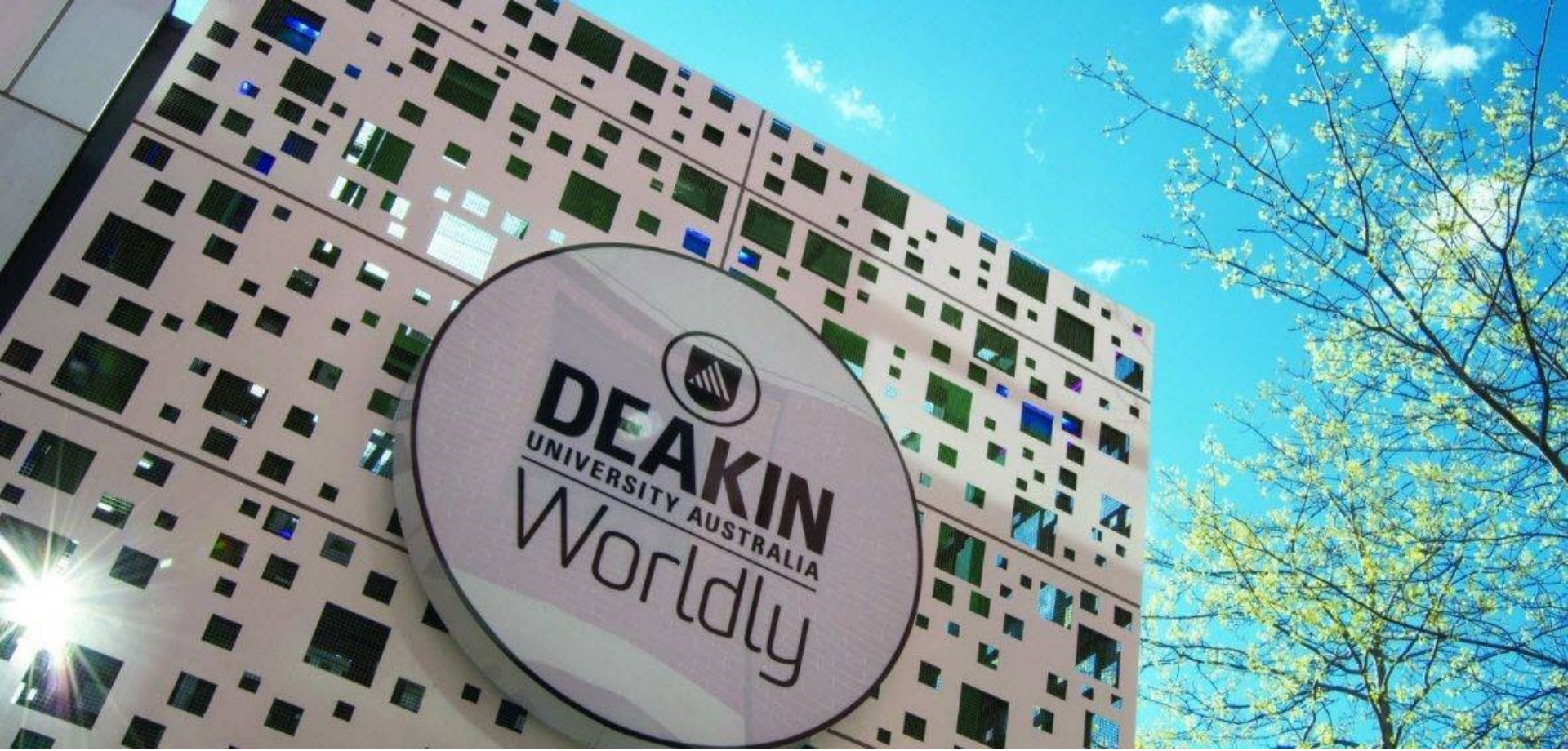


SIT382 SYSTEM SECURITY

Week 04

Intrusion Detection, Firewall and Prevention

Last lecture – DoS & DDoS

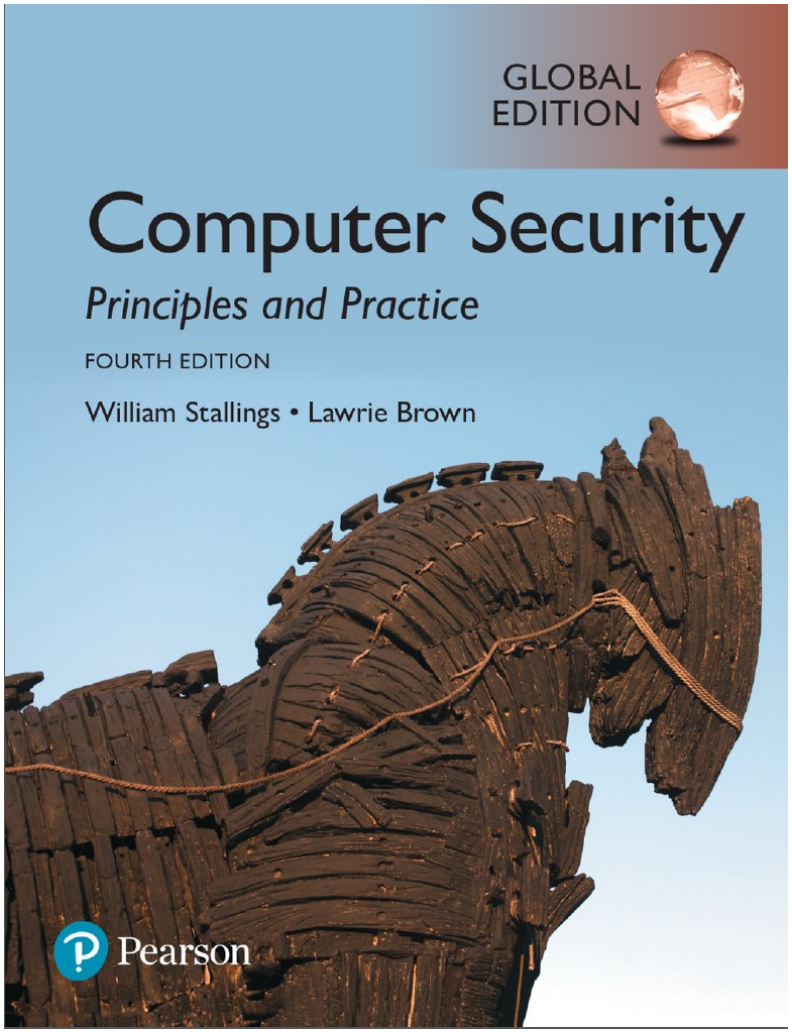
- Introduced denial of service (DoS) attacks
- Classic flooding and SYN spoofing attacks
- ICMP, UDP, TCP SYN floods
- Distributed denial of service (DDoS) attacks
- Reflection and amplification attacks
- Defenses against DoS attacks
- Responding to DoS attacks

Outline

- Intrusion detection
  - Basic principles
  - Analysis approaches
  - Host-based, network-based, distributed host-based
  - Cyber-deception
- Firewalls
- Intrusion prevention systems

Outline

- Intrusion detection
  - Basic principles
  - Analysis approaches
  - Host-based, network-based, and distributed intrusion detection
  - Cyberdeception
- Firewalls
- Intrusion prevention systems



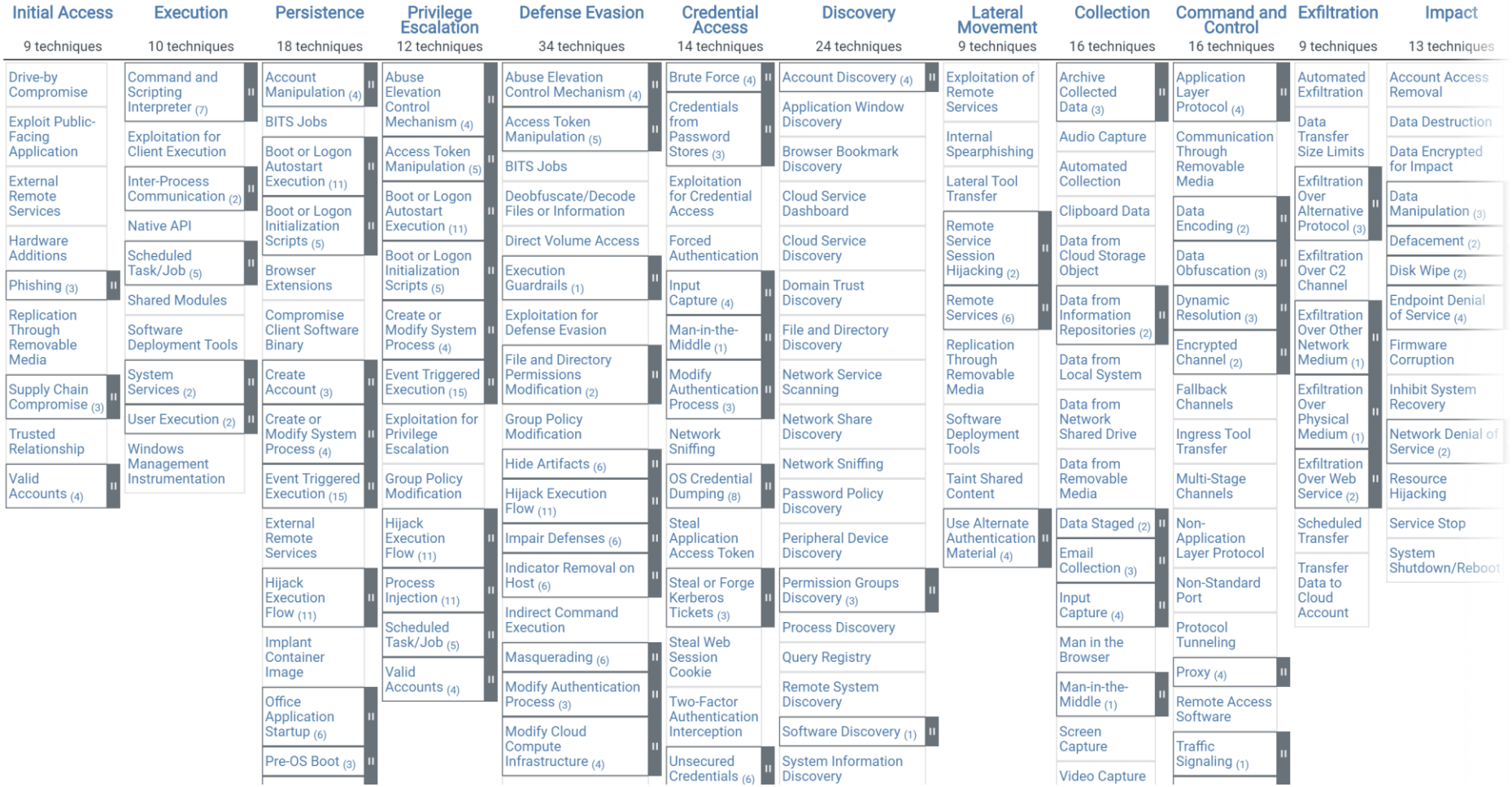
Learning objectives

- Understand basic principles of intrusion detection
- Discuss key features of host-based and network-based intrusion detection
- Explain concepts of distributed host-based intrusion detection
- Explain concepts of cyberdeception

Intruder behavior

- Target acquisition and information gathering
  - Identify the target systems using publicly available information
- Initial access
  - Exploit vulnerability
  - Guess weak authentication credentials
  - Install malware using social engineering or drive-by download
- Privilege escalation
  - Increase privileges, typically via a local access vulnerability
- Information gathering or system exploit
  - Access or modify information
  - Navigate to other systems
- Maintain access
  - Install backdoors or other malware
  - Add covert authentication credentials or configuration changes to the system to enable continued access
- Cover tracks
  - Disable or edit audit logs to remove evidence of attack activity
  - Use rootkits to hide installed files or code

MTIRE ATT&CK



MITRE ATT&CK Matrix for Enterprise (MITRE ATT&CK, n.d.)

Security intrusion & detection

- **Security intrusion**: “a security event, or combination of multiple security events, that constitutes a security incident in which an intruder ***gains, or attempts to gain***, access to a system (or system resource) without having authorization to do so” (“RFC 4949”, 2007)
- **Intrusion detection**: “sensing and analysing system events for the purpose of noticing (i.e. becoming aware of) attempts to access system resources in an unauthorised manner” (“RFC 4949”, 2007)

Components of an intrusion detection system (IDS)

- Sensors
  - Collect data
  - Input for a sensor may be any part of a system that could contain evidence of an intrusion
- Analyser
  - Receive input from one or more sensors
  - Determine if an intrusion has occurred
  - Provide guidance on what actions to take
- User interface
  - View output of the system
  - Control behaviour of the system

Classification of IDS

- Host-based IDS (HIDS)
  - Monitor single host activity
- Network-based IDS (NIDS)
  - Monitor network traffic
- Distributed or hybrid
  - Combine information from a number of sensors, often both host and network based, in a central analyser to better identify and respond to intrusion activity

Basic principles of IDS

- Assumption
  - Intruder behaviour differs from legitimate users
- Problems
  - False positives
    - Valid user as intruder
  - False negatives
    - Intruder not identified

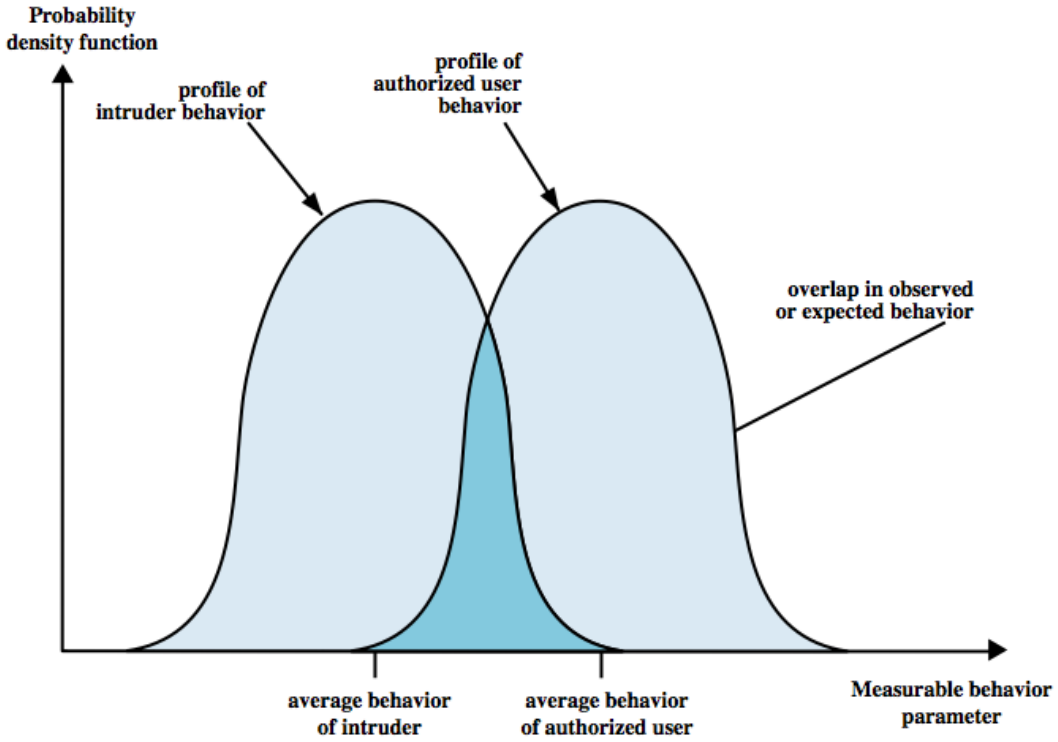
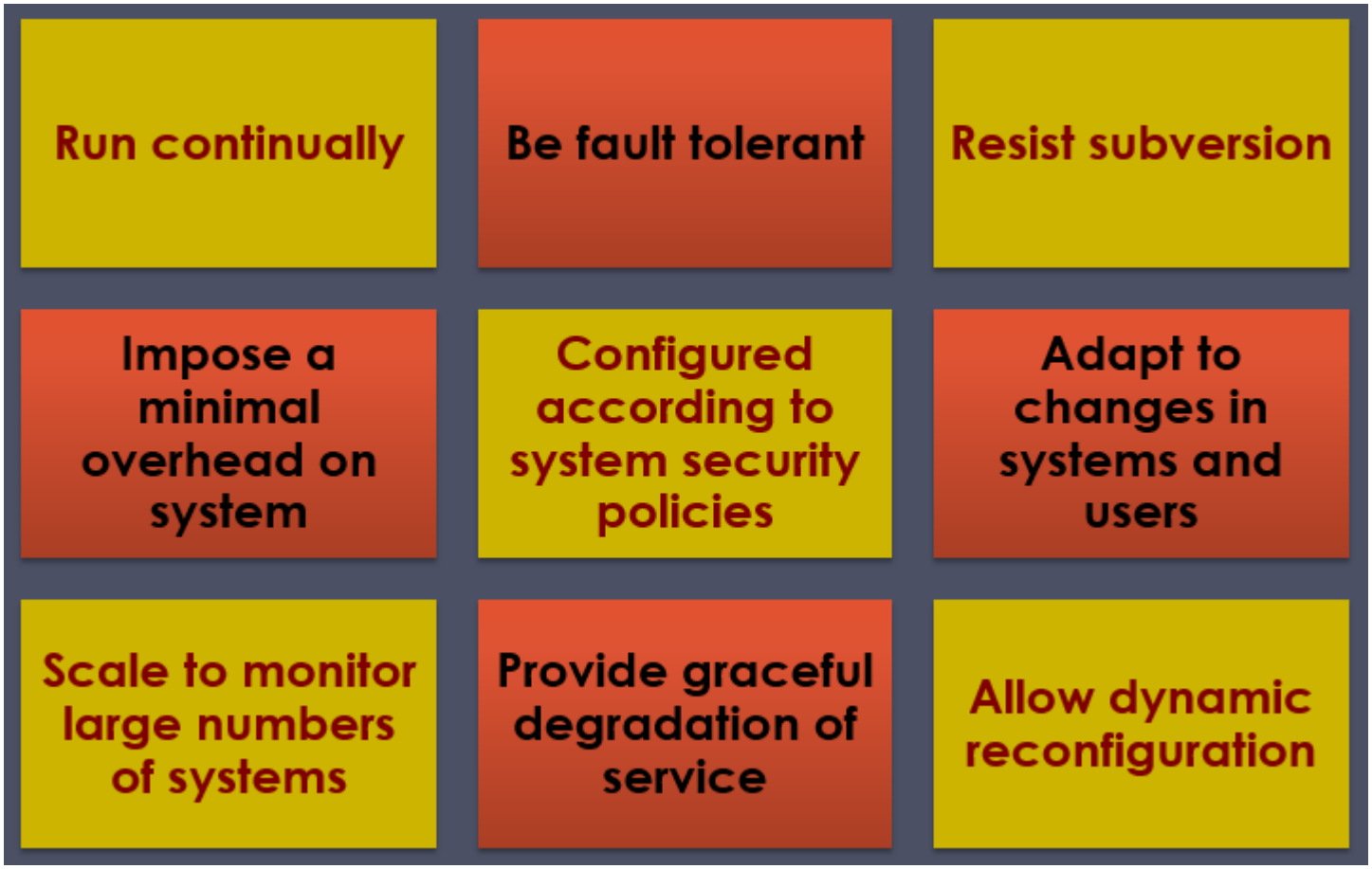


Figure 8.1 (Stallings et al., 2017)

IDS requirements



Analysis approaches - anomaly detection

- Definition
  - Collection of data relating to the behavior of legitimate users over a period of time
  - Current observed behaviour is analysed to determine whether this behavior is that of a legitimate user or that of an intruder
- Classification of approaches
  - Statistical
    - Develop a statistic profile of observed metrics
    - Low computation cost and lack of assumption about expected behaviours
    - Difficult to select suitable metrics
  - Knowledge-based
    - Classify observed data using a set of rules that model legitimate behaviours
    - Robust and flexible
    - Difficult to develop high-profile knowledge and need for human experts
  - Machine learning based
    - Automatically develop a model using labelled data
    - Require significant time and computational resources; high false alarms
    - Flexible and able to capture interdependencies among observed metrics

Analysis approaches - signature/heuristic detection

- Signature
  - Use a set of known patterns of malicious data
  - Widely used in anti-virus products
- Rule-based heuristic identification
  - Use rules to identify known attacks
  - Example of rule-based NIDS: Snort IDS
- Limitation
  - Can only identify known attacks for which it has patterns or rules

Host-based IDS (HIDS)

- Specialised software to monitor activity on the system to detect suspicious behaviour
- Primary purpose
  - Detect intrusions, log suspicious events, and send alerts
  - Can detect both external and internal intrusions
- Common data sources
  - System call traces
  - Audit (log file) records
  - File integrity checksums
  - Registry access

Anomaly-based HIDS

- Mostly on UNIX and Linux systems
  - Data sources: audit records, mainly system call traces
- Windows
  - Data sources: audit log entries, registry file, dynamic link library (DLL) function calls
- Other data sources
  - Important files
  - Tripwire
    - Free software security and data integrity tool for monitoring and alerting on specific file change(s) on a range of systems

Signature or heuristic HIDS

- Approaches
  - Build a database of signatures (patterns of data in malware)
  - Create heuristic rules (known malicious behaviour)
- Limitation
  - Can not detect zero-day attacks

Distributed HIDS

- Challenges
  - Different data formats
  - Integrity and confidentiality of transmitted data
  - Single point of failure using centralised architecture

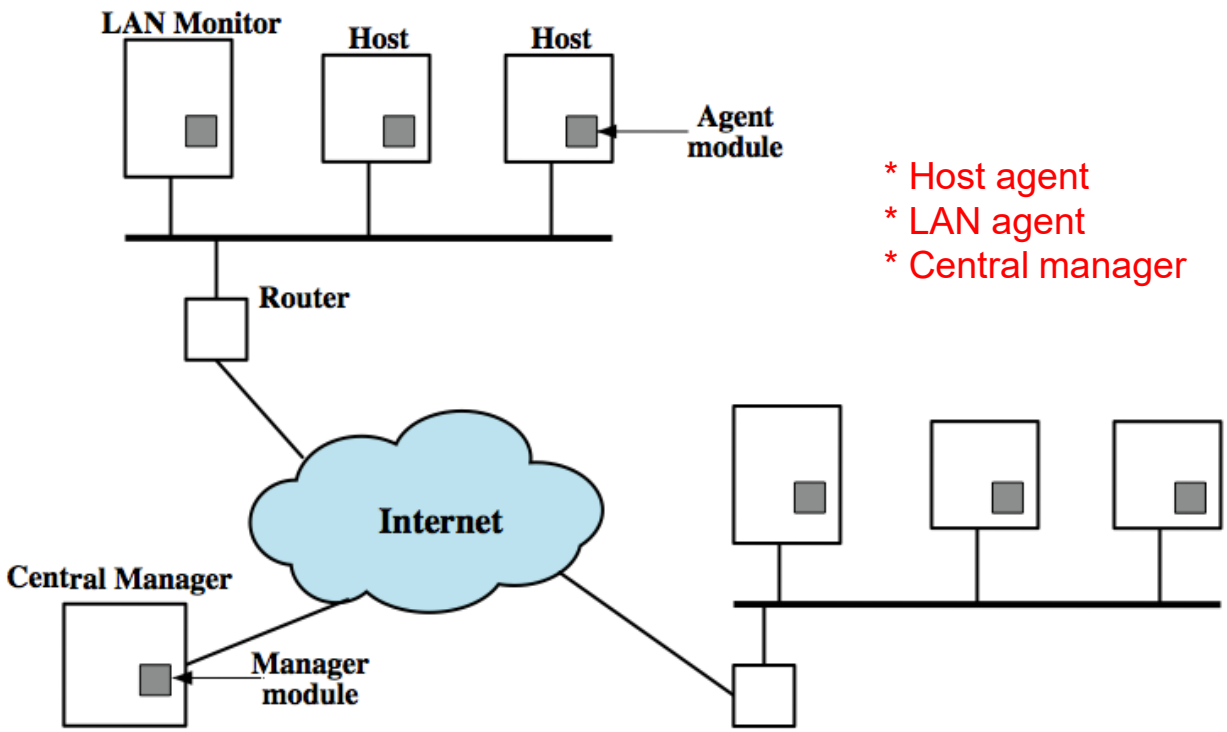


Figure 8.2 (Stallings et al., 2017)

Agent architecture

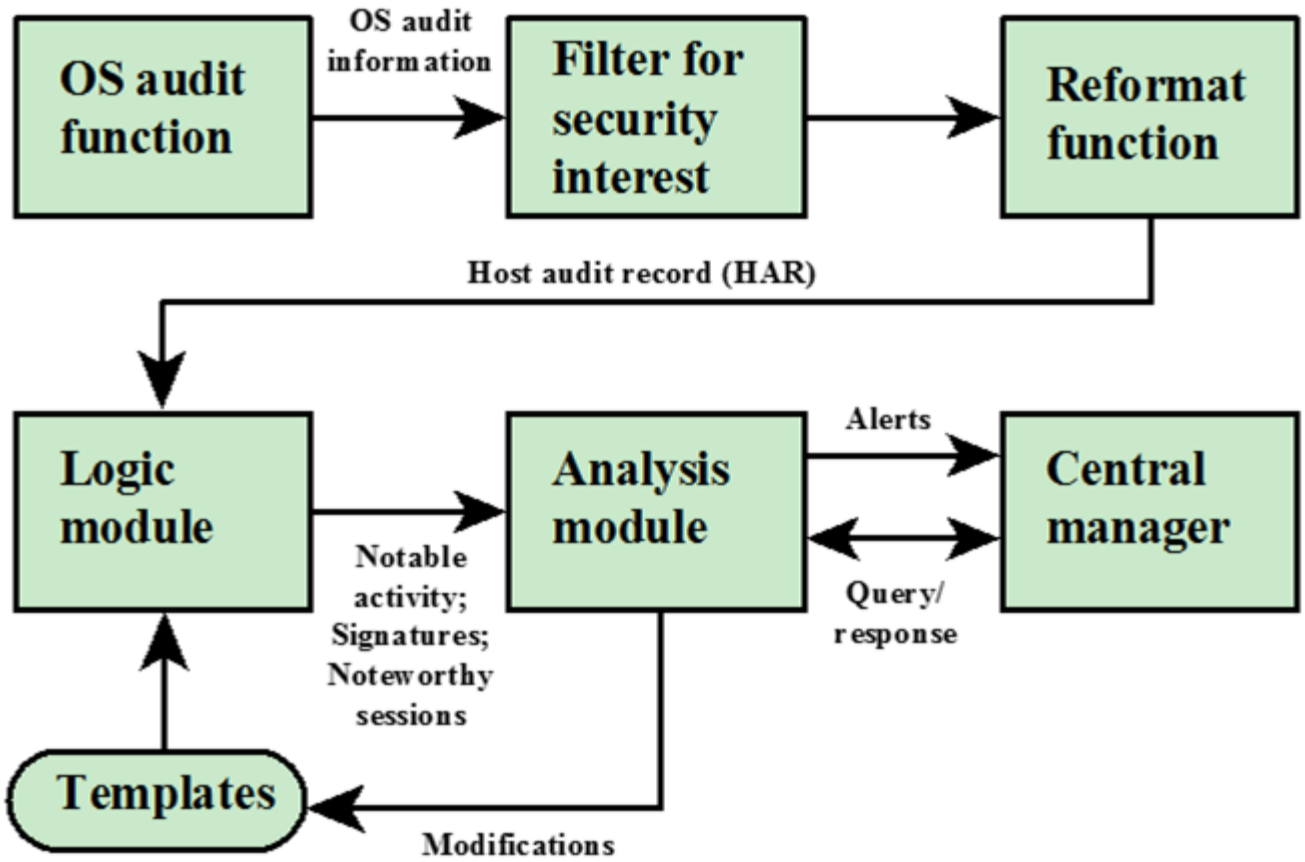


Figure 8.3 (Stallings et al., 2017)

Network-Based IDS (NIDS)

- Monitor traffic at selected points on a network
- Analyse packets in (near) real time
- May examine network, transport and/or application level protocol activity directed toward systems
- Different with HIDS
  - HIDS: user and software activity on a host
  - NIDS: network traffic towards systems
- Comprise a number of sensors
  - Inline
    - Traffic passes through the sensor
    - Standalone or combined with another network device (e.g. firewall)
    - Block the malicious traffic (acting as both intrusion detection and prevention)
  - Passive
    - Monitor a copy of traffic

Passive sensors

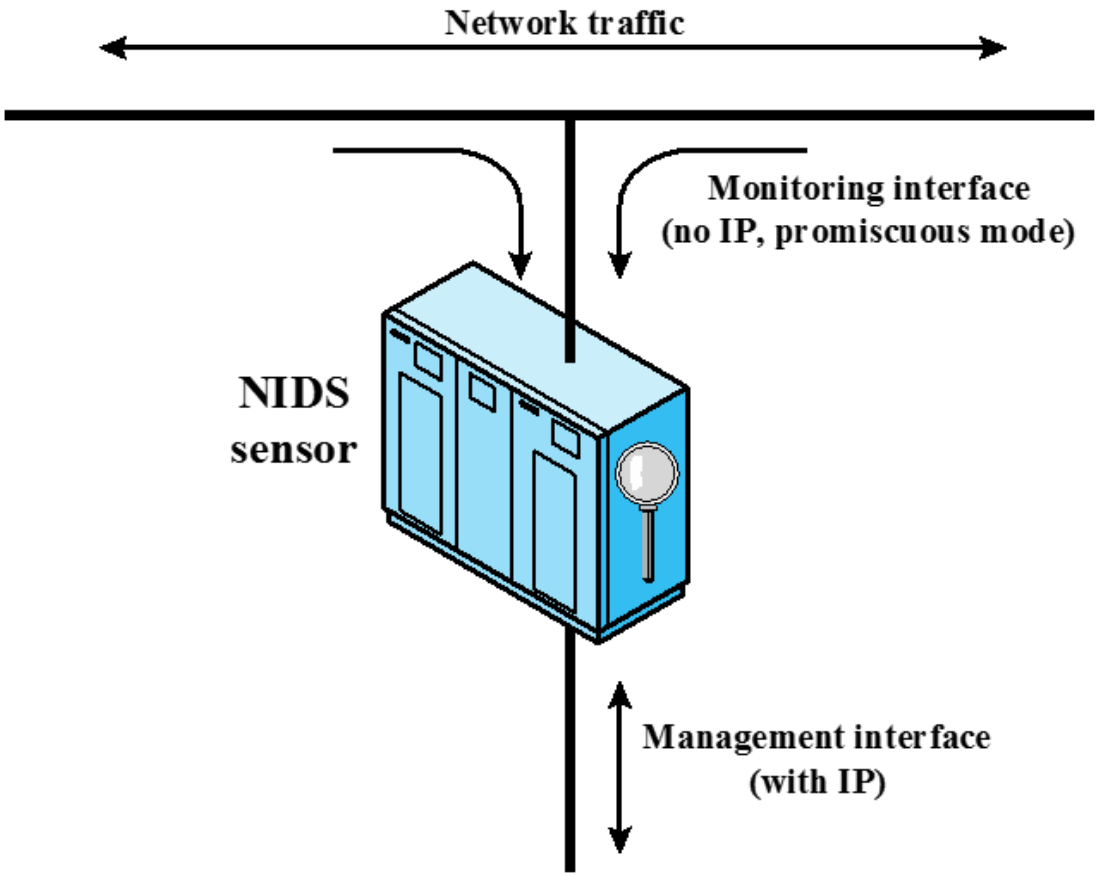


Figure 8.4 (Stallings et al., 2017)

NIDS sensor deployment

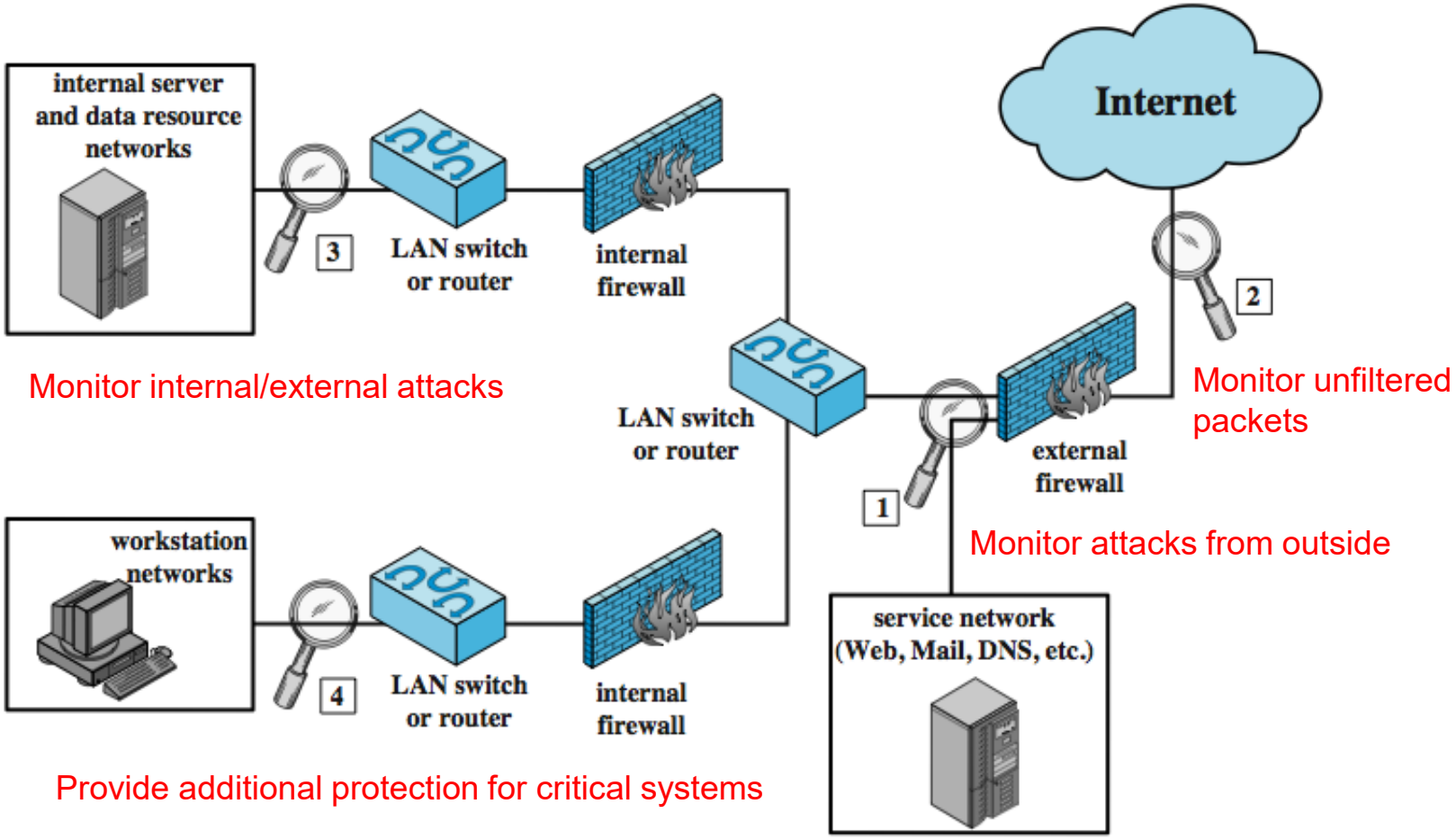


Figure 8.5 (Stallings et al., 2017)

NIDS intrusion detection techniques

- Signature detection (listed by NIST SP 800-94)
  - Application, transport, network layer reconnaissance and attacks
  - Unexpected application services
  - Policy violations
- Anomaly detection (listed by NIST SP 800-94)
  - Denial of service attacks, scanning, worms

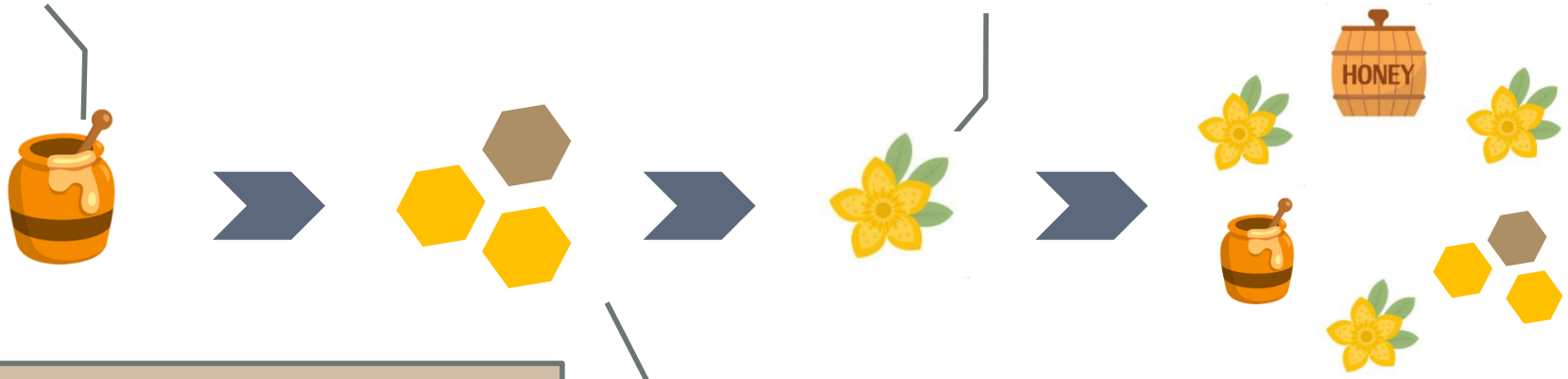
Logging of alerts

- Typical information logged by a NIDS sensor
  - Timestamp
  - Connection or session ID
  - Event or alert type
  - Rating (e.g., severity, impact)
  - 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 (e.g., authenticated username)

Cyberdeception

A honeypot is an “an information system resource whose value lies in unauthorised or illicit use of that resources” (Peter et al., n. d.)

A honeytoken is “some type of digital entity (e.g., fake email address, fake database record)” (Spitzner, 2003)



Evolution of deception technology (Perils, 2018)

A honeynet is “a network of high interaction honeypots that simulates a production network and configured such that all activity is monitored, recorded and in a degree, discreetly regulated” (Honeypot project, 2001)

A deception platform integrates honeypot technology, virtualisation, and automation technologies

Honeypot classification

- Low interaction honeypot
  - Consist of a software package that emulates particular IT services or systems to provide a realistic initial interaction, but does not execute a full version of those services or systems
  - Provide a less realistic target
- High interaction honeypot
  - A real system, with a full operating system, services and applications, which are instrumented and deployed where they can be accessed by attackers

Honeypot deployment

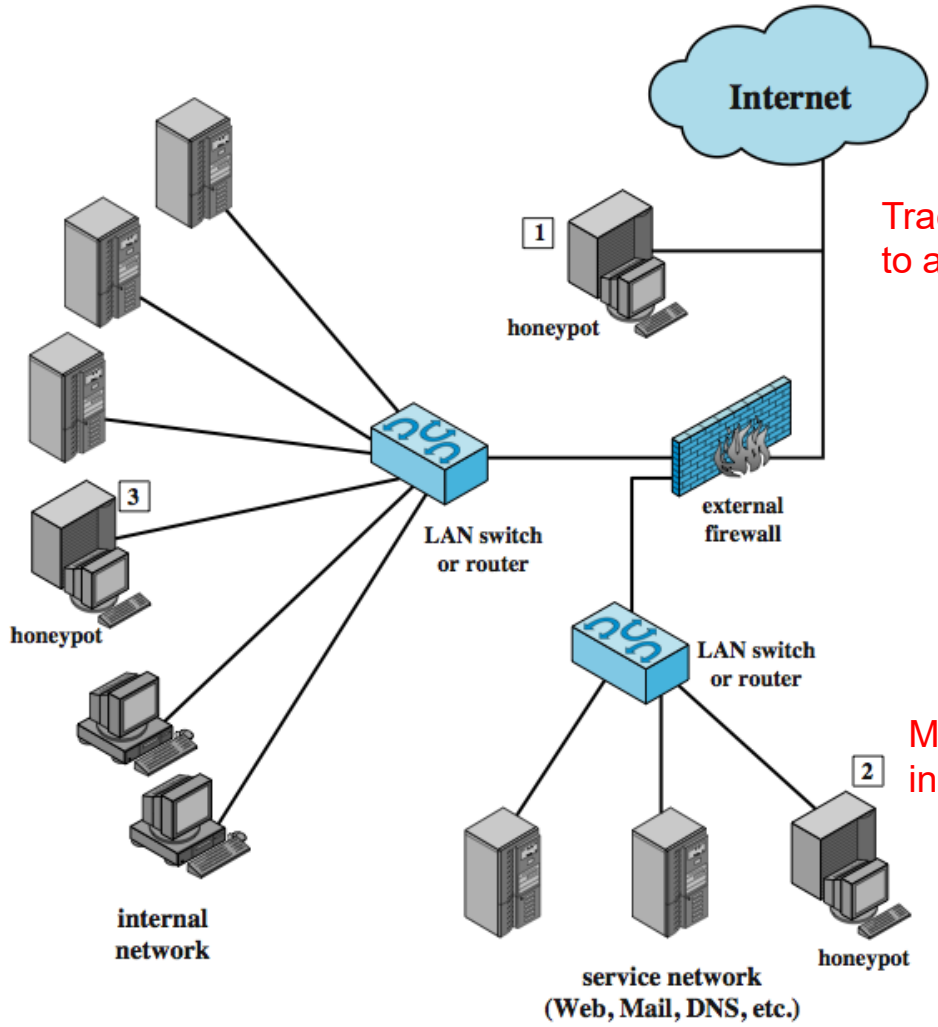


Figure 8.8 (Stallings et al., 2017)

Snort IDS

http://www.snort.org

- Lightweight IDS
  - Open source (rule-based)
  - Real-time packet capture and rule analysis
  - Components: decoder, detector, logger, alerter

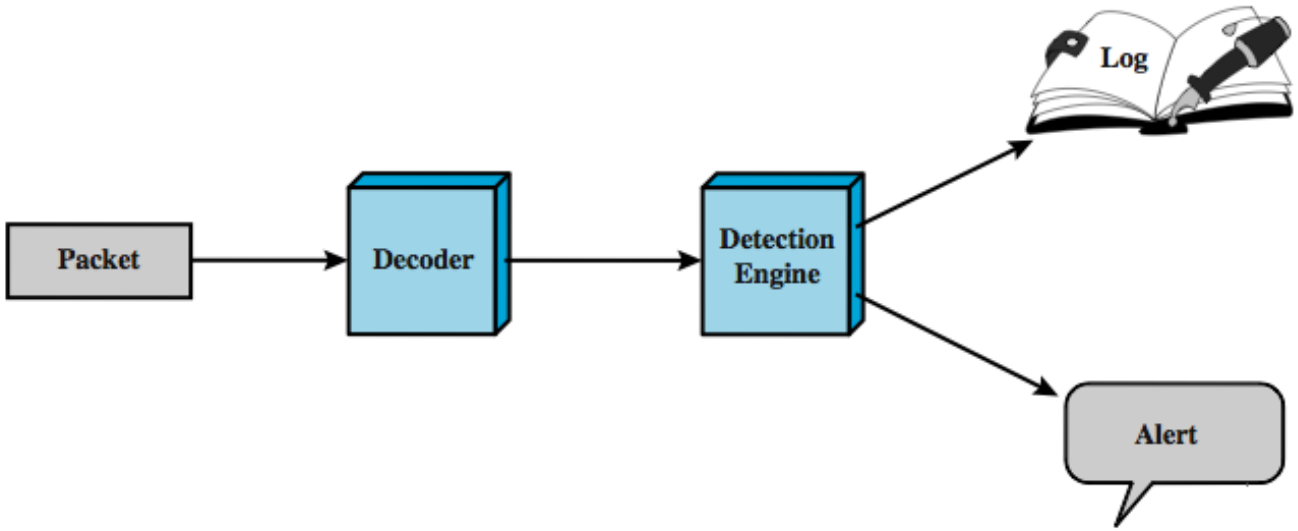


Figure 8.9 (Stallings et al., 2017)

Outline

- Intrusion detection
  - Basic principles
  - Analysis approaches
  - Host-based, network-based, distributed, hybrid intrusion detection
  - Cyberdeception
- Firewalls
- Intrusion prevention systems

Learning objectives

- Explain key features of firewalls
- Discuss basing options for firewalls
- Understand choices of firewall location and configurations
- Distinguish between firewall and intrusion prevention system

Definition of firewall

- “A device or system that controls the flow of traffic between networks using differing security postures” (“RFC 4949”, 2007)
- Effective means of protecting a local system or network of systems from network-based security threats

Firewall characteristics

- Characteristics used by firewall access policy (NIST SP 800-41)
  - IP address and protocol values
    - Network and transport layer characteristics
    - Used by packet filtering and stateful inspection firewalls
    - Limit access to specific services
  - Application protocol
    - Used by application-level gateway
    - Monitor information exchange for application protocols (e.g. HTTP web requests to authorised sites only)
  - User identity
    - For insider users who identify themselves using some form of authentication
  - Network activity
    - Activity patterns (e.g. only in business hours)

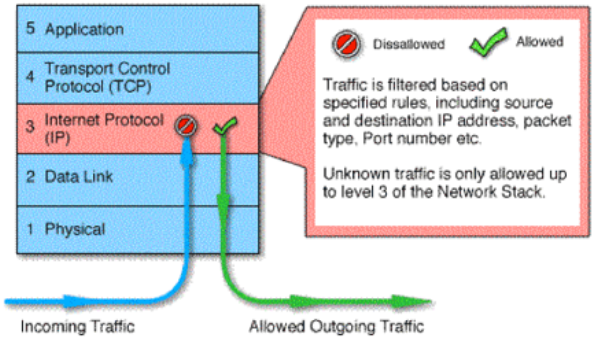
Firewall capabilities and limits

- Capabilities
  - Define a single choke point
  - Provide a location for monitoring security events
  - Convenient platform for some Internet functions (e.g. NAT, IPSec)
- Limitations
  - Cannot protect against attacks bypassing firewall
  - May not protect fully against internal threats
  - Improperly secured wireless LAN
  - Laptop or portable storage device infected outside then used inside

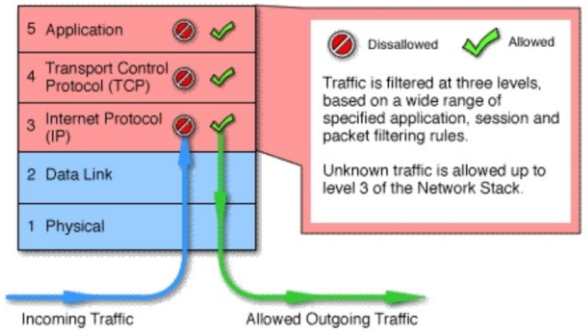
Types of firewalls

- Monitor network traffic at a number of levels from TCP/IP model
- Positive filter vs negative filter

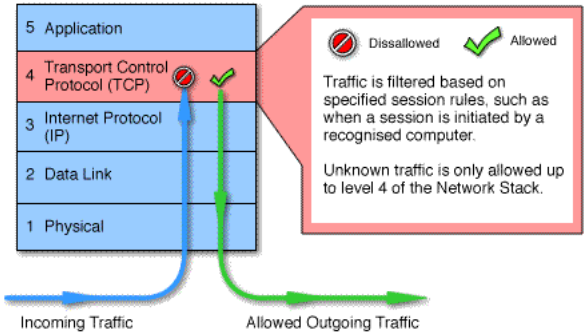
Packet filtering firewall
("Network protocols and firewalls", 2019)



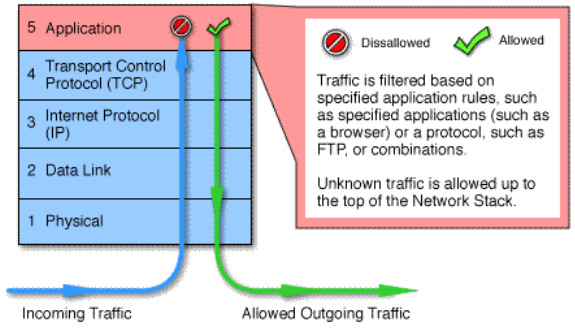
Stateful inspection firewalls
("Network protocols and firewalls", 2019)



Circuit-level gateway/proxy
("Network protocols and firewalls", 2019)



Application-level gateway
("Network protocols and firewalls", 2019)



Packet filtering firewall

- Apply a set of rules to each incoming and outgoing IP packet based on information in packet header
  - Source IP address, destination IP address, port number, IP protocol field
- Two default policies if no match to any rule
  - Discard
    - Prohibit unless expressly permitted
    - Conservative
  - Forward
    - Permit unless expressly prohibited
    - Ease of use
    - Reduced security

Packet filter examples

- Simple example rule set for SMTP
  - Allow inbound mail (port 25 is for SMTP incoming)
  - Allow response to an inbound SMTP connection
  - Allow outbound mail
  - Allow response to an outbound SMTP connection
  - Default policy

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

Table 9.1 (Stallings et al., 2017)

Packet filter examples (cont.)

- Problems and countermeasures
  - Rule 4
    - Attacker opens connection to internal web proxy server over 8080 -> specify source port to 25
  - Rule 3 & 4
    - Attacker sends packets with TCP source port at 25 -> add ACK flag

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

Table 9.1 (Stallings et al., 2017)

Packet filter examples (cont.)

- Problems and countermeasures
  - Rule 4
    - Attacker opens connection to internal web proxy server over 8080 -> specify source port to 25
  - Rule 3 & 4
    - Attacker sends packets with TCP source port at 25 -> add ACK flag

Rule	Direction	Src address	Src port	Dest address	Protocol	Dest port	Flag	Action
4	In	External	25	Internal	TCP	>1023	ACK	Permit

Rule 4 (Stallings et al., 2017)

Weaknesses of packet filter firewalls

- Weaknesses by NIST SP 800-41
  - Cannot prevent attacks on application bugs
  - Limited logging functionality
  - No support of advanced user authentication
  - Vulnerable to attacks on vulnerabilities of TCP/IP specification and protocol stack (e.g. IP address spoofing)
  - Improper configuration
- Attacks and countermeasures
  - IP address spoofing
    - Discard external packets with an inside source address
  - Source routing attacks (attackers specify routes)
    - Discard packets with this option
  - Tiny fragment attacks (to circumvent filtering rules that depend on TCP header)
    - Require first fragment contain predefined minimum of the TCP header

Stateful inspection firewall

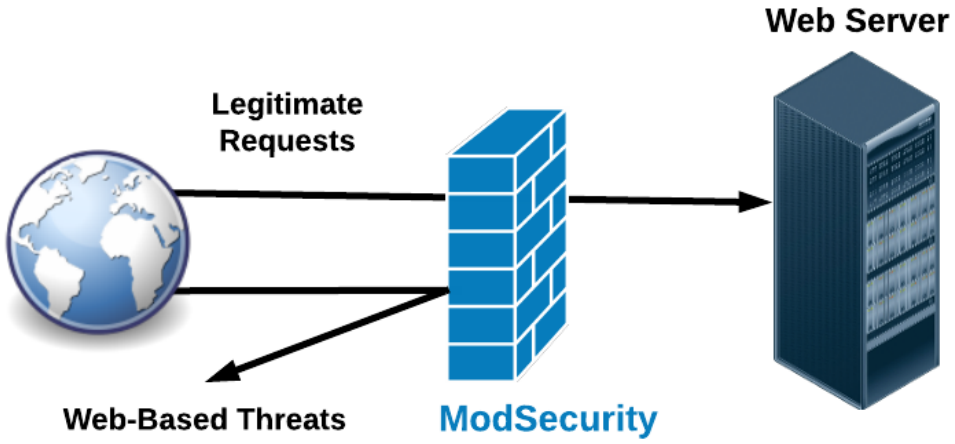
- Review packet header information and TCP connections
  - An application creates a TCP connection with a remote host
  - TCP port number for the remote (server) application is less than 1024 (well-known ports to particular applications)
  - TCP port number for the local (client) application is between 1024 and 65535 (registered and private ports).

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
223.21.22.12	1046	192.168.1.6	80	Established

Table 9.2 (Stallings et al., 2017)

Application-level gateway

- Act as a relay of application-level traffic
  - User contacts gateway using an application
  - User responds with a valid user ID and authentication information
  - Gateway contacts the application on remote host and relays TCP segments between application server and user
- Have proxy code for each application
- More secure but higher processing overhead
- Implementation example: ModSecurity



Open source web application firewall (Belisle, 2018)

Circuit-level gateway

- How it works
  - Set up two TCP connections, to an inside user and to an outside host
  - Once connection is established, relays TCP segments from one connection to the other without examining contents
  - Security function is to determine which connections are allowed
- Standalone or integrated with application-level gateway
- Typical usage
  - Inside users are trusted
  - Configure application-level gateway on inbound connections and circuit-level gateway on outbound connections
  - Hence lower overheads

Firewall basing - bastion host

- Critical strongpoint in network
- Host application/circuit-level gateways
- Common characteristics
  - Run secure OS with only essential services
  - May require user authentication for access
  - For each proxy
    - Limited command set and access to hosts on the protected network
    - Detailed audit information
    - Small software packages designed for security
    - Independent of other proxies
    - No disk access
    - Run as non-privileged user in a private and secured directory

Firewall basing - host-based firewall

- Software module to secure individual host
  - Filter and restrict packet flows
  - Often used on servers
- Advantages
  - Tailored filtering rules
  - Protection from both internal/external attacks
  - Used in conjunction with standalone firewalls to provide additional layer of protection

Firewall basing - personal firewall

- Control traffic flow between PC/workstation and Internet/corporate network
  - Used in home or corporate intranets
  - Software module on PC or firewall functionality integrated with router
  - Less complex than either server-based firewalls or standalone firewalls
- Roles
  - Deny unauthorised access
  - Monitor outgoing traffic to detect/block worm/malware activity

Firewall locations

- External firewall
  - Protection for the DMZ and internal network
- Internal firewall
  - More stringent filtering capability
  - Provide two way protection
  - Protect portions of internal network from each other

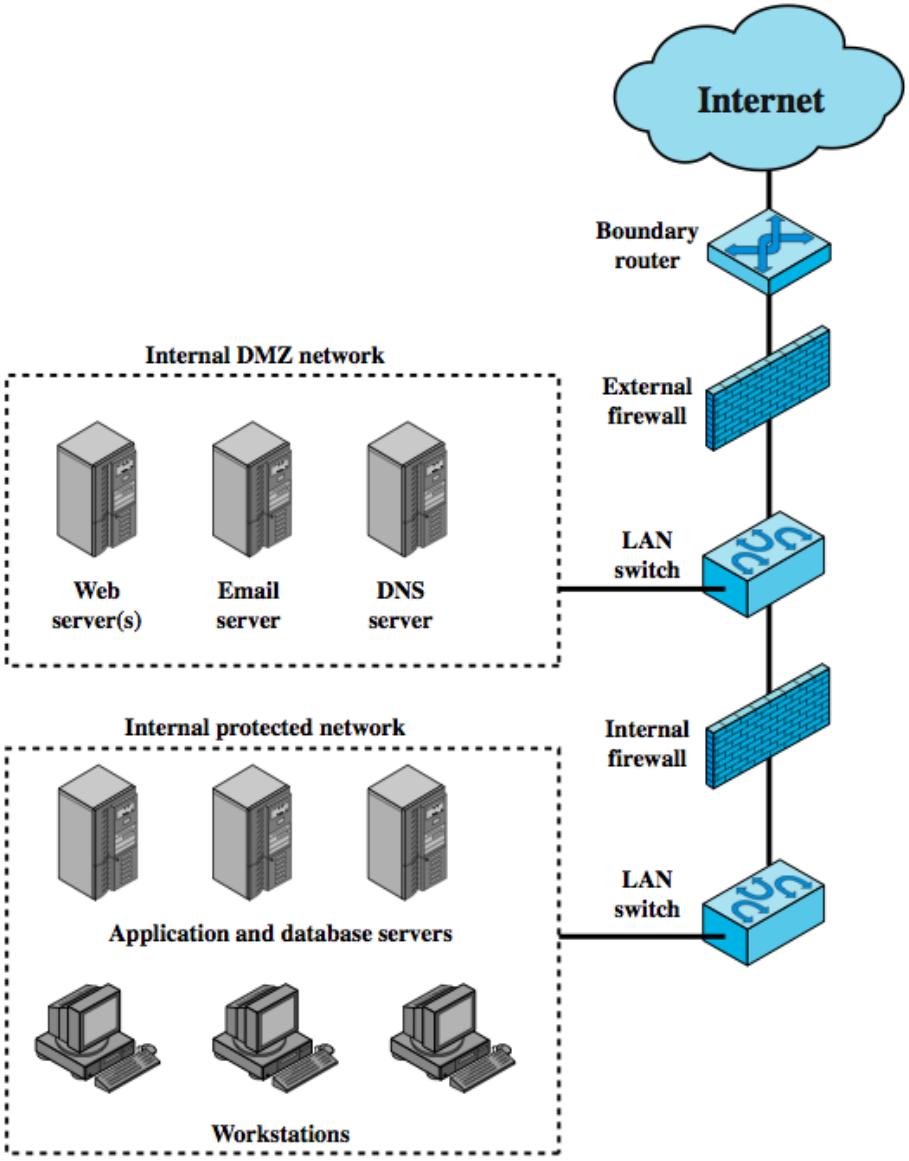


Figure 9.2 (Stallings et al., 2017)

Virtual private networks

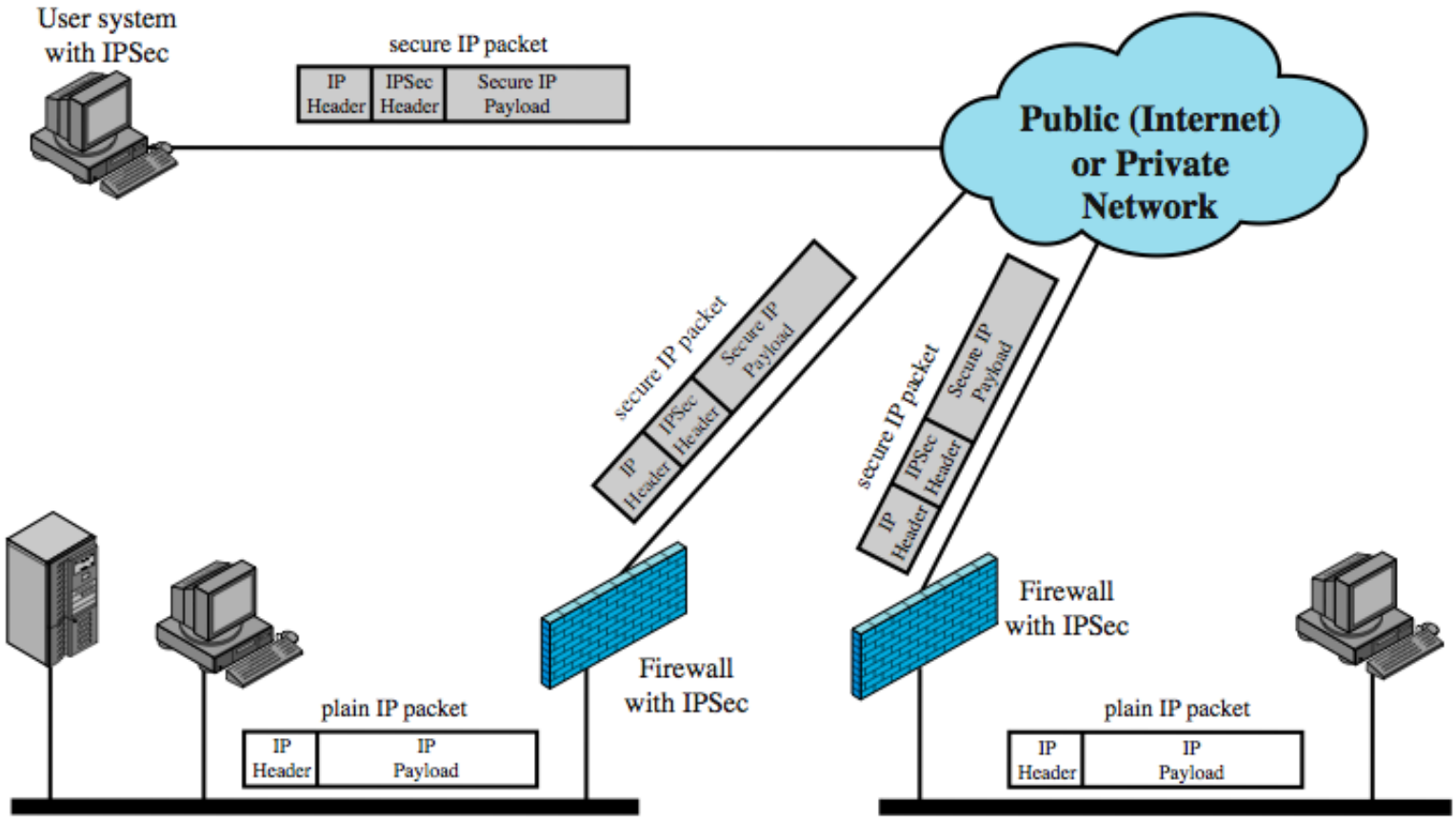


Figure 9.3 (Stallings et al., 2017)

Distributed firewalls

- A combination of standalone firewalls and host-based firewalls
- Establish both an internal and an external DMZ
  - Web servers that need less protection could be placed in an external DMZ

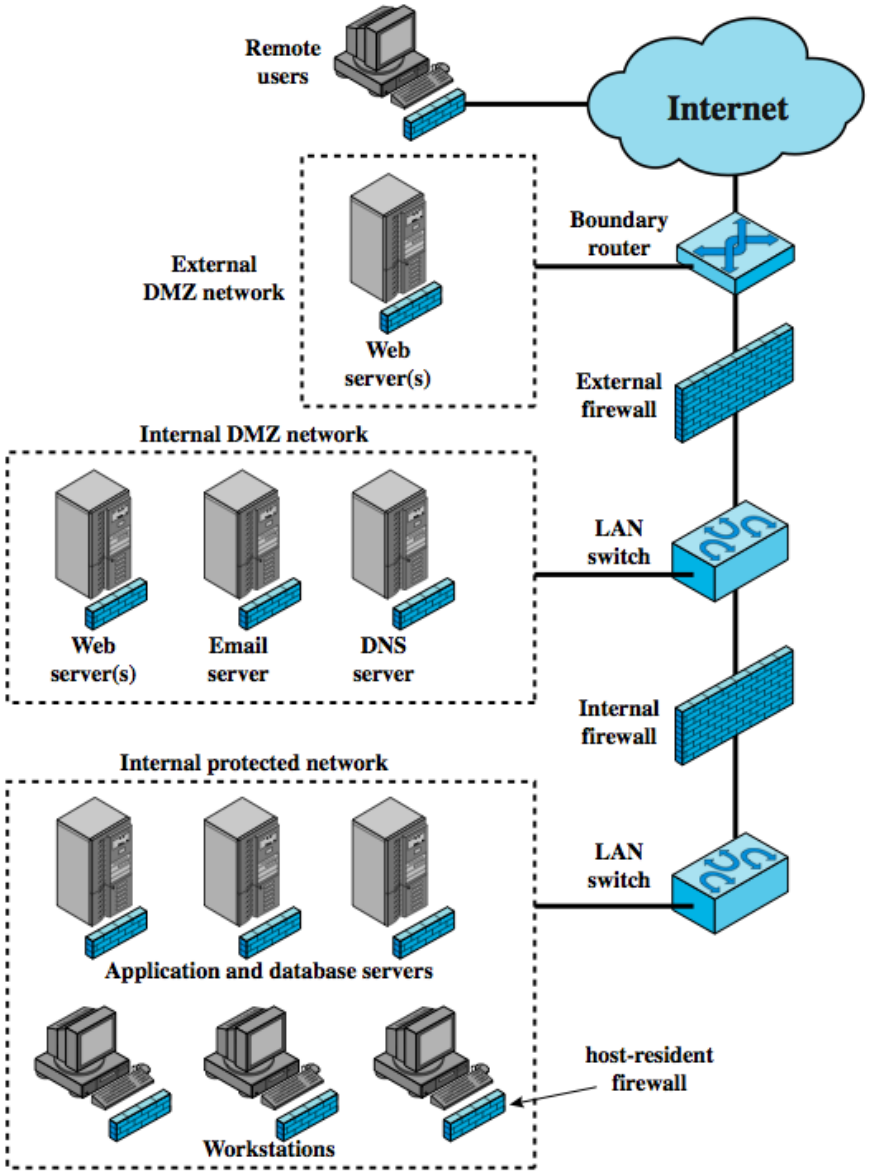


Figure 9.4 (Stallings et al., 2017)

Intrusion prevention system

- IPS
  - Extension of IDS to detect attacks using IDS algorithms and block packets as a firewall does
- Host-based IPS
  - Signature or anomaly detection
    - Example of malicious behavior: modification of system resources, buffer overflow, privilege escalation, directory traversal
  - Sandbox approach
    - Especially suited to mobile code
      - Quarantine code in an isolated system area; run the code and monitor behaviours; prevent the code from executing
  - Tailored to a specific platform

Intrusion prevention system (cont.)

- Network-based IPS
  - Signature and anomaly detection
  - May provide flow data protection
  - General methods to identify attacks
    - Pattern matching
    - Stateful matching
    - Protocol anomaly
    - Traffic anomaly
    - Statistical anomaly

Summary

- Intrusion detection approaches
  - Host-based (single and distributed)
  - Network-based
  - Signature and anomaly detection
- Cyberdeception
- Types of firewalls
  - Packet filter, stateful inspection, application and circuit gateways
- Firewall basing and locations
- Intrusion prevention systems

Next lecture – Buffer overflow

- Definitions
- Buffer Overflow Basics
- Buffer Overflow Defenses
- Other Overflow Attacks
- Software Security
- Defensive programming
- Injection attacks
- Writing safe code
- Interacting with OS

References

- Stallings, W. and Brown, L. (2017). Computer Security: Principles and Practice. Pearson Education.
- RFC 4949 (2007). Retrieved from https://tools.ietf.org/html/rfc4949
- Perils, W. (2018) Deception Technology. Retrieved from https://www.wwt.com/article/deception-technology
- Peter, E. and Schiller, T. A Practical Guide to Honeypots. Retrieved from https://www.cse.wustl.edu/~jain/cse571-09/ftp/honey/#sec1.2
- Honeynet Project (2001) Know Your Enemy: Honeynets. Retrieved from https://www.symantec.com/connect/articles/know-your-enemy-honeynets
- Spitzner, L. (2003) Honeytokens: The Other Honeypot. Retrieved from https://www.symantec.com/connect/articles/honeytokens-other-honeypot
- Network protocols and firewalls (2019). Retrieved from http://gauss.ececs.uc.edu/Courses/c653/lectures/Week1/firewall.1.html
- Belisle, R. (2018) Why ModSecurity Should Be Your Web Application Firewall. Retrieved from https://blog.nexcess.net/why-modsecurity-should-be-your-web-application-firewall/
- MITRE ATT&CK. Retrieved from https://attack.mitre.org/