CS 493
Secure Software Systems
Intrusion Detection – part 2 revised

Dr. Williams Central Connecticut State University

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

© Chad Williams Nov-17 Adapted slides from Pearson Higher Ed, copyright 2014

Honeypot Classifications

- 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
  - Provides a less realistic target
  - Often sufficient for use as a component of a distributed IDS to warn of imminent attack
- 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
  - Is a more realistic target that may occupy an attacker for an extended period
  - However, it requires significantly more resources
  - If compromised could be used to initiate attacks on other systems

© Chad Williams Nov-17 Adapted slides from Pearson Higher Ed, copyright 2014

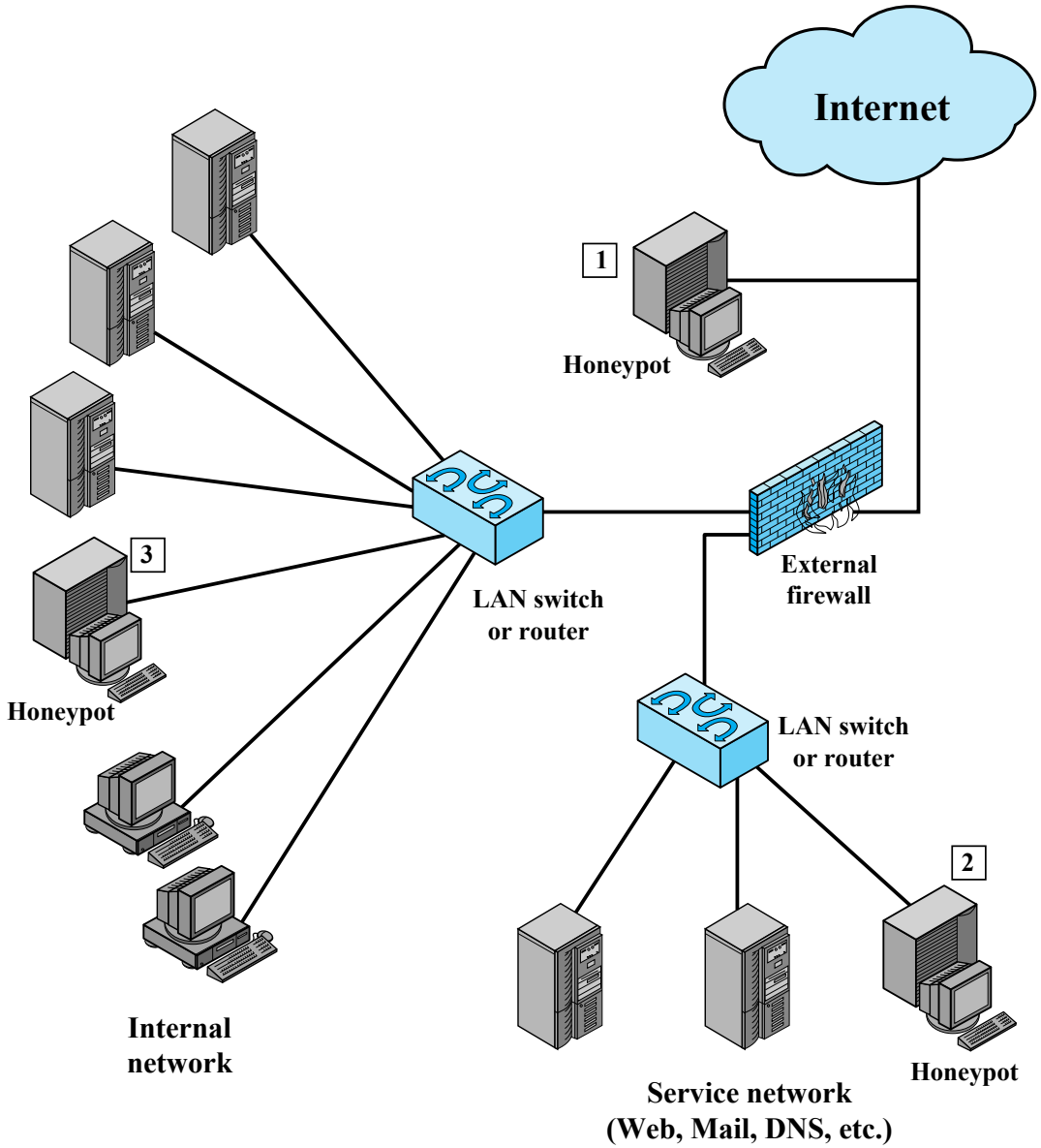


Figure 8.8 Example of Honeypot Deployment

© Chad Williams Nov-17 Adapted slides from Pearson Higher Ed, copyright 2014

FIREWALLS AND INTRUSION PREVENTION SYSTEMS

The Need For Firewalls

- Internet connectivity is essential
  - However it creates a threat
- Effective means of protecting LANs
- Inserted between the premises network and the Internet to establish a controlled link
  - Can be a single computer system or a set of two or more systems working together
- Used as a perimeter defense
  - Single choke point to impose security and auditing
  - Insulates the internal systems from external networks

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

Firewall Access Policy

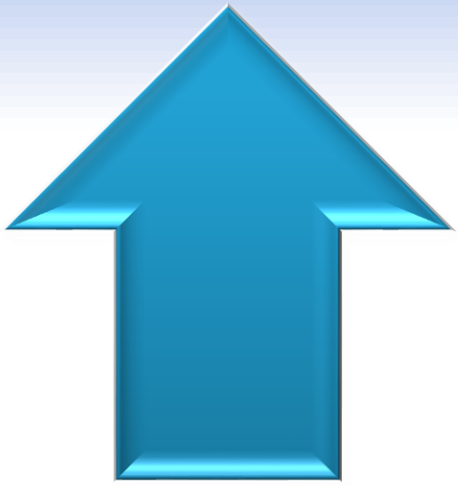
- A critical component in the planning and implementation of a firewall is specifying a suitable access policy
  - This lists the types of traffic authorized to pass through the firewall
  - Includes address ranges, protocols, applications and content types
- This policy should be developed from the organization's information security risk assessment and policy
- Should be developed from a broad specification of which traffic types the organization needs to support
  - Then refined to detail the filter elements which can then be implemented within an appropriate firewall topology

Firewall Filter Characteristics

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

IP address and protocol values	Application protocol	User identity	Network activity
This type of filtering is used by packet filter and stateful inspection firewalls	This type of filtering is used by an application-level gateway that relays and monitors the exchange of information for specific application protocols	Typically for inside users who identify themselves using some form of secure authentication technology	Controls access based on considerations such as the time or request, rate of requests, or other activity patterns
Typically used to limit access to specific services			

Firewall Capabilities And Limits



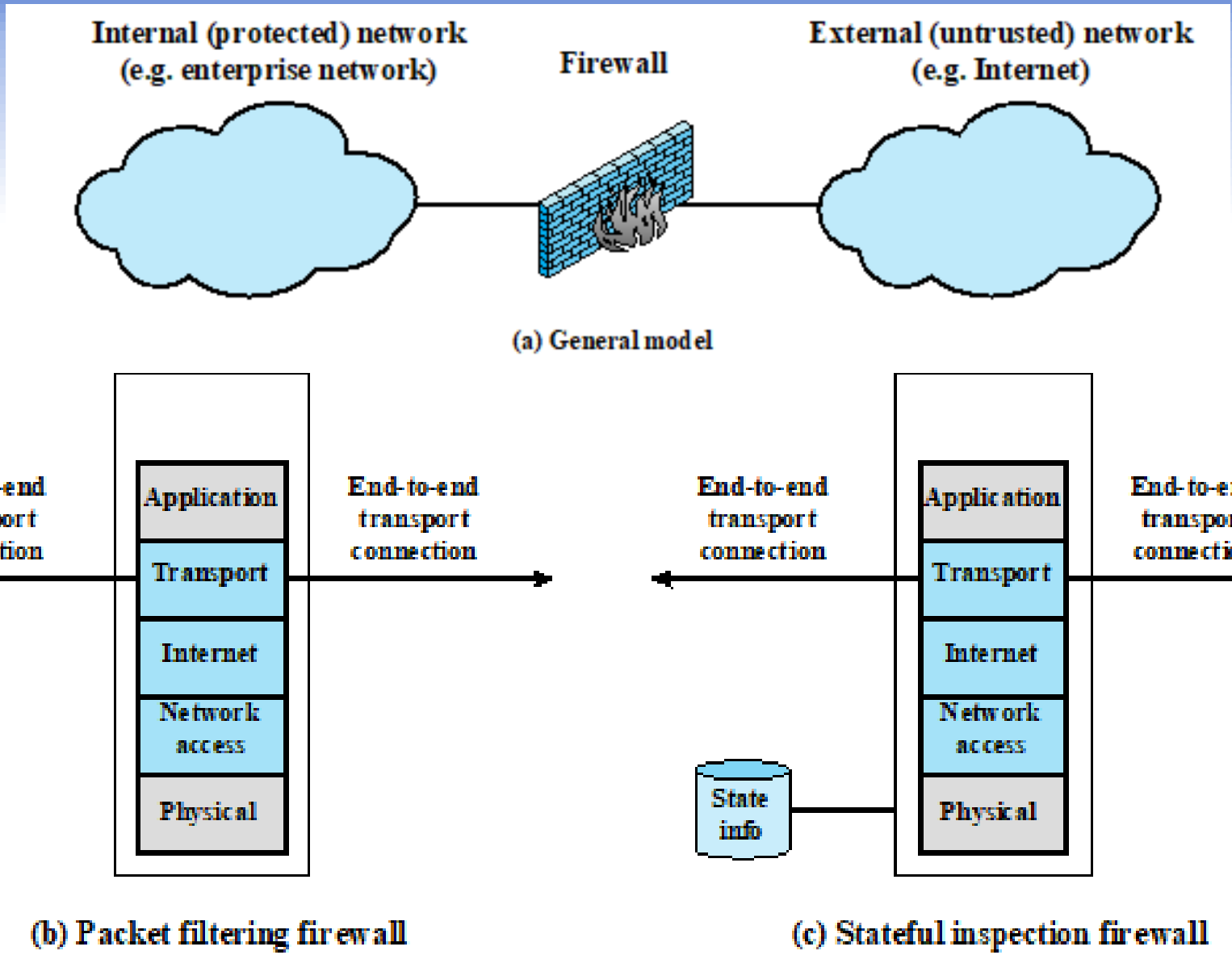
Capabilities:

- **Defines a single choke point**
- **Provides a location for monitoring security events**
- **Convenient platform for several Internet functions that are not security related**
- **Can serve as the platform for IPSec**

Limitations:

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

Firewall types



© Chad Williams Nov-17 Adapted slides from Pearson Higher Ed, copyright 2014

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
  - Forwards or discards the packet based on rules match

Filtering rules are based on information contained in a network packet
• Source IP address • Destination IP address • Source and destination transport-level address • IP protocol field • Interface

- Two 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

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

Packet Filter Advantages And Weaknesses

- Advantages
  - Simplicity
  - Typically transparent to users and are very fast
- Weaknesses
  - Cannot prevent attacks that employ application specific vulnerabilities or functions
  - Limited logging functionality
  - Do not support advanced user authentication
  - Vulnerable to attacks on TCP/IP protocol bugs
  - Improper configuration can lead to breaches

Stateful Inspection Firewall

Tightens rules for TCP traffic by creating a directory of outbound TCP connections

- There is an entry for each currently established connection
- Packet filter allows incoming traffic to high numbered ports only for those packets that fit the profile of one of the entries in this directory

Reviews packet information but also records information about TCP connections

- Keeps track of TCP sequence numbers to prevent attacks that depend on the sequence number
- Inspects data for protocols like FTP, IM and SIPS commands

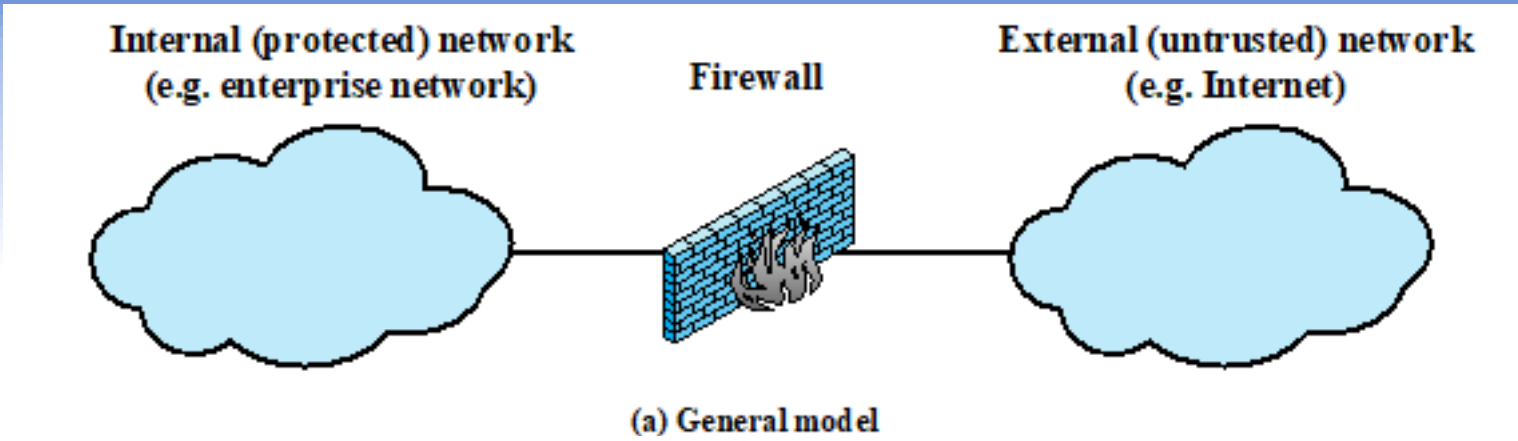
Table 9.2

Example Stateful Firewall

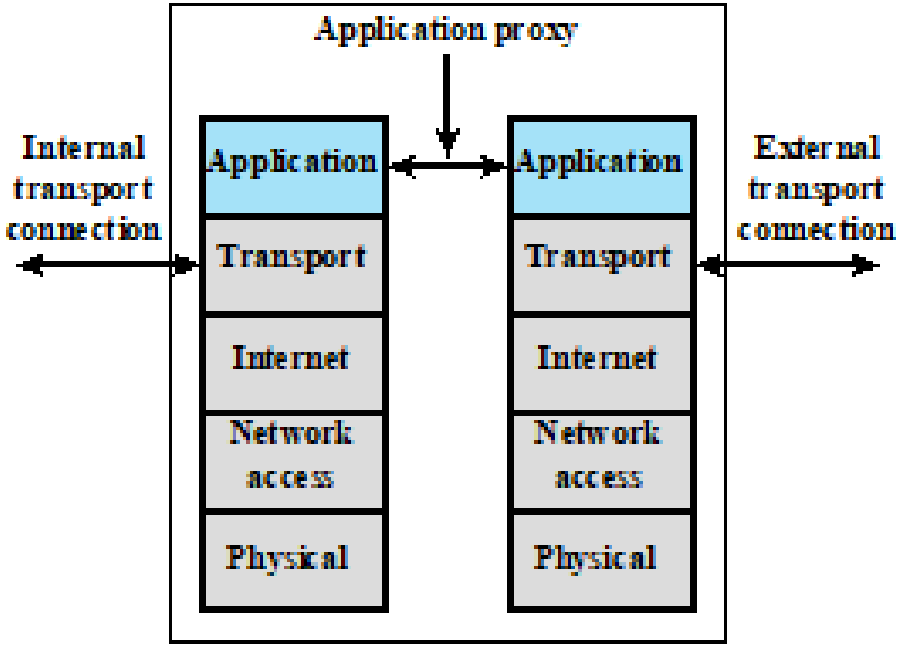
Connection State Table

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

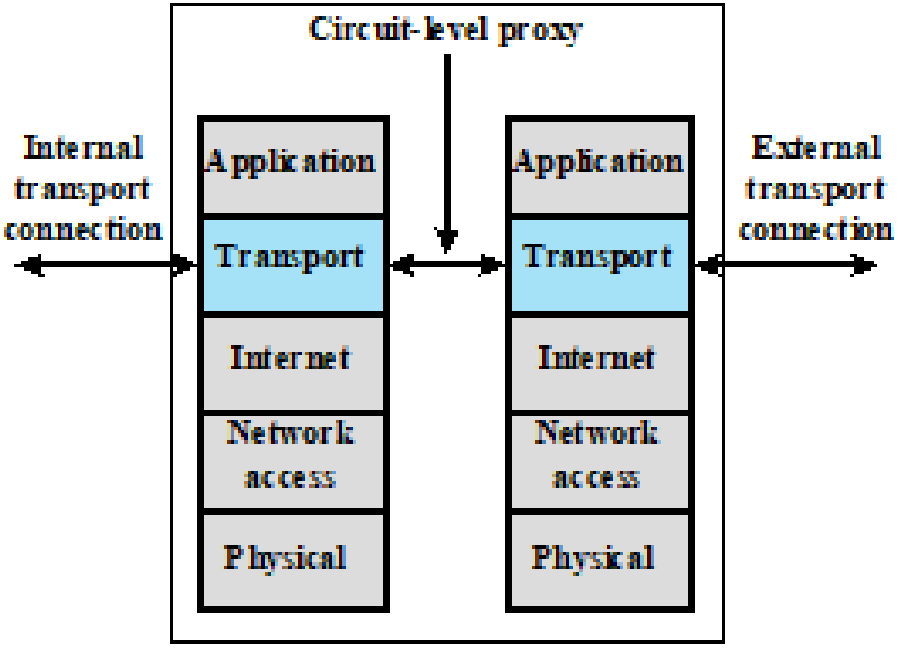
Firewall types



(a) General model



(d) Application proxy firewall



(e) Circuit-level proxy firewall

Application-Level Gateway

- Also called an application proxy
- Acts as a relay of application-level traffic
  - User contacts gateway using a TCP/IP application
  - User is authenticated
  - Gateway contacts application on remote host and relays TCP segments between server and user
- Must have proxy code for each application
  - May restrict application features supported
- Tend to be more secure than packet filters
- Disadvantage is the additional processing overhead on each connection

Circuit-Level Gateway

Circuit level proxy

- Sets up two TCP connections, one between itself and a TCP user on an inner host and one on an outside host
- Relays TCP segments from one connection to the other without examining contents
- Security function consists of determining which connections will be allowed

Typically used when inside users are trusted

- May use application-level gateway inbound and circuit-level gateway outbound
- Lower overheads

Host-Based Firewalls

- Used to secure an individual host
- Available in operating systems or can be provided as an add-on package
- Filter and restrict packet flows
- Common location is a server

Advantages:

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

Personal Firewall

- Controls traffic between a personal computer or workstation and the Internet or enterprise network
- For both home or corporate use
- Typically is a software module on a personal computer
- Can be housed in a router that connects all of the home computers to a DSL, cable modem, or other Internet interface
- Typically much less complex than server-based or stand-alone firewalls
- Primary role is to deny unauthorized remote access
- May also monitor outgoing traffic to detect and block worms and malware activity

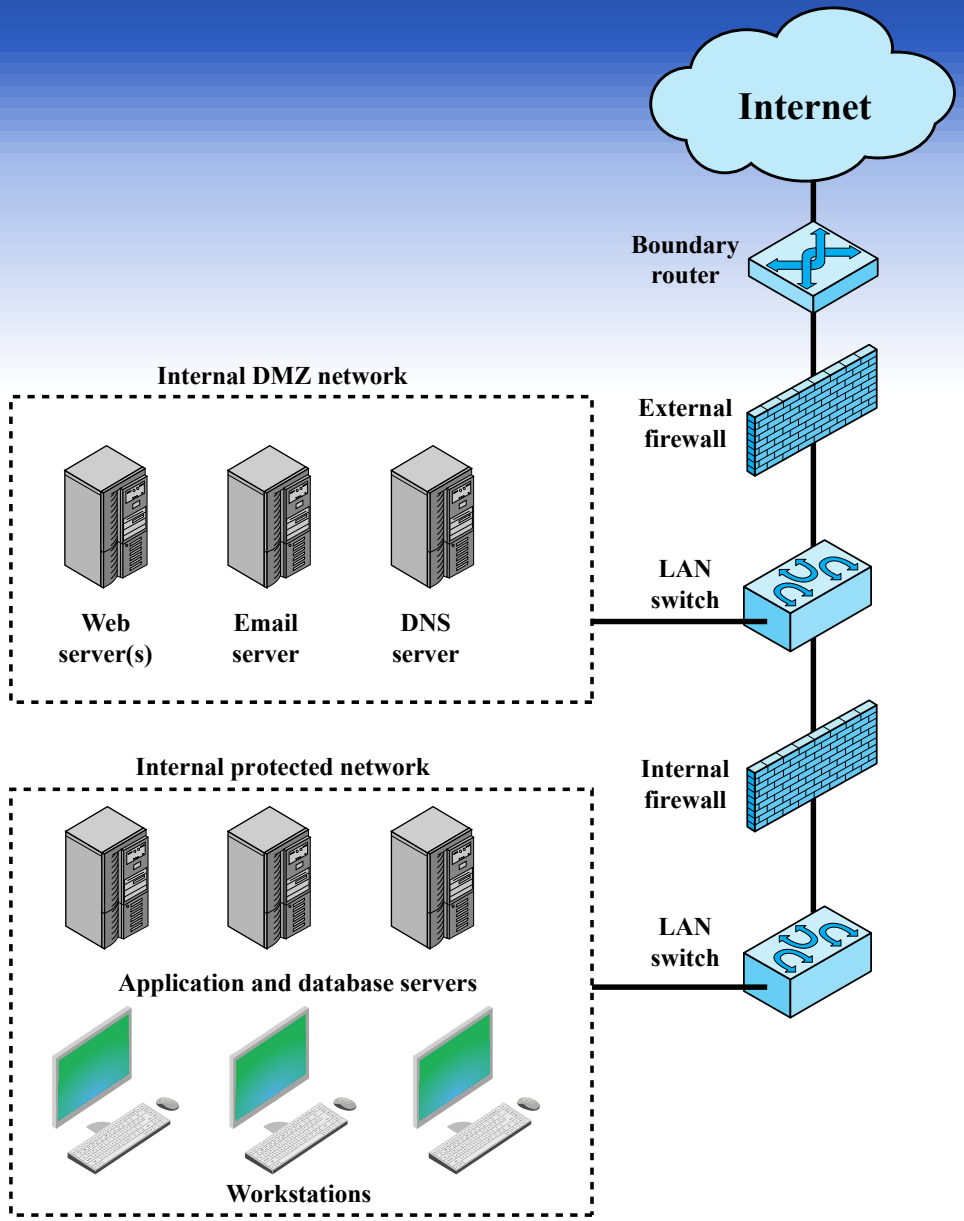


Figure 9.2 Example Firewall Configuration

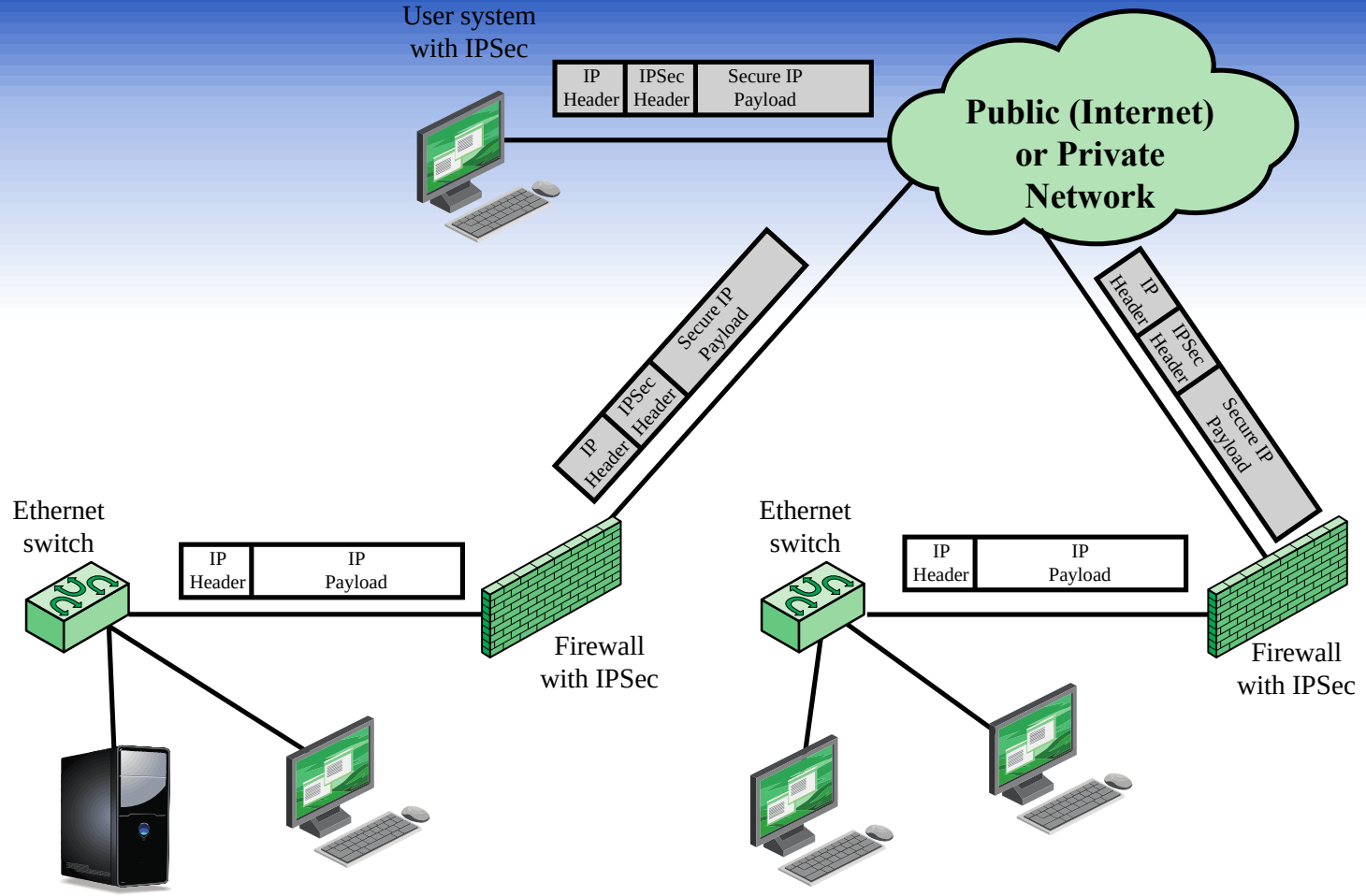


Figure 9.3 A VPN Security Scenario

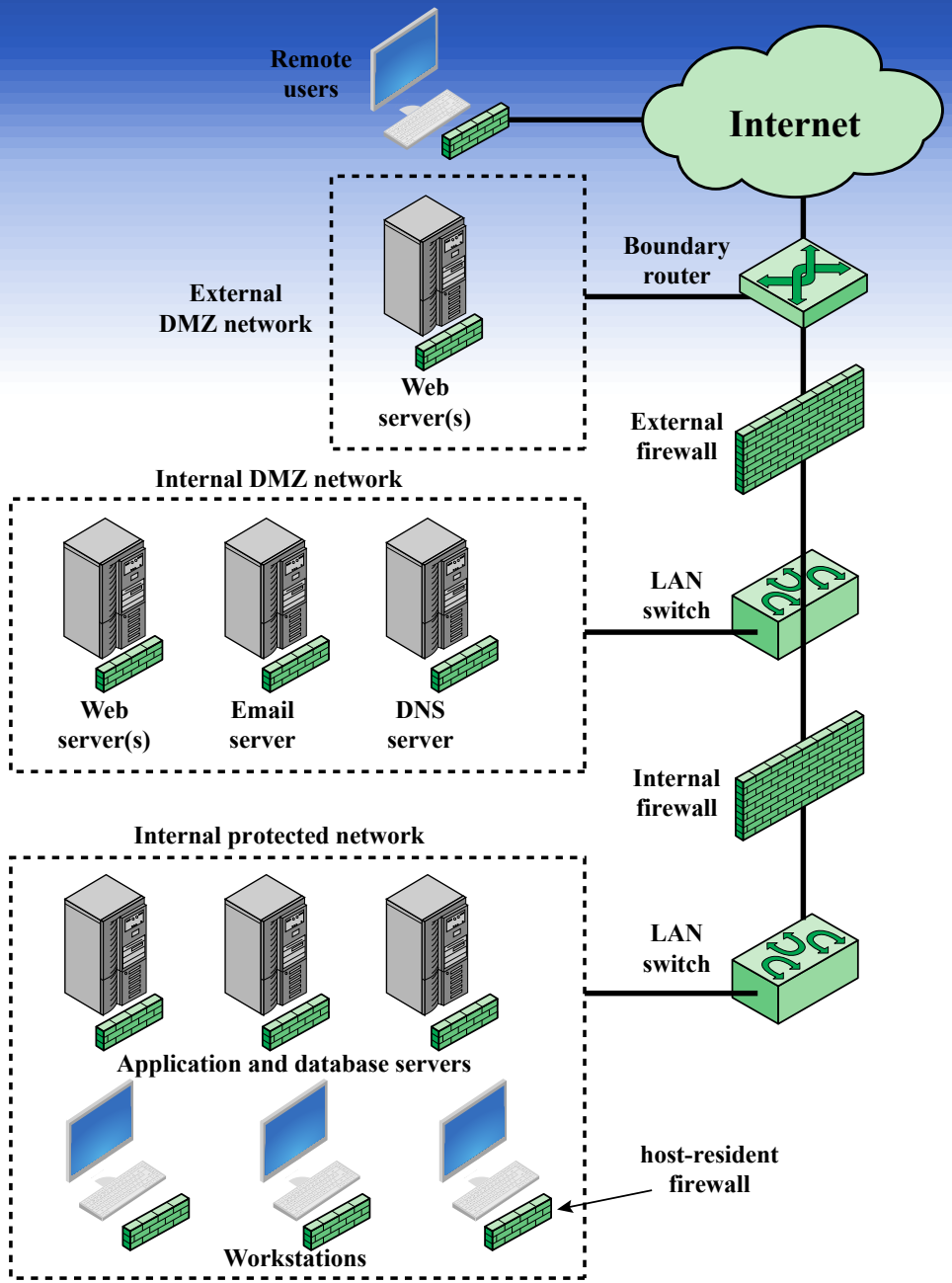


Figure 9.4 Example Distributed Firewall Configuration

Intrusion Prevention Systems (IPS)

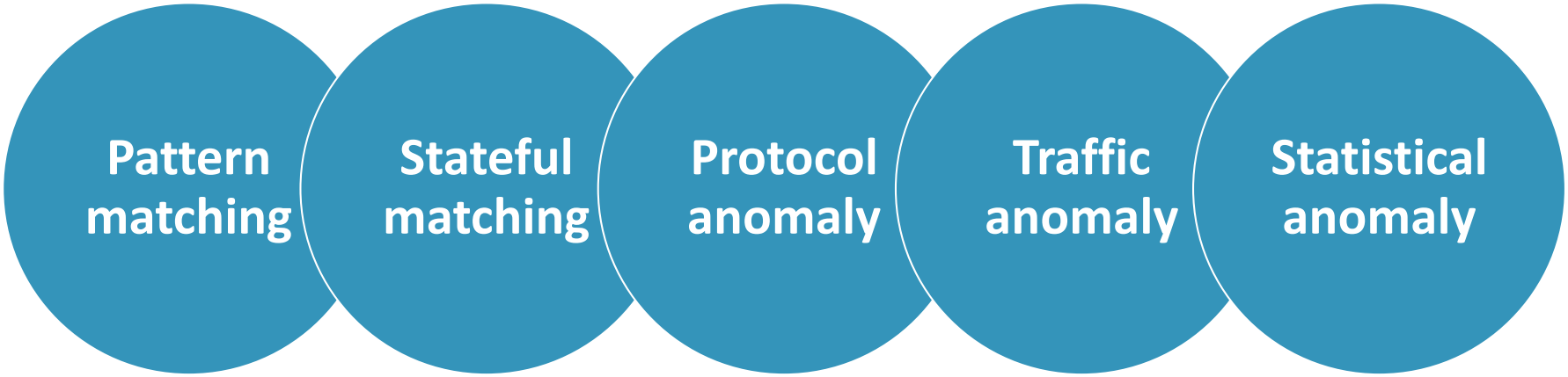
- Also known as Intrusion Detection and Prevention System (IDPS)
- Is an extension of an IDS that includes the capability to attempt to block or prevent detected malicious activity
- Can be host-based, network-based, or distributed/hybrid
- Can use anomaly detection to identify behavior that is not that of legitimate users, or signature/heuristic detection to identify known malicious behavior can block traffic as a firewall does, but makes use of the types of algorithms developed for IDSs to determine when to do so

Host-Based IPS (HIPS)

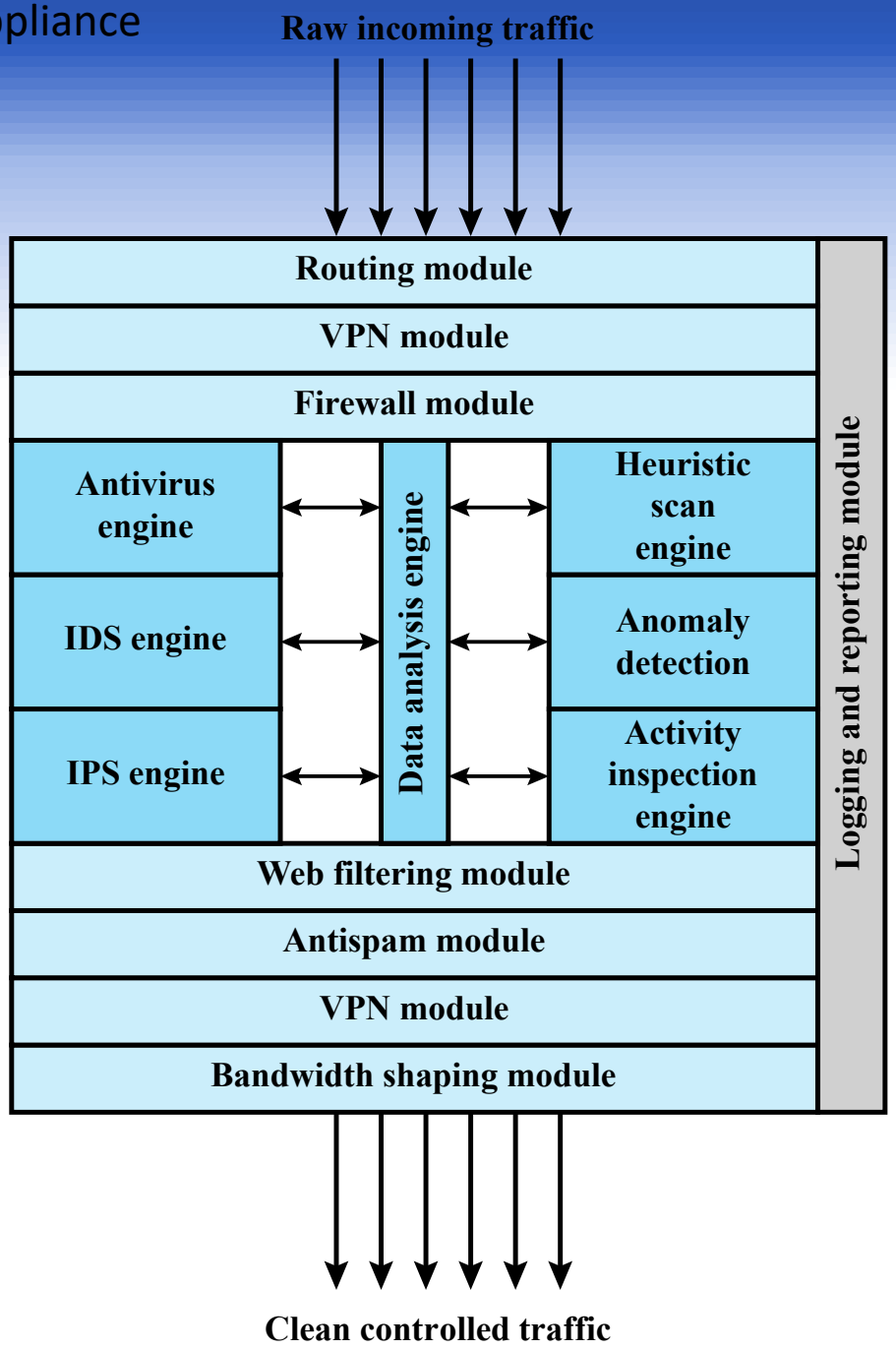
- Can make use of either signature/heuristic or anomaly detection techniques to identify attacks
  - Signature: focus is on the specific content of application network traffic, or of sequences of system calls, looking for patterns that have been identified as malicious
  - Anomaly: IPS is looking for behavior patterns that indicate malware
- Examples of the types of malicious behavior addressed by a HIPS include:
  - Modification of system resources
  - Privilege-escalation exploits
  - Buffer-overflow exploits
  - Access to e-mail contact list
  - Directory traversal

Network-Based IPS (NIPS)

- Inline NIDS with the authority to modify or discard packets and tear down TCP connections
- Makes use of signature/heuristic detection and anomaly detection
- May provide flow data protection
  - Requires that the application payload in a sequence of packets be reassembled
- Methods used to identify malicious packets:



Unified Threat Management Appliance



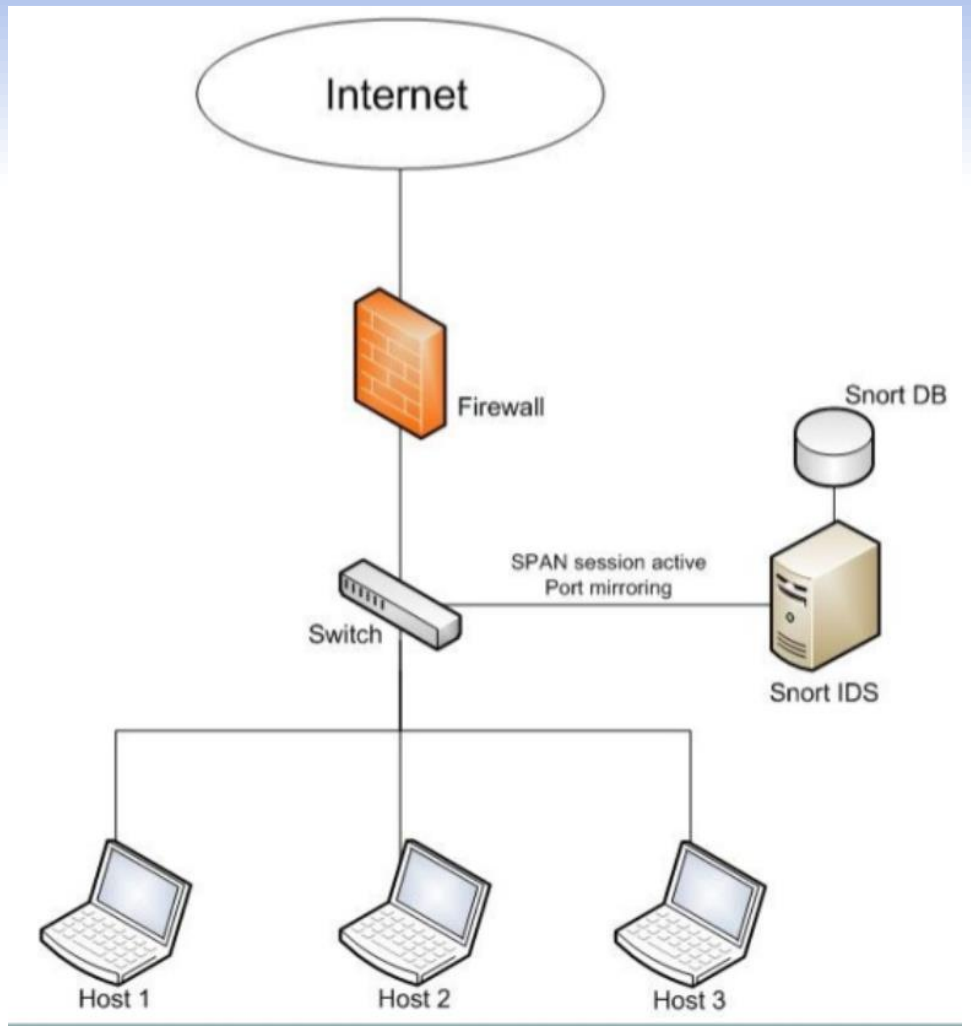
Example IDS/IPS

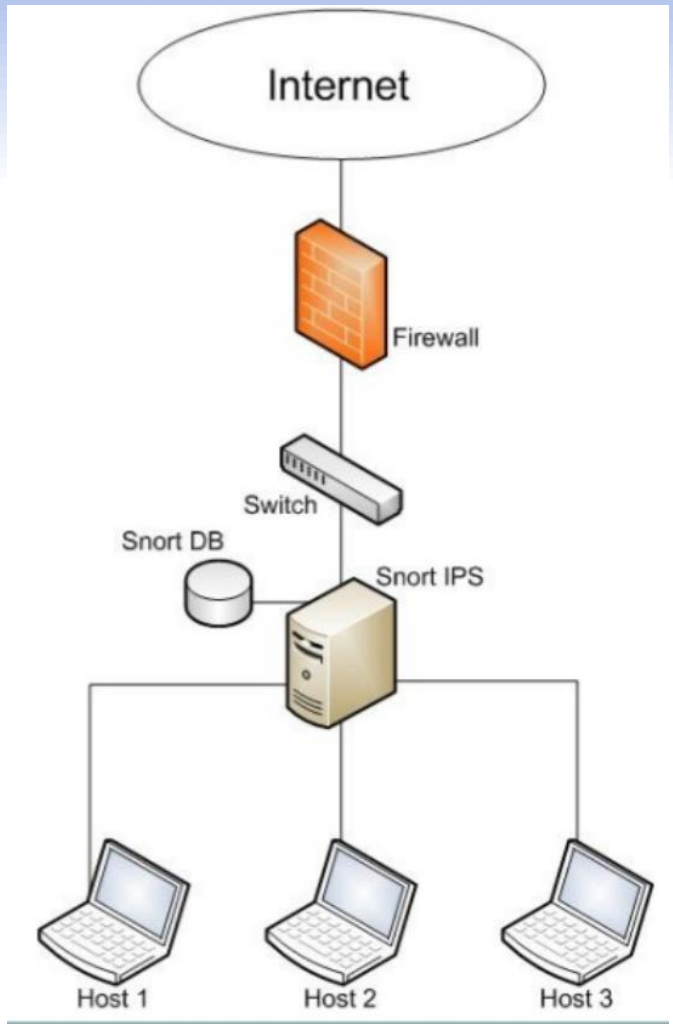
https://www.snort.org/

Open source network intrusion detection/intrusion preventions system

© Chad Williams Nov-17 Adapted slides from Pearson Higher Ed, copyright 2014

IDS vs IPS network





Author: Simone Tino

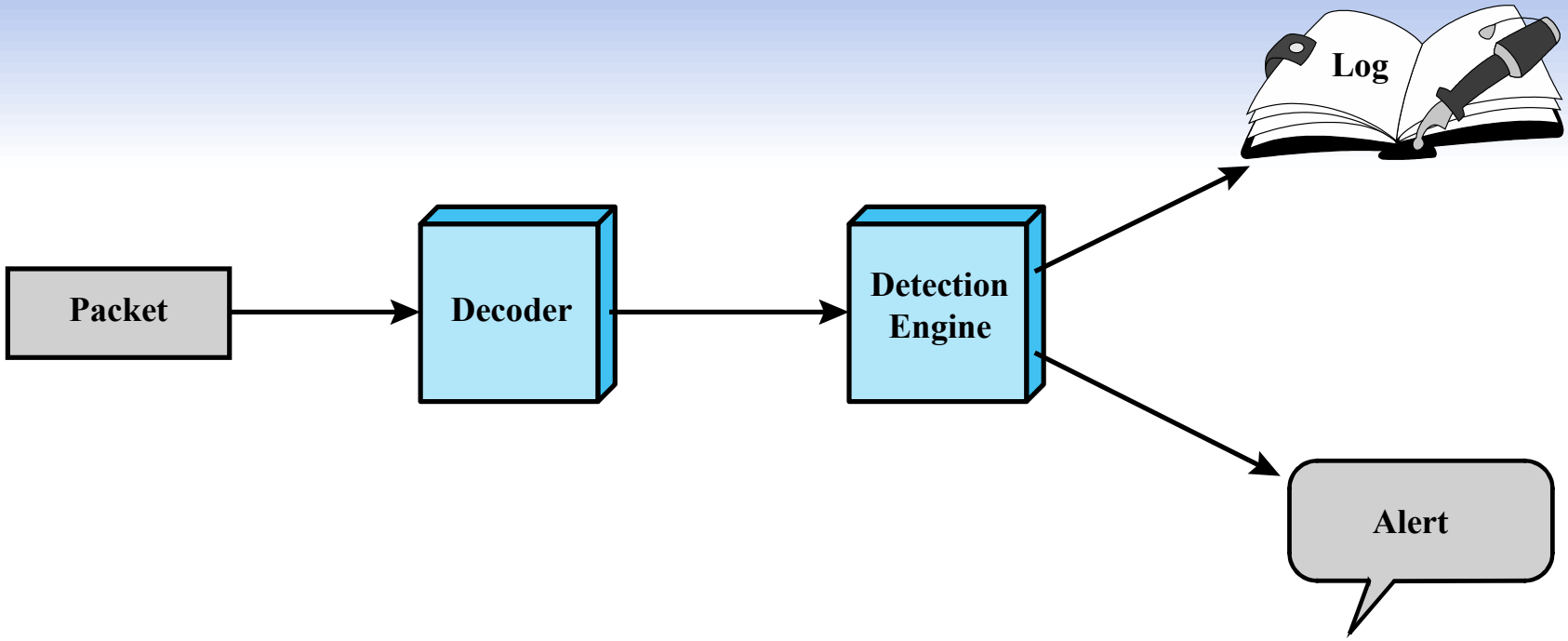


Figure 8.9 Snort Architecture

© Chad Williams Nov-17 Adapted slides from Pearson Higher Ed, copyright 2014

Action	Protocol	Source IP address	Source Port	Direction	Dest IP address	Dest Port

(a) Rule Header

Option Keyword	Option Arguments	• • •

(b) Options

Figure 8.10 Snort Rule Formats

© Chad Williams Nov-17 Adapted slides from Pearson Higher Ed, copyright 2014

Table 8.3
Snort Rule Actions

Action	Description
alert	Generate an alert using the selected alert method, and then log the packet.
log	Log the packet.
pass	Ignore the packet.
activate	Alert and then turn on another dynamic rule.
dynamic	Remain idle until activated by an activate rule , then act as a log rule.
drop	Make iptables drop the packet and log the packet.
reject	Make iptables drop the packet, log it, and then send a TCP reset if the protocol is TCP or an ICMP port unreachable message if the protocol is UDP.
sdrop	Make iptables drop the packet but does not log it.

© Chad Williams Nov-17 Adapted slides from Pearson Higher Ed, copyright 2014

Snort rules options

meta-data
msg Defines the message to be sent when a packet generates an event. **reference** Defines a link to an external attack identification system, which provides additional information. **classtype** Indicates what type of attack the packet attempted.
payload
content Enables Snort to perform a case-sensitive search for specific content (text and/or binary) in the packet payload. **depth** Specifies how far into a packet Snort should search for the specified pattern. Depth modifies the previous content keyword in the rule. **offset** Specifies where to start searching for a pattern within a packet. Offset modifies the previous content keyword in the rule. **nocase** Snort should look for the specific pattern, ignoring case. Nocase modifies the previous content keyword in the rule.

© Chad Williams Nov-17 Adapted slides from Pearson Higher Ed, copyright 2014

Snort rules options

non-payload
ttl Check the IP time-to-live value. This option was intended for use in the detection of traceroute attempts.
id Check the IP ID field for a specific value. Some tools (exploits, scanners and other odd programs) set this field specifically for various purposes, for example, the value 31337 is very popular with some hackers.
dsize Test the packet payload size. This may be used to check for abnormally sized packets. In many cases, it is useful for detecting buffer overflows.
flags Test the TCP flags for specified settings.
seq Look for a specific TCP header sequence number.
icmp-id Check for a specific ICMP ID value. This is useful because some covert channel programs use static ICMP fields when they communicate. This option was developed to detect the stacheldraht DDoS agent.
post-detection
logto Log packets matching the rule to the specified filename.
session Extract user data from TCP Sessions. There are many cases where seeing what users are typing in telnet, rlogin, ftp, or even web sessions is very useful.

Example rule

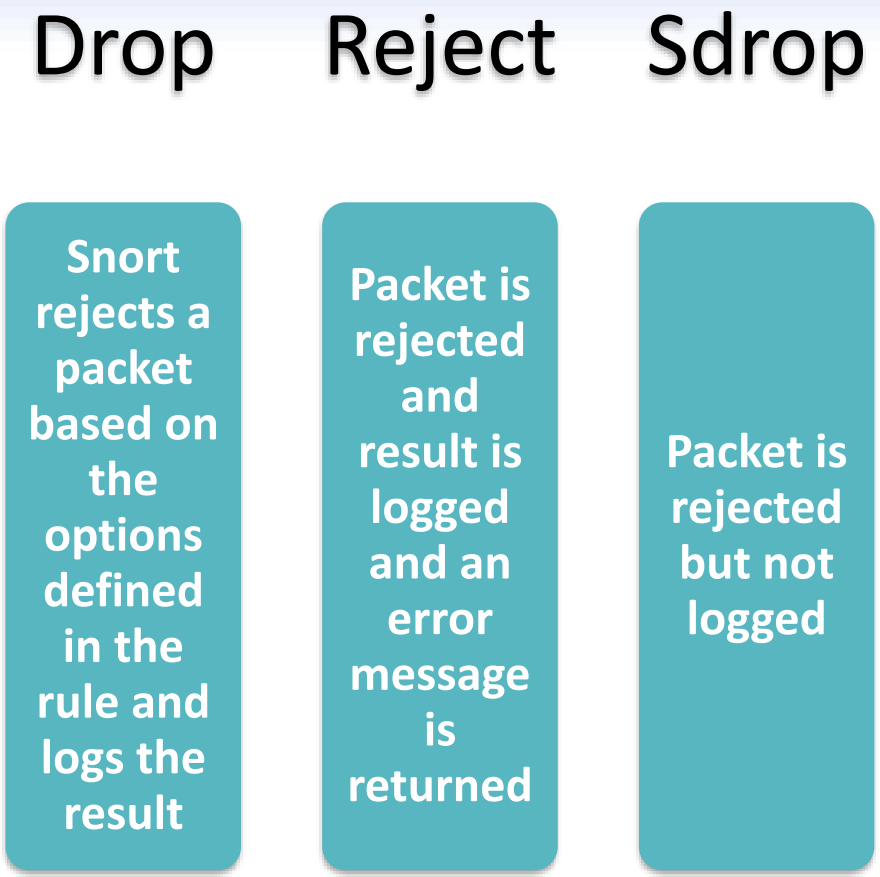
EXAMPLE

Rule Header	`alert tcp $EXTERNAL_NET $HTTP_PORTS -> $HOME_NET any`
Message	`msg: "BROWSER-IE Microsoft Internet Explorer CacheSize exploit attempt";`
Flow	`flow: to_client,established;`
Detection	`file_data; content:"recordset"; offset:14; depth:9; content:".CacheSize"; distance:0; within:100; pcre:"/CacheSize\s*=\s*/"; byte_test:10,>,0x3ffffffe,0,relative,string;`
Metadata	`policy max-detect-ips drop, service http;`
References	`reference:cve,2016-8077;`
Classification	`classtype: attempted-user;`
Signature ID	`sid:65535;rev:1;`

© Chad Williams Nov-17 Adapted slides from Pearson Higher Ed, copyright 2014

Snort Inline

- Enables Snort to function as an intrusion prevention system
- Includes a replace option which allows the Snort user to modify packets rather than drop them
  - Useful for a honeypot implementation
  - Attackers see the failure but cannot figure out why it occurred



Summary

- Intruders
  - Intruder behavior
- Intrusion detection
  - Basic principles
- Analysis approaches
  - Anomaly detection
  - Signature or heuristic detection
- Honeypots

- Host-based intrusion detection
  - Data sources and sensors
- Network-based intrusion detection
  - Types of network sensors
  - NIDS/IPS sensor deployment
- Example system: Snort
  - Snort architecture
  - Snort rules