have HIDS
  - Protect against network-oriented attacks
    - DDoS, bandwidth consumption
    - Independent of host OS
  - Monitoring user activities
    - Look into the data portions of the packets for malicious command sequences
- Con: may not detect encrypted traffic
  - Data portions and some header information can be encrypted
- Con: can not detect some attacks in the host
- Con: high requirement for computation capability of IDS/IPS

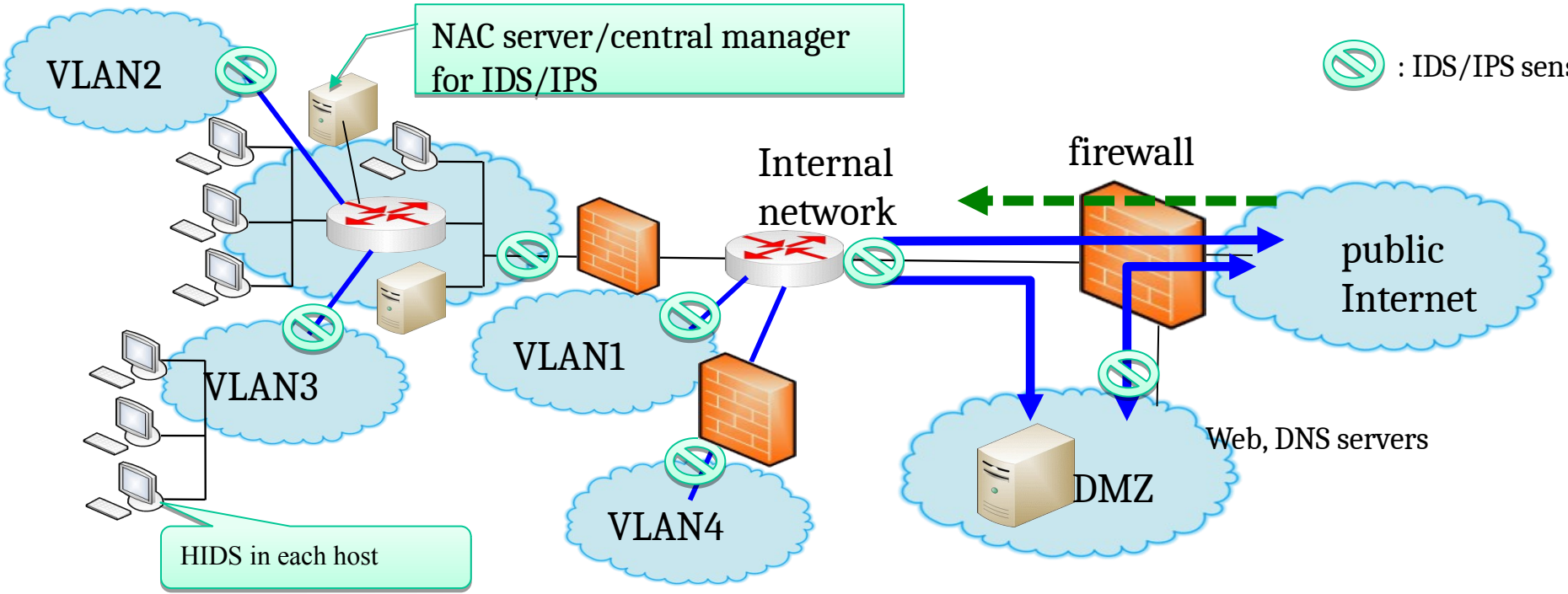
Then: combine host and network IDS/IPS

- Both HIDS and NIDS technologies are not equally adept at detecting and blocking certain attacks
- There are attacks that can only be detected by HIDS
  - E.g., local privilege escalation, metamorphic malware
- Attacks that can only be detected by NIDS
  - E.g.. Routing advertisement injection
- Integrating the strengths of both architectures provides a solution whose sum is greater than its parts
- More accurate result for quarantining a host or block/filter traffic
- The basis for NAC (Network Access Control) products

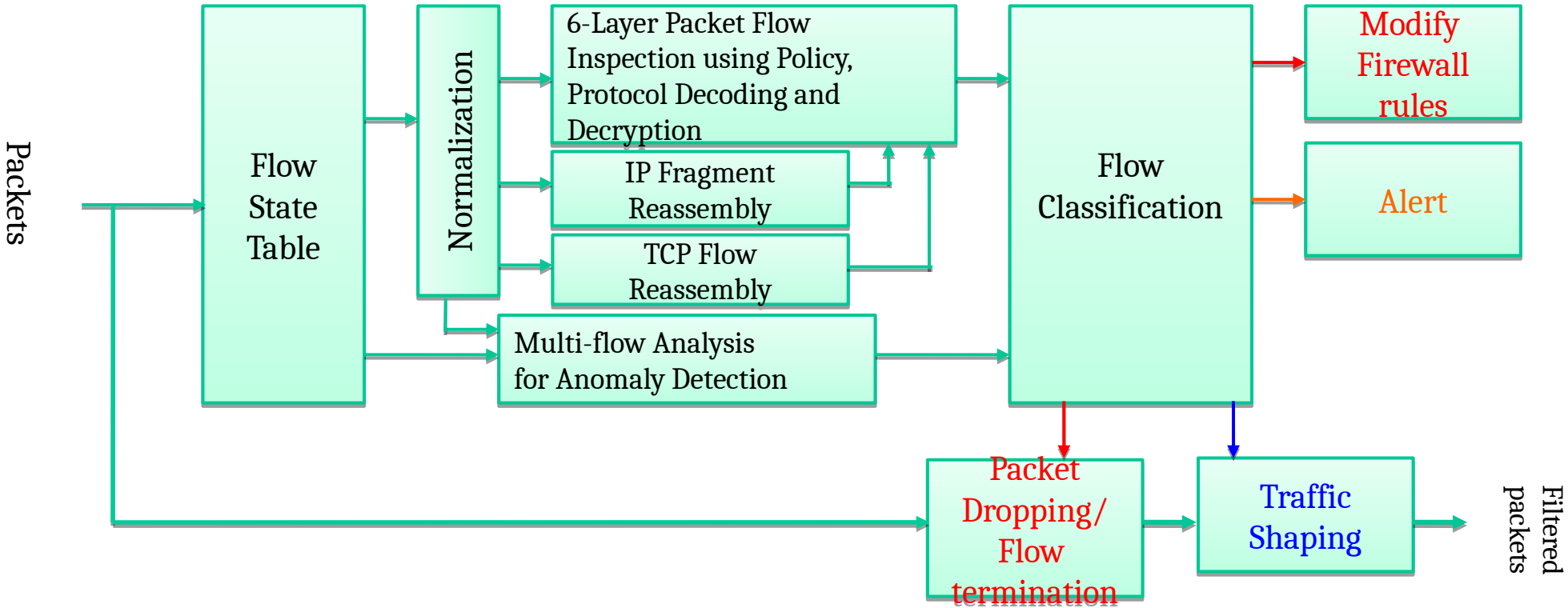
Distributed IDS

- Extend focus from single systems to information infrastructure
  - More effective defense has these working together to detect intrusions
  - Agent-based coordination between host and NAC server
- Monitoring and correlating public, internal VLANs, and DMZ segments of the IDS/IPS sensors and firewalls
- Correlation among these segments to yield an accurate picture of network attacks that were either blocked or made it into the internal network
- Correlate HIDS and NIDS for constant monitoring and blocking in NAC (central control)
- Exchange Format Working Group (IDWG) of the IETF
- Intrusion Detection Exchange Protocol (IDXP): RFC 4767
  - An application-level protocol for exchanging data between IDS's
  - IDXP supports mutual-authentication, integrity, and confidentiality over a connection-oriented protocol
  - The protocol provides for the exchange of the Intrusion Detection Message Exchange Format (IDMEF) messages in implementations of the data model in the Extensible Markup Language (XML)
  - The IDMEF message elements are described in RFC 4765, and developed by the Intrusion Detection

Distributed intrusion detection



Network-based IPS block diagram



State information and analysis

- State information for a session (flow):
  - Maintaining state information enables sensors to gain context for attack detection
  - Inspecting the entire content of the data packet
- State information is captured and updated in real time
- State information is the basis for Layer 2-7 detection
  - Utilize multiple token matches to capture attack signatures/behaviors that span packet boundaries or are out-of-order in a packet stream
  - Detect/block malware, Trojans, key loggers, P2P, botnets, worms
- Appropriate use of the state information is the key to detection
  - Accuracy depends on the selection of parameters and their transitions

Normalization

TCP normalization

- Inspect invalid or suspect conditions
  - E.g., a SYN sent to the client from the server or a SYNACK sent to the server from the client
- Block certain types of network attacks
  - E.g., insertion attacks and evasion attacks
    - Insertion attacks occur when the inspection module accepts a packet that the end system rejects
    - Evasion attacks occur when the inspection module rejects a packet while the end system accepts it
- Discards segments containing
  - Bad segment checksum
  - Bad TCP header or payload length
  - Suspect TCP flags (for example, NULL, SYN/FIN, or FIN/URG)
- To configure TCP normalization
  - Assemble various TCP commands into a parameter map for filtering as policy
  - E.g., parameter map contains ranges for MSS, # of SYN retries, # of out of order segments, control of timeout, random sequence number, Window scale factor, urgent flag, etc

IP normalization

- Inspect IP packets using configured parameter map for:
  - General security checks
  - ICMP security checks
  - Fragmentation security checks
  - IP fragment reassembly
  - IP fragmentation if a packet exceeds the outbound maximum transmission unit (MTU)
- Configure IP normalization parameter map
  - ToS
  - TTL
  - Unicast reverse path
  - Fragment reassembly
  - Maxiumum # of fragments
  - MTU

Actions from IPS

- Packet dropping
- Session termination
- Firewall rules modification for blocking suspicious hosts
- Traffic shaping for slowing down less critical traffic such as P2P, video
- Alerts generation
- Log generation

That's all for today

- **Questions?**
- See you next lecture!
- References:
  - NIST Guide to Intrusion Detection and Prevention Systems (IDPS)
  - Chapter 19 textbook
- Other interesting readings:
  - A Framework for Constructing Features and Models for Intrusion Detection Systems
  - Specification-based anomaly detection: a new approach for detecting network intrusions