Module 16

Firewalls and IDS

Ansh Bhawnani

IDS, Firewall and Honeypot Concepts

Module 16

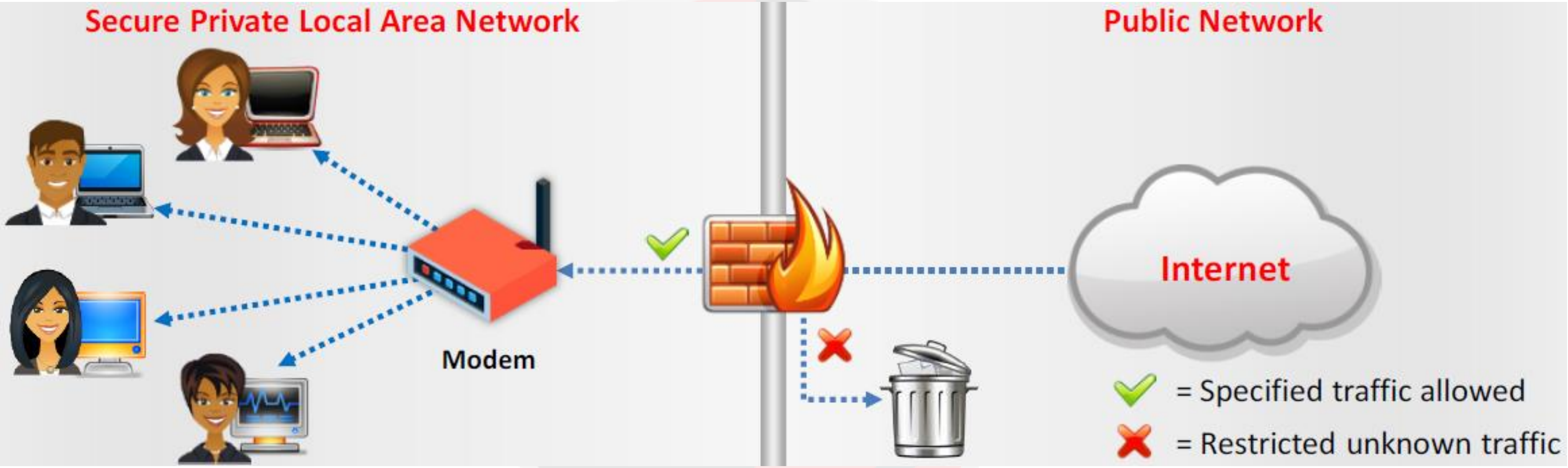
1. Firewalls

Module 16

IDS, Firewall and Honeypot Concepts

- Firewall are hardware and/or software designed to prevent unauthorized access to or from a private network.
- They are placed at the junction or gateway between the two networks, which is usually a private network and a public network such as the Internet.
- Firewall examine all messages entering or leaving the Intranet and blocks those that do not meet the specified security criteria.
- Firewalls may be concerned with the type of traffic or with the source or destination addresses and ports.

IDS, Firewall and Honeypot Concepts


Secure Private Local Area Network
Public Network
Modem
Internet
✓ = Specified traffic allowed
✗ = Restricted unknown traffic

2. Firewall Architecture

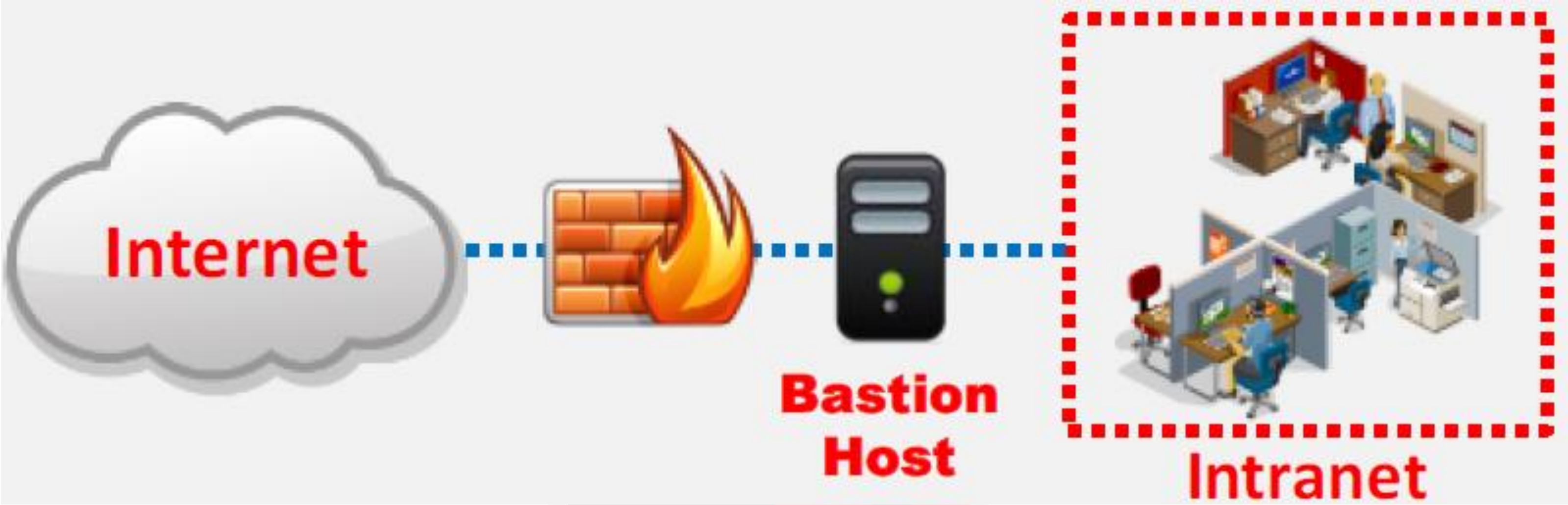
Module 16

IDS, Firewall and Honeypot Concepts

Bastion Host:

- Bastion host is a computer system designed and configured to protect network resources from attack.
- Traffic entering or leaving the network passes through the firewall, it has two interfaces:
  - public interface directly connected to the Internet.
  - private interface connected to the Intranet.

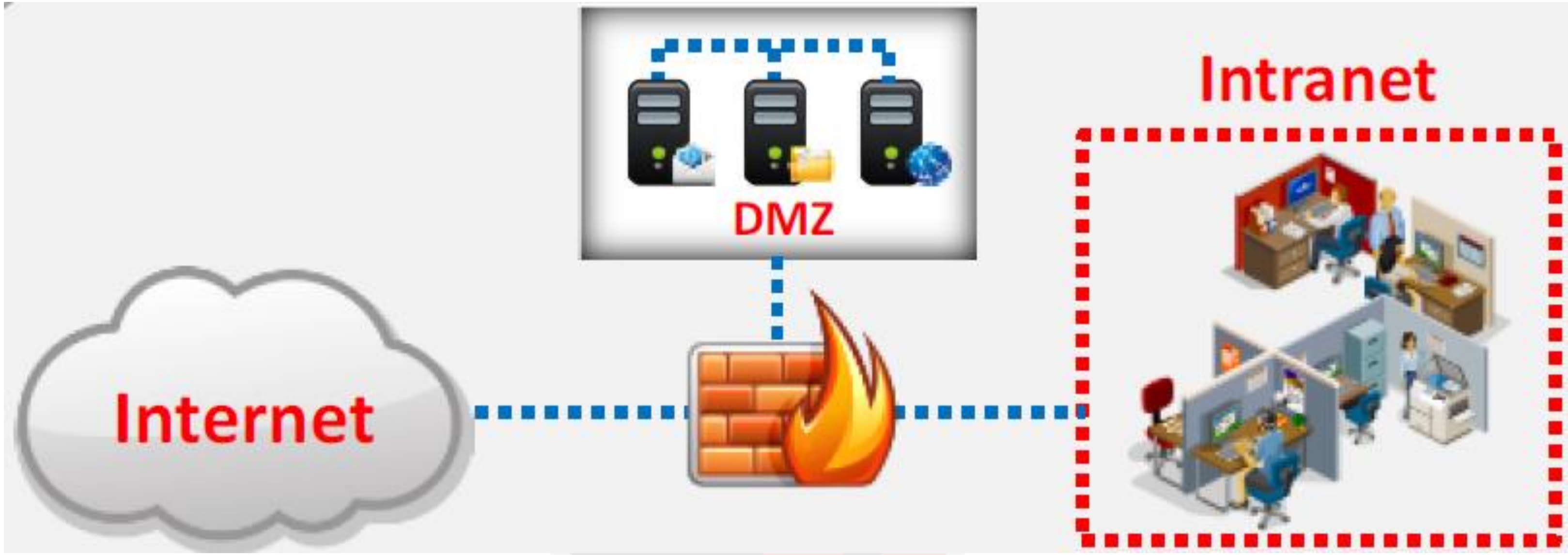
IDS, Firewall and Honeypot Concepts



IDS, Firewall and Honeypot Concepts

- **Screened Subnet or DeMilitarized Zone**:
  - The screened subnet or DMZ (additional zone) contains hosts that offer public services.
  - The DMZ zone responds to public requests, and has no hosts accessed by the private network.
  - Private zone can not be accessed by Internet users.

IDS, Firewall and Honeypot Concepts



IDS, Firewall and Honeypot Concepts

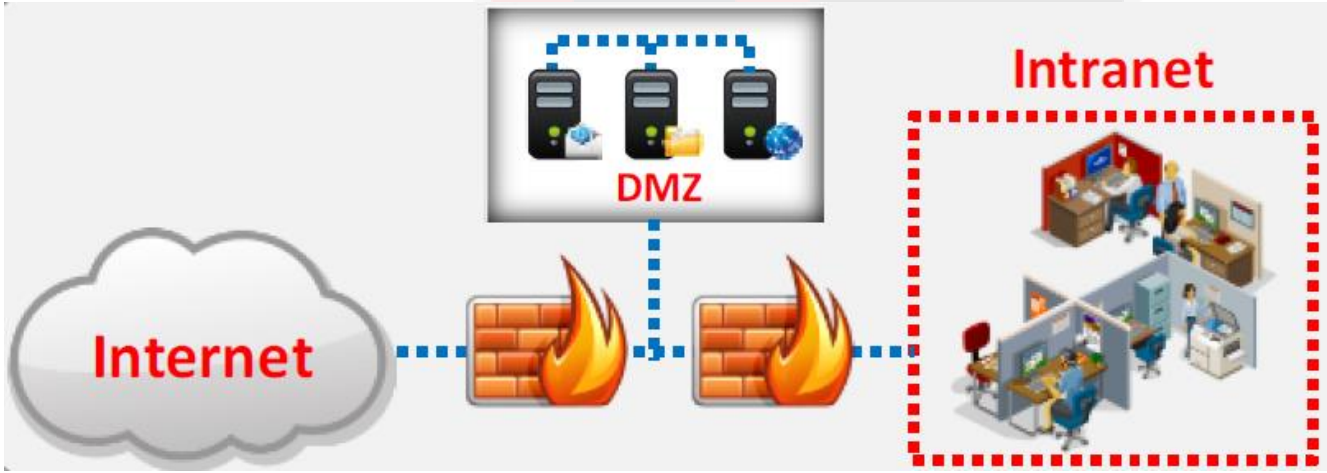
DeMilitarized Zone:

- DMZ exposes an organization's external-facing services to an untrusted network, while rest of the organization's network is firewalled.
- DMZ is a network that serves as a buffer between the internal secure network and insecure Internet.
- It can be created using firewall with three or more network interfaces assigned with specific roles such as Internal trusted network, DMZ network, and external un-trusted network.

IDS, Firewall and Honeypot Concepts

Multi-homed Firewall:

- In this case, a firewall with two or more interfaces is present that allows further subdivision of the network based on the specific security objectives of the organization.



3. Types of Firewalls

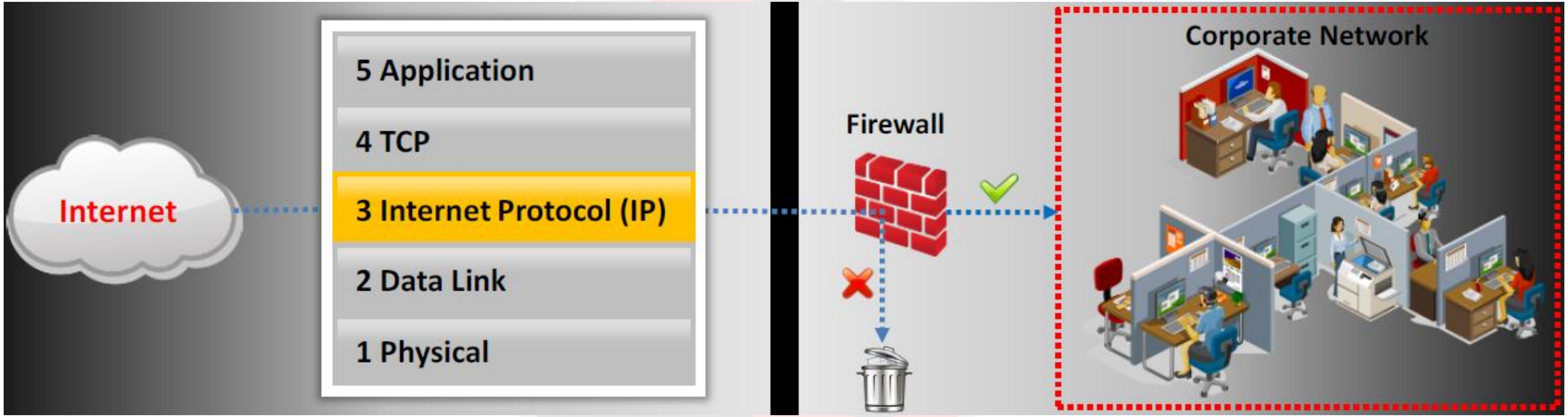
Module 16

IDS, Firewall and Honeypot Concepts

Packet Filtering Firewall

- Packet filtering firewalls work at the network layer of the OSI model (or the IP layer or TCP/IP), they are usually a part of a router.
- In a packet filtering firewall, each packet is compared to a set of criteria before it is forwarded.
- Depending on the packet and the criteria, the firewall can drop the packet and forward it, or send a message to the originator.
- Rules can include the source and the destination IP address, the source and the destination port number, and the protocol used.

IDS, Firewall and Honeypot Concepts

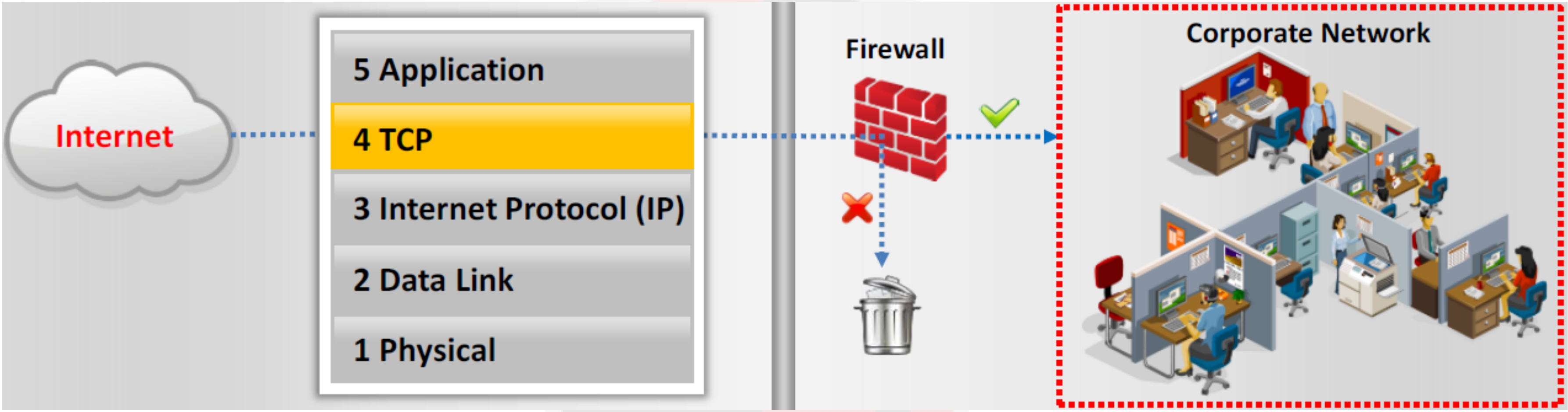


IDS, Firewall and Honeypot Concepts

Circuit-Level Gateway Firewall

- Circuit-level gateways work at the session layer of the OSI model (or the TCP layer of TCP/IP)
- Information passed to a remote computer through a circuit-level gateway appears to have originated from the gateway.
- They monitor requests to create sessions, and determine if those sessions will be allowed.
- Circuit proxy firewalls allow or prevent data streams, they do not filter individual packets.

IDS, Firewall and Honeypot Concepts

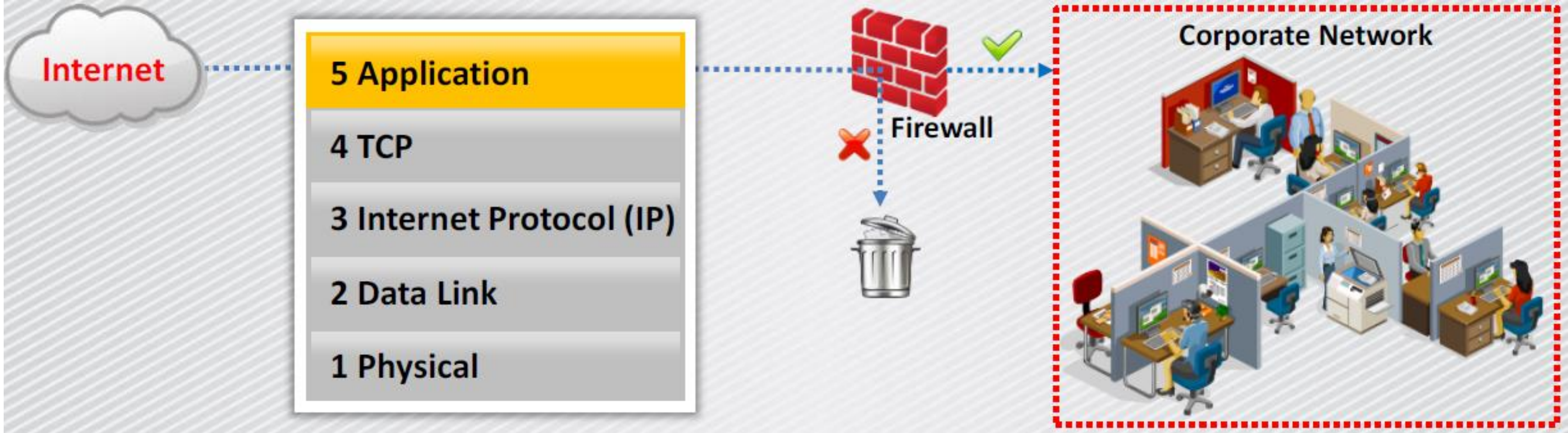


IDS, Firewall and Honeypot Concepts

Application-Level Firewall

- Application-level gateways (proxies) can filter packets at the application layer of the OSI model (or the application layer of TCP/IP).
- Incoming and outgoing traffic is restricted to services supported by proxy; all other service requests are denied.
- Application-level gateways configured as a web proxy prohibit FTP, gopher, telnet, or other traffic.
- Application-level gateways examine traffic and filter on application-specific commands such as http:post and get.

IDS, Firewall and Honeypot Concepts



IDS, Firewall and Honeypot Concepts

Application-Level Firewall

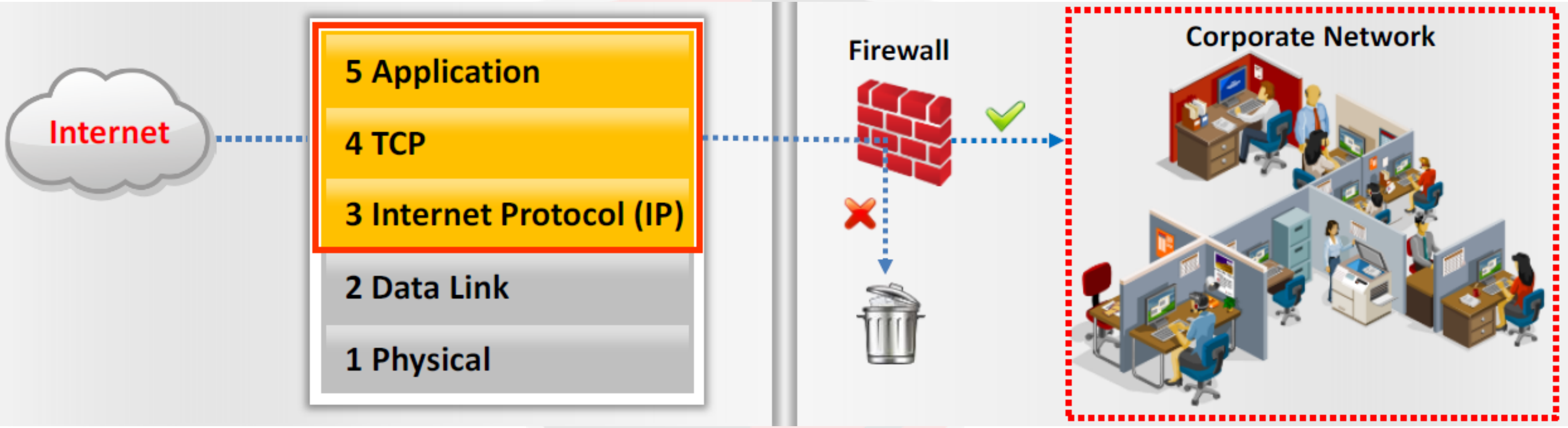
- Application-layer firewalls can function in one of two modes:
  - **Active application-level firewalls**: They examine all incoming requests, including the actual message that exchanged against known vulnerabilities, such as SQL injection, parameter and cookie tampering, and cross-site scripting. The requests deemed genuine and allowed to pass through them.
  - **Passive application-level firewalls**: They work similarly to an IDS, in that they also check all incoming requests against known vulnerabilities, but they do not actively reject or deny request if a potential attack is discovered.

IDS, Firewall and Honeypot Concepts

Stateful Multilayer Inspection Firewall (?)

- Stateful multilayer inspection firewalls combine the aspects of the other three types of firewalls.
- They filter packets at the network layer of the OSI model (or the IP layer of TCP/IP), to determine whether session packets are legitimate, and they evaluate the contents of packets at the application layer.
- SPI makes decisions also on the SYN, ACK, sequence numbers and other data contained in the TCP header.
- SPI firewalls track the state of each session and can dynamically open and close ports as specific sessions require.

IDS, Firewall and Honeypot Concepts



IDS/IPS Systems

Module 16

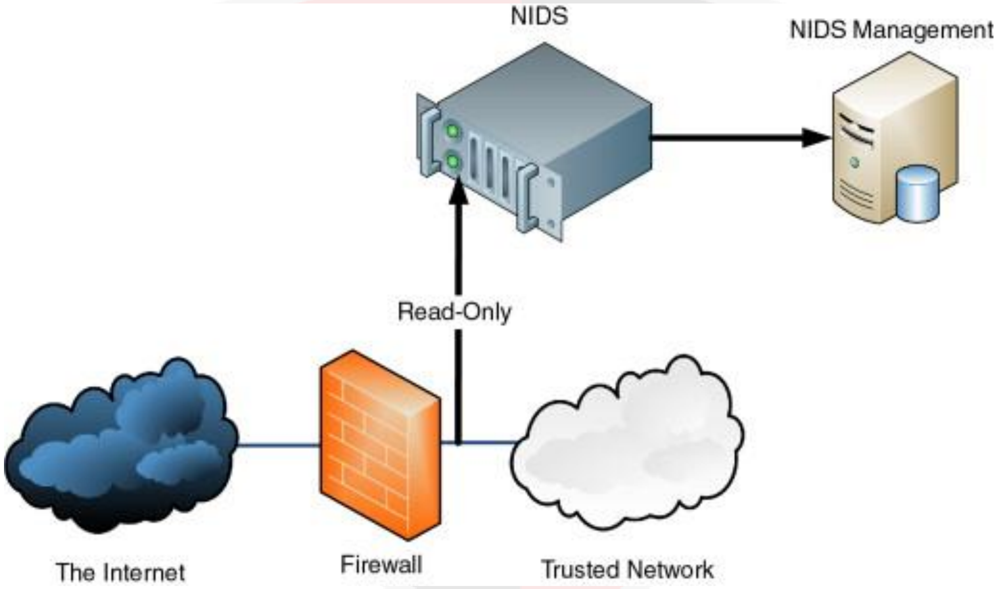
1. Intrusion Detection Systems (IDS)

Module 16

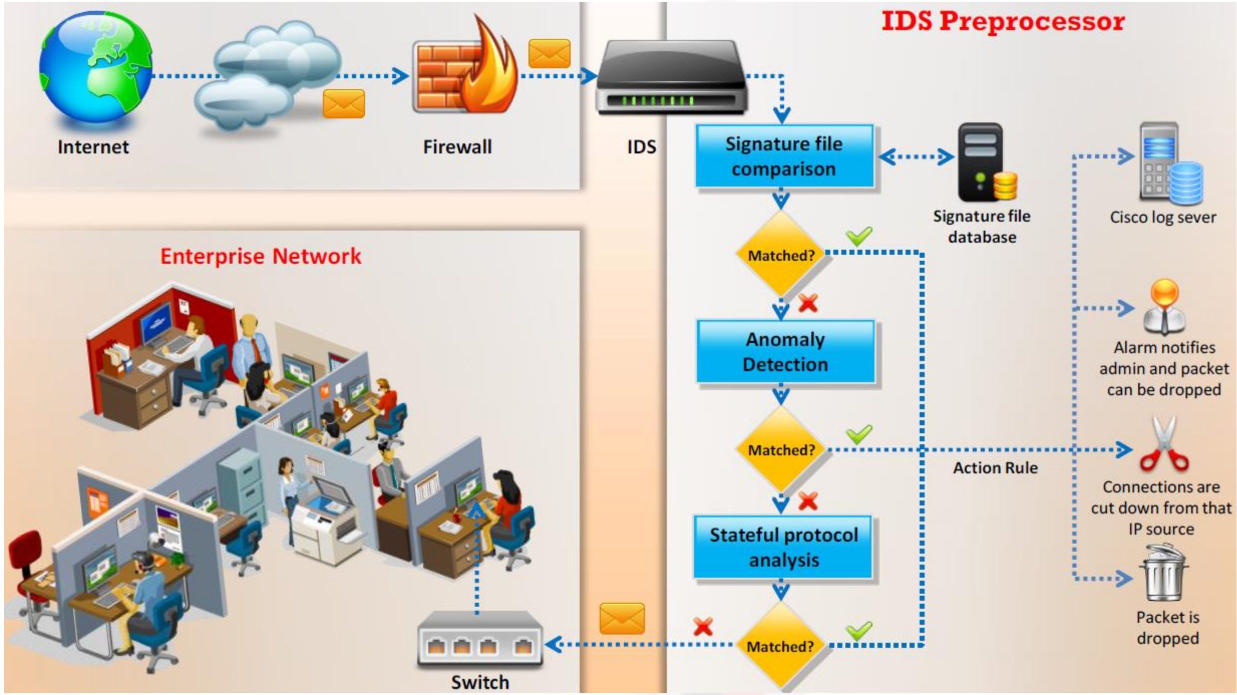
IDS, Firewall and Honeypot Concepts

- An intrusion detection system (IDS) inspects all inbound and outbound network traffic for suspicious patterns that may indicate a network or system security breach.
- The IDS checks traffic for signatures that match known intrusion patterns, and signals an alarm when a match is found.
- It needs to be properly set up to recognize what normal traffic on the network looks like as compared to malicious activity, to avoid false alarms.

IDS, Firewall and Honeypot Concepts



IDS, Firewall and Honeypot Concepts



2. Ways to Detect an Intrusion

Module 16

IDS, Firewall and Honeypot Concepts

Signature based Recognition

- Signature-based IDS detects the attacks on the basis of the specific patterns such as number of bytes or number of 1's or number of 0's in the network traffic.
- It also detects on the basis of the already known malicious instruction sequence that is used by the malware. It uses a database of 1000s of predefined signatures.
- **Advantage**: High processing rate and quick response
- **Drawback**: Only detects already known patterns, fails to detect zero-day exploits

IDS, Firewall and Honeypot Concepts

Anomaly based Recognition

- Primarily introduced to detect unknown attacks, it uses *machine learning* to create a trustful activity model and anything coming is compared with that model and it is declared suspicious if it is not found in model.
- The classification is based on heuristics or rules, rather than patterns or signatures, and attempts to detect any type of misuse that falls out of normal system operation.
- **Advantage**: Can detect zero-day attacks
- **Drawback**: False positives, resource and time intensive

IDS, Firewall and Honeypot Concepts

Stateful Protocol based Recognition

- Identifies deviations of protocol state similarly to the anomaly-based method but uses predetermined universal profiles based on "accepted definitions of benign activity" developed by vendors and industry leaders.
- Monitoring requests with its corresponding response; every request should have a predictable response and those responses that fall outside of expected results will be flagged and analyzed further.
- **Advantage**: Identifies unexpected sequences of commands
- **Drawback**: Resource intensive - lots of overhead, cannot detect attacks that do not violate the characteristics of acceptable protocol behavior

3. General Indications of Intrusions

Module 16

IDS, Firewall and Honeypot Concepts

- **System Intrusions**:
  - The presence of new, unfamiliar files, or programs.
  - Changes in file permissions.
  - Unexplained changes in a file's size.
  - Rogue files on the system that do not correspond to your master list of signed files.
  - Unfamiliar file names in directories.
  - Missing files.

IDS, Firewall and Honeypot Concepts

- **Network Intrusions**:
  - Repeated probes of the available services on your machines.
  - Connections from unusual locations.
  - Repeated login attempts from remote hosts.
  - Arbitrary data in log files, indicating attempts to cause a DoS or to crash a service.

IDS, Firewall and Honeypot Concepts

General Indications of System Intrusions

- Short or incomplete logs
- Unusual graphic displays or text messages
- Unusually slow system performance
- Modifications to system software and configuration files
- Missing logs or logs with incorrect permissions or ownership
- System crashes or reboots
- Gaps in the system accounting
- Unfamiliar processes

IDS, Firewall and Honeypot Concepts

Types of Intrusion Detection Systems

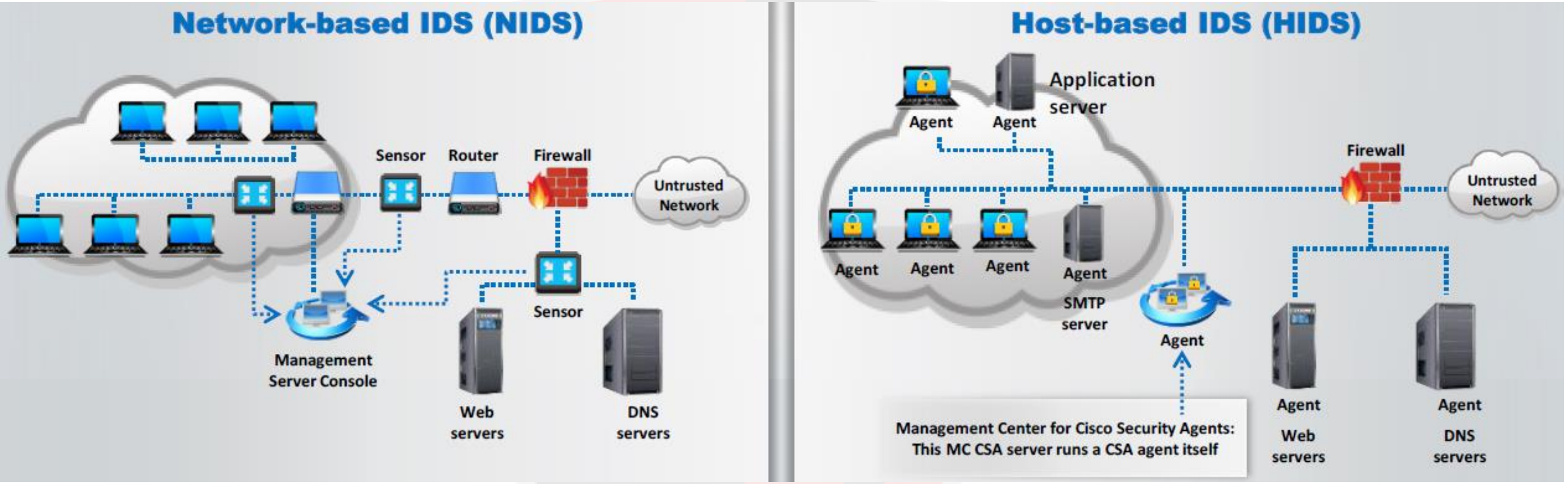
- **Network-Based Intrusion Detection Systems:**
  - These mechanisms typically consist of a black box that is placed on the network in the promiscuous mode, listening for patterns indicative of an intrusion.
  - It detects malicious activity such as Denial-of-Service attacks, port scans, or even attempts to crack into computers by monitoring network traffic.

IDS, Firewall and Honeypot Concepts

Host-Based Intrusion Detection Systems:

- These mechanisms usually include auditing for events that occur on a specific host.
- These are not as common, due to the overhead they incur by having to monitor each system event.

IDS, Firewall and Honeypot Concepts



IDS, Firewall and Honeypot Concepts

System Integrity Verifiers (SIV)

- System Integrity Verifiers detect changes in critical system components which help in detecting system intrusions.
- SIVs compares a snapshot of the file system with an existing baseline snapshot.

4. IDS vs Firewalls vs IPS

Module 16

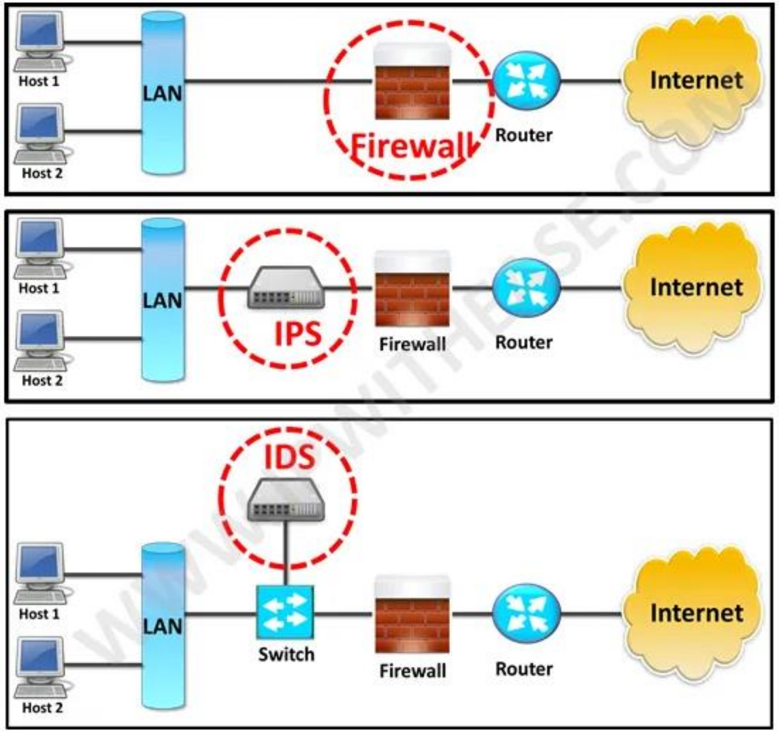
IDS, Firewall and Honeypot Concepts

Parameter	Firewall	IPS	IDS
Philosophy	Firewall is a network security device that filters incoming and outgoing network traffic based on predetermined rules	IPS is a device that inspects traffic, detects it, classifies and then proactively stops malicious traffic from attack.	It is a device or software application that monitors a traffic for malicious activity or policy violations and sends alert on detection.
Principle of working	Filters traffic based on IP address and port numbers	Inspects real time traffic and looks for traffic patterns or signatures of attack and then prevents on detection	Detects real time traffic and looks for traffic patterns or signatures of attack and them generates alerts
Configuration mode	Layer 3 mode or transparent mode	Inline mode, generally being in layer 2	Inline mode, generally being in layer 2
Placement	Inline at the Perimeter of Network	Inline generally after Firewall	Non-Inline through port span (or via tap)

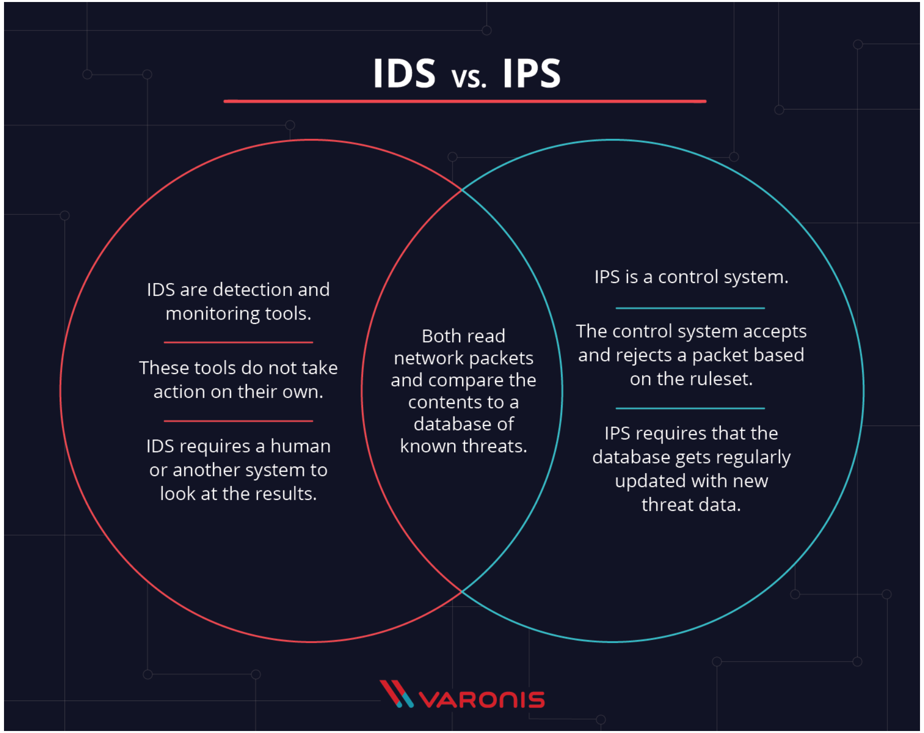
IDS, Firewall and Honeypot Concepts

Parameter	Firewall	IPS	IDS
Traffic patterns	Not analyzed	Analyzed	Analyzed
Placement wrt each other	Should be 1st Line of defense	Should be placed after the Firewall device in network	Should be placed after firewall
Action on unauthorized traffic detection	Block the traffic	Preventing the traffic on Detection of anomaly	Alerts/alarms on detection of anomaly
Related terminologies	> Stateful packet filtering > permits and blocks traffic by port/protocol rules	> Anomaly based detection > Signature detection > Zero day attacks > Blocking the attack	> Anomaly based detection > Signature detection > Zero day attacks > Monitoring > Alarm

IDS, Firewall and Honeypot Concepts



IDS, Firewall and Honeypot Concepts



IDS vs. IPS

IDS are detection and monitoring tools.

These tools do not take action on their own.

IDS requires a human or another system to look at the results.

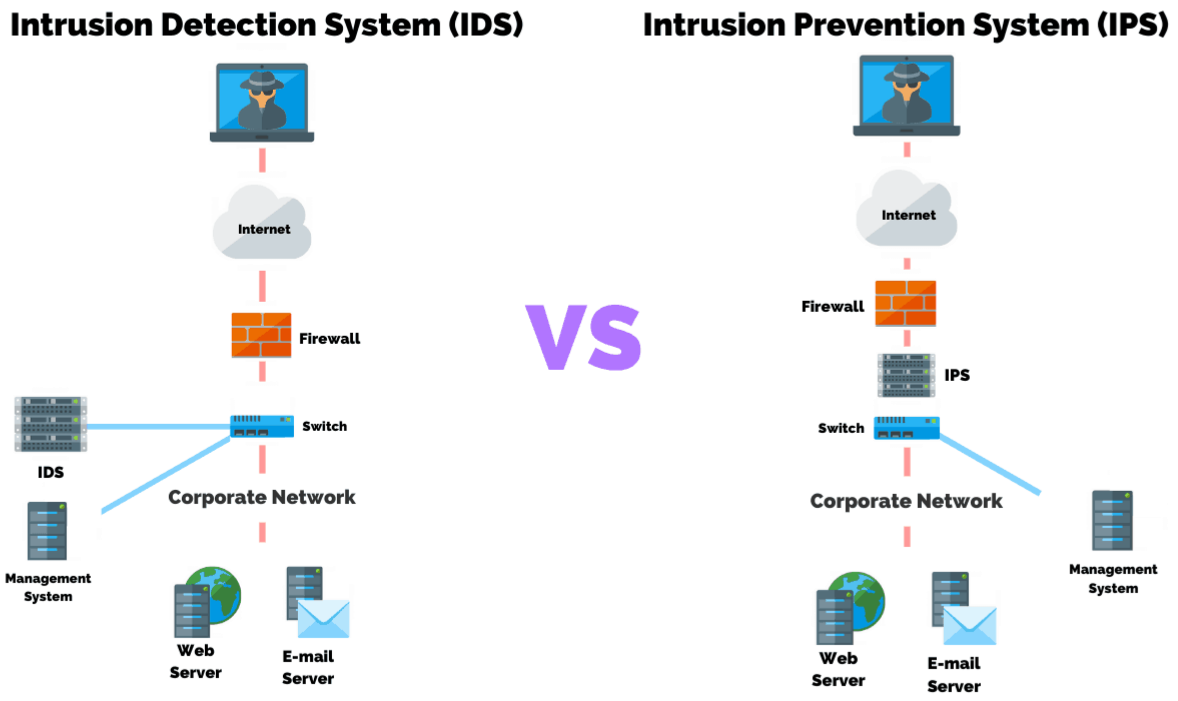
Both read network packets and compare the contents to a database of known threats.

IPS is a control system.

The control system accepts and rejects a packet based on the ruleset.

IPS requires that the database gets regularly updated with new threat data.

IDS, Firewall and Honeypot Concepts



5. Honeypots

Module 16

Honeypots

- A honeypot is “an information system resource whose value lies in unauthorized or illicit use of that resources” - **Lance Spitzner**
- “A server that is configured to detect an intruder by mirroring a real production system. It appears as an ordinary server doing work, but all the data and transactions are phony. Located either in or outside the firewall, the honeypot is used to learn about an intruder's techniques as well as determine vulnerabilities in the real system“
- A honeypot is a company saying: “*Look at me, I’m vulnerable and full of confidential information – why don’t you attack me, instead of our real systems?*”.

Honeypots

- There is a whole spectrum of why you would want a honeypot, some of them would be:
  - *Research Exploits*
  - Find *Zero-Day* Exploits
  - ***Learn*** more ***about*** your actual system (if the honeypot is a copy of your actual system)
  - ***Learn*** about the ***types of attack*** that your real system is vulnerable to and how to best protect it.

Honeypots

- A honeypot will provide the:
  - ***What*** – What did they use to attack/exploit?
  - ***How*** – How did they attack/exploit?
  - ***Motives*** – Why would they attack/exploit us?

Honeypots

- With that being said, there are two types of honeypots:
  - **Corporate honeypot** – This is a honeypot that is set up in a production environment and serves as a tool for studying attacks with the purpose of using the knowledge to further strengthen the network's security.
  - **Honeypots for Research:** This type of honeypot is more focused on researching the motives of an attacker. This typically use different configurations to lure the attackers in. ***For example***, a research to find out what type of exploits people on the internet would throw at this specific system, and to create defensive solutions for future.

Honeypots

- The data types that honeypots capture from (or about) the attackers can include, but is not limited to:
  - The **usernames**, **roles**, and **privileges** that the attackers use
  - The **IP addresses** of the network or host that are being using for the attack
  - What data is being **accessed**, **altered** or **deleted**
  - The **actual keystrokes** the attackers type out, which lets administrators see exactly what they are doing

Honeypots

- **Pros of using a honeypot network**
  - It is a low-cost security measure that could yield high-value information about your attackers.
  - Honeypots are arguably **the best way to catch a hacker or an attack just as it is happening. It allows administrators to go through the whole process step-by-step, following it all in real-time** with each alert.
- **Cons of using a honeypot network**
  - It is not easy to set up and configure and it would be pure insanity to try and do so without an expert on hand; it could backfire and expose a network to worse attacks.

Honeypots

Honeypot Strategies

- **Low-interaction method**
  - In this method you will be using **fake data, folders, and databases as bait with the intent of monitoring attacks to see what would happen in a real-life data breaching scenario**.
  - Of course, they would have access to other peripheral information sets like ***IP addresses***, ***usernames***, and ***passwords*** – over which the administrators keep a keen eye.
  - It only provides certain fake services but it's no real operating system that an attacker can operate on, were designed to emulated vulnerable services

Honeypots

Honeypot Strategies

High-interaction method

- In this setup you would **allow the attackers to interact with data, software (including OS), services, and hardware that appear to be as realistic as possible. The intent here is to gauge and capture the skills of the attackers**.
- This setup is mostly used in research scenarios where the results of the studies are used to improve the defense capabilities of *anti-viruses* and *anti-malware*.
- Nothing in a high interaction honeypot is emulated, its all real. Therefore, a higher complexity and maintenance is involved.

Honeypots

False positives

- A honeypot alert is not **fool-proof**. When it comes to honeypot alerts, beware of a different kind of false positive.
- **For instance**: an attacker can create a *diversion*, spoofing your production system pretending that they are attacking the honeypot. Meanwhile, your honeypot would detect these spoofed attacks as actual attacks. This would drive your IT admins to investigate the wrong attack.
- Meanwhile, during this false alert, an attacker would be focusing on a real attack against the production system.

Honeypots

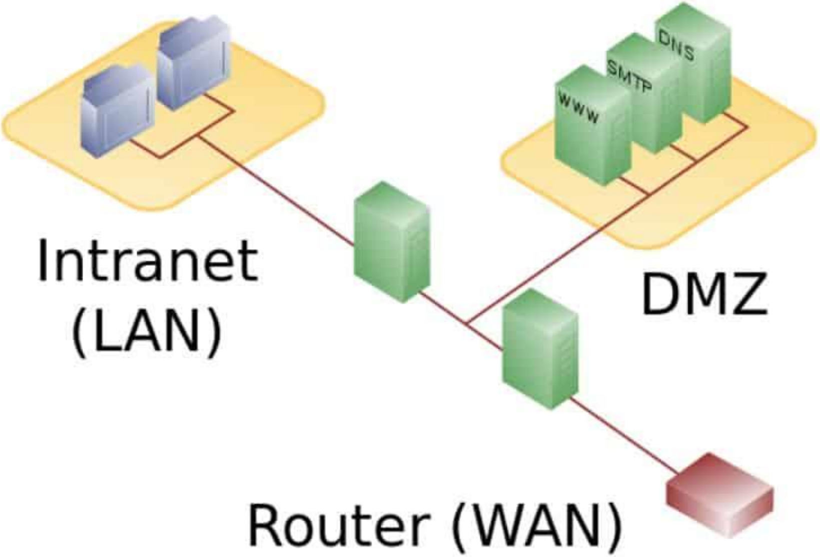
- Being discovered – by the intruders
  - If the network is too easy to attack, or gain access to,
  - If there are too many unnecessary services running, or too many ports open,
  - If the configurations of the running software solutions are still in their default settings,
  - If there is little-to-no traffic passing through the network
  - If too much effort has been put into making it look like they walked into a candy store
  - If the servers appear to be empty or have lots of free diskspaces

Honeypots

- Protect yourself well!
  - Never use real data
  - Never connect your honeypot to your main network
  - Use virtual machines
  - Firewalls and routers should be the only way to get to your honeypot
  - Usernames and roles should be unique to the honeypot
  - Always test, test, and test

Honeypots

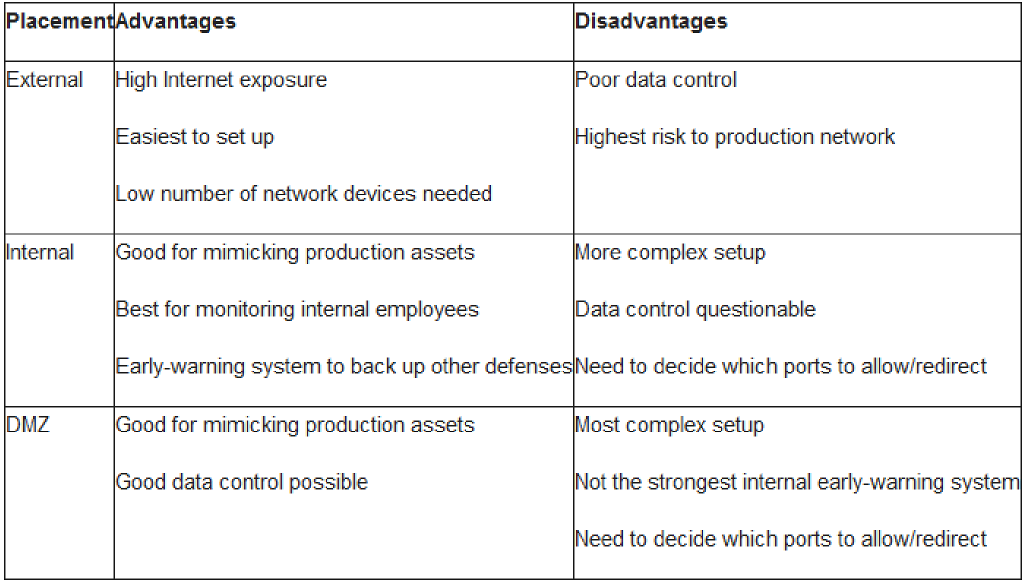
- Placement of the honeypot



Honeypots

Placement of the honeypot

Table 2-1: Honeypot Placement Location Comparison



Placement	Advantages	Disadvantages
External	High Internet exposure Easiest to set up Low number of network devices needed	Poor data control Highest risk to production network
Internal	Good for mimicking production assets Best for monitoring internal employees Early-warning system to back up other defenses	More complex setup Data control questionable Need to decide which ports to allow/redirect
DMZ	Good for mimicking production assets Good data control possible	Most complex setup Not the strongest internal early-warning system Need to decide which ports to allow/redirect

Honeypots

Honeytokens

- Honeytokens are **files or data sets that would appear to be interesting to the attacker but are actually fake replicas of the real deal**.
- The **honeytokens can also be embedded files or data sets in what would otherwise appear to be a legitimate server or database**. It makes it easy for administrators to keep track of the data in case it is stolen –
- **Examples of this sort of honeytoken include email addresses and usernames or login IDs**. If an attacker gains access to these pieces of information, it would be easy to know which database they have breached which would, in turn, help in figuring out how they managed to do it.

6. Intrusion Detection Tool: Snort

Module 16

IDS Systems

Snort

- Snort is an open source network intrusion detection system, capable of performing real-time traffic analysis and packet logging on IP networks.
- It can perform protocol analysis and content searching/matching, and is used to detect a variety of attacks and probes, such as buffer overflows, stealth port scans, CGI attacks, SMB probes, and OS fingerprinting attempts.
- It uses flexible rules language to describe traffic that it should collect or pass, as well as a detection engine that utilizes a modular plug-in architecture.

IDS Systems

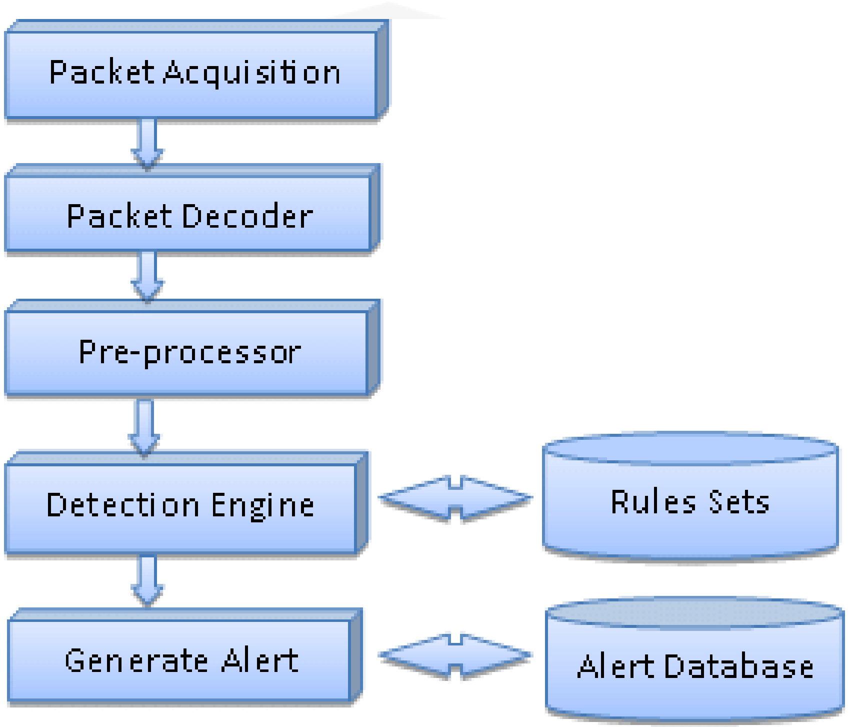
Uses of Snort:

- Straight packet sniffer like tcpdump
- Packet logger (useful for network traffic debugging, etc.)
- Network intrusion prevention system

5. How Snort Works

Module 16

IDS Systems


Packet Acquisition
Packet Decoder
Pre-processor
Detection Engine
Rules Sets
Generate Alert
Alert Database

6. Snort Rules

Module 16

IDS Systems

- Snort's rule engine enables custom rules to meet the needs of the network.
- Snort rules help in differentiating between normal Internet activities and malicious activities.
- Snort rules must be contained on a single line, the Snort rule parser does not handle rules on multiple lines.
- Snort rules with **two** logical parts:
  - **Rule header**: Identifies rule's actions such as alerts, log, pass, activate, dynamic, etc.
  - **Rule options**: Identifies rule's alert messages.

IDS Systems

Example:

- alert tcp any any -> 192.168.1.0/24 111 (content: "|00 01 86 a5|";msg: "mountd access";)
  - **alert**: Rule Action
  - **tcp**: Rule Protocol
  - ->: Rule Format Direction
  - **192.168.1.0/24**: Rule IP address
  - **111**: Rule Port
  - **content: "|00 01 86 a5|"**: Payload detection rule
  - **msg: "mountd access":** Alert message

IDS Systems

Rule Actions:

- The rule action tells Snort what to do when it finds a packet that matches the rule criteria.
- **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

IDS Systems

- **Protocols**:
  - The next field in a rule is the protocol. There are three IP protocols that Snort currently analyzes for suspicious behavior, tcp, udp, and icmp. In the future there may be more, such as ARP, IGRP, GRE, OSPF, RIP, IPX, etc.
    - **tcp**
    - **udp**
    - **icmp**

IDS Systems

IP Addresses and Port numbers:

- The next portion of the rule header deals with the IP address and port information for a given rule.
- The keyword "*any*" may be used to define any address. Snort does not have a mechanism to provide host name lookup for the IP address fields in the rules file.
- The addresses are formed by a straight numeric IP address and a ***CIDR*** block.
- ***E.g*, log udp any any -> 192.168.1.0/24 1:1024**

IDS Systems

The Direction Operator

- The *direction operator* "->" indicates the orientation, or "direction", of the traffic that the rule applies to.
- The information on the left side of the direction operator is considered to be the traffic coming from the source host, and information on the right side of the operator is the destination host. There is also a ***bidirectional operator***, which is indicated with a "<>" symbol.
- *E.g*, **log !192.168.1.0/24 any <> 192.168.1.0/24 23**

IDS Systems

Activate/Dynamic Rules

- Activate/dynamic rule pairs give Snort a powerful capability. You can now have one rule activate another when it's action is performed for a set number of packets. This is very useful if you want to set Snort up to perform follow on recording when a specific rule "goes off". Activate rules act just like alert rules, except they have a *required* option field: "activates“
- *For e.g,* **activate tcp !$HOME_NET any -> $HOME_NET 143 (flags: PA; content: "|E8C0FFFFFF|\bin|; activates: 1; msg: "IMAP buffer overflow!";)**

IDS Systems

Rule Options

- Rule options form the heart of Snort's intrusion detection engine, combining ease of use with power and flexibility.
- All Snort rule options are separated from each other using the ***semicolon*** ";" character. Rule option keywords are separated from their arguments with a *colon* ":".
- **For e.g., msg, logto, ttl, tos, id, fragbits, dsize, flag, resp, seq, ack, etc.**

Evading IDS

Module 16

1. Payload obfuscation

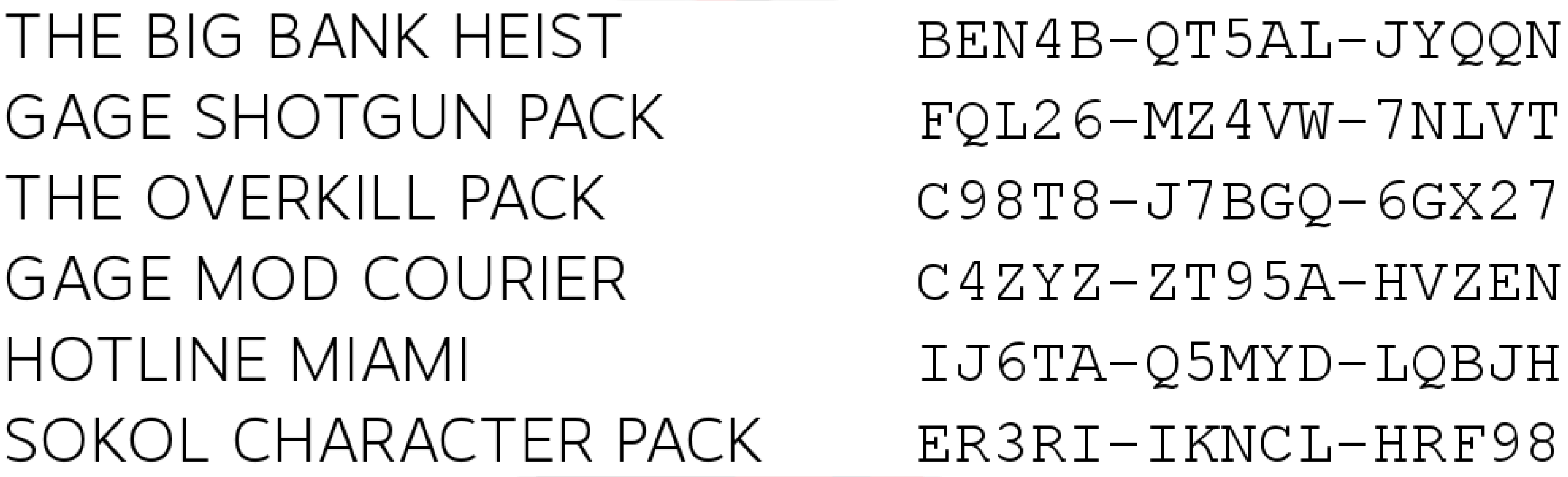
Module 16

Evading IDS

Encoding and encryption

- Application layer protocols like HTTP allow for multiple encodings of data which are interpreted as the same value. For example, the string "cgi-bin" in a URL can be encoded as "%63%67%69%2d%62%69%6e" (i.e., in hexadecimal). An IDS must be aware of all of the possible encodings that its end hosts accept.
- Attacks on encrypted protocols such as HTTPS cannot be read by an IDS unless the IDS has a copy of the private key used by the server to encrypt the communication. The IDS won't be able to match the encrypted traffic to signatures if it doesn't account for this.

Evading IDS



THE BIG BANK HEIST	BEN4B-QT5AL-JYQQN
GAGE SHOTGUN PACK	FQL26-MZ4VW-7NLVT
THE OVERKILL PACK	C98T8-J7BGQ-6GX27
GAGE MOD COURIER	C4ZYZ-ZT95A-HVZEN
HOTLINE MIAMI	IJ6TA-Q5MYD-LQBJH
SOKOL CHARACTER PACK	ER3RI-IKNCL-HRF98

Evading IDS

Polymorphism

- To obfuscate their attacks, attackers can use polymorphic shellcode to create unique attack patterns. This technique typically involves encoding the payload in some fashion (e.g., XOR-ing each byte with 0x95), then placing a decoder in front of the payload before sending it.
- When the target executes the code, it runs the decoder which rewrites the payload into its original form which the target then executes.
- Polymorphic attacks don't have a single detectable signature, making them very difficult for detection.
- Shikata ga nai ("it cannot be helped") is a popular polymorphic encoder in the Metasploit framework

2. Insertion Attacks

Module 16

Evading IDS

- Attacker tries to confuse the IDS by sending invalid packets
- An IDS blindly believes and accepts a packet that an end system rejects.
- An attacker exploits this condition and inserts data into the IDS.
- This attack occurs when NIDS is less strict in processing packets.
- Attacker obscures extra traffic and IDS concludes traffic is harmless.
- Hence, the IDS gets more packets than the destination.

Evading IDS

- Methods:
  - Fragmentation and small packets
  - Overlapping fragments and TCP segments
  - Protocol ambiguities
  - Low-bandwidth attacks

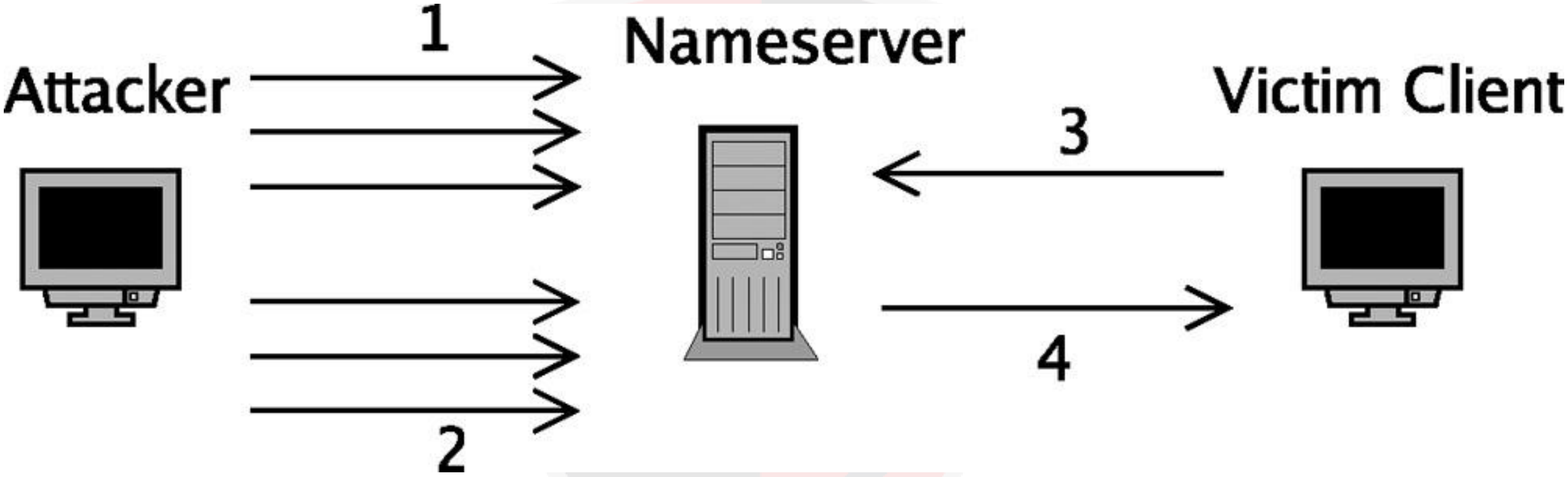
3. Denial of Service

Module 16

Evading IDS

- Due to the fact that passive IDS are inherently fail-open, launching a denial-of-service attack against the IDS on a network is a feasible method of circumventing its protection.
- It can be done by exploiting a bug in the IDS, consuming all of the computational resources on the IDS, or deliberately triggering a large number of alerts to disguise the actual attack.
- If the attackers know the IP address of this centralized logging server, they can launch a denial-of-service attack on that server so that the IDS won’t be able to log any more events.

Evading IDS



Evading IDS

- **Methods:**
  - CPU exhaustion
  - Memory exhaustion
  - Operator fatigue

4. Session Splicing

Module 16

Evading IDS

- Attacker splits the attack traffic in to many packets such that no single packet triggers the IDS.
- It is effective against IDSs that do not reconstruct packet before checking them against intrusion signatures.
- If attackers are aware of delay in packet reassembly, they can add delays between packet transmissions to bypass the reassembly.
- IDS will stop working if the target host keeps session active for a time longer than the IDS reassembly time.
- Any attack attempt after a successful splicing attack will not be logged by the IDS.

Evading Firewalls

Module 16

1. SSH Tunneling

Module 16

Evading Firewalls

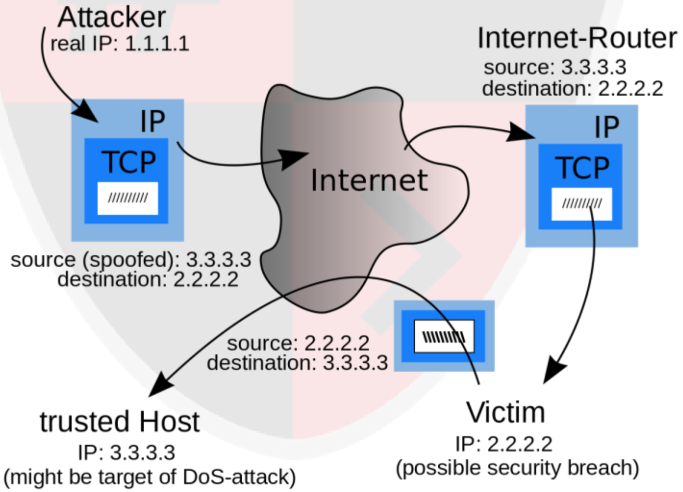
- SSH tunnelling is a somewhat like VPN. In VPN, you connect to a VPN server and all your traffic is encrypted and gets routed through that server.
- The premise is same but instead of a VPN server you have your home PC or router, acting as a server, for traffic routing and it takes few more steps to setup.
- The client side computer will connect to an SSH server through port 22. Most firewalls allow communication over port 22, as it is the port used by HTTPS and. Also, SSH also uses the same port so most firewalls allow it.
- **OpenSSH**: Attackers use OpenSSH to encrypt and tunnel all the traffic from a local machine to a remote machine to avoid detection by perimeter security controls.

2. IP Spoofing

Module 16

Evading Firewalls

- One way an attacker can attempt to evade a firewall is to appear as something else such as a trusted host. Using spoofing to modify IP address information, the attacker can make the source of the attack appear as if the traffic is coming from a host trusted by the firewall.



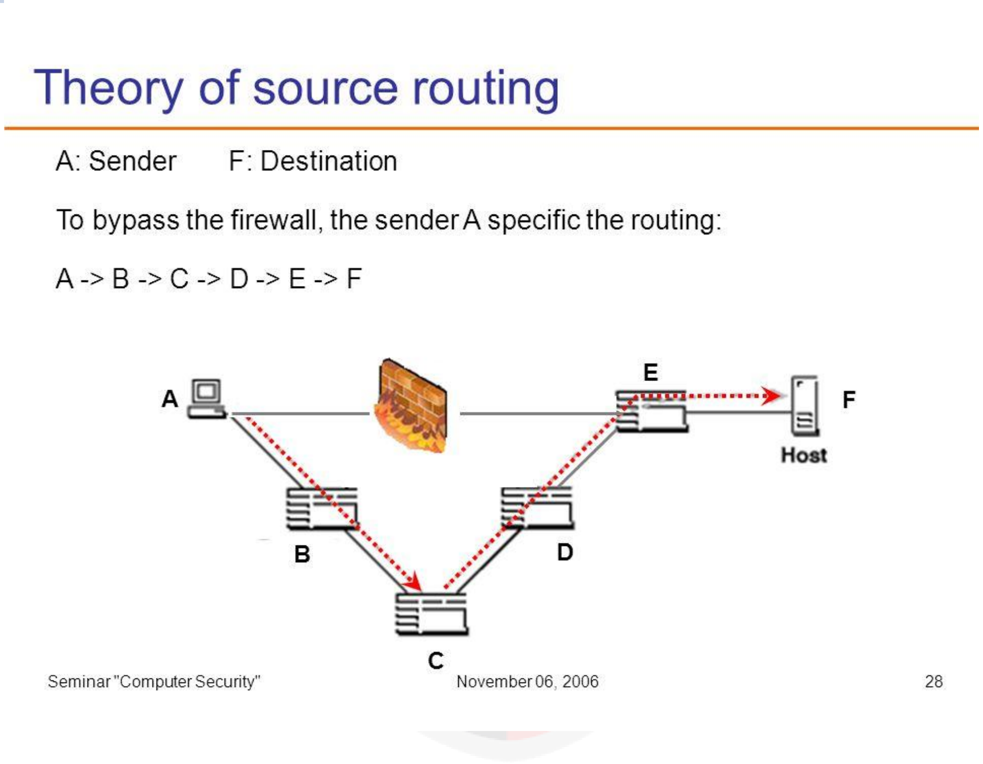
3. Source Routing

Module 16

Evading Firewalls

- When using source routing, the attacker designates the route a packet should take through the network in such a way that the designated route should bypass the firewall entirely, evading any restrictions the firewall has in place.
- Through the use of source routing, it is entirely possible for the attacker to specify the route he wishes the packet to use, instead of leaving it up to the routing protocol the organization has in place.
- This technique may also enable an attacker to reach a target host that is normally unreachable from the location of the attacker. This may include private RFC 1918 IP addresses that should not be present on the Internet.
- When combined with IP address spoofing, the attacker may have the ability to use a spoofed source address and still receive a response. Source routing also known as path addressing.

Evading Firewalls



Theory of source routing

A: Sender F: Destination

To bypass the firewall, the sender A specific the routing:

A -> B -> C -> D -> E -> F

Seminar "Computer Security" November 06, 2006 28

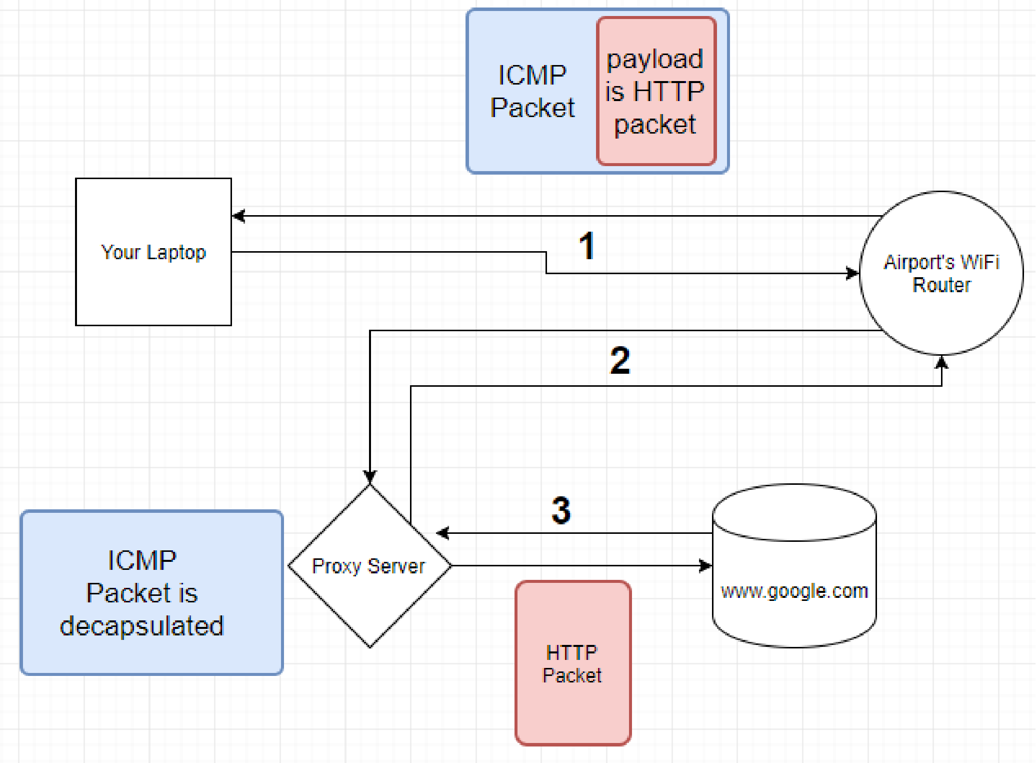
4. ICMP Tunneling

Module 16

Evading Firewalls

- ICMP tunneling works by injecting arbitrary data into an echo packet sent to a remote computer. The remote computer replies in the same manner, injecting an answer into another ICMP packet and sending it back. The client performs all communication using ICMP echo request packets, while the proxy uses echo reply packets.
- These packets are not necessarily forwarded to the client, as the client could be behind a translated address (NAT). This bidirectional data flow can be abstracted with an ordinary serial line.
- ICMP tunneling is possible because RFC 792, which defines the structure of ICMP packets, allows for an arbitrary data length for any type 0 (echo reply) or 8 (echo message) ICMP packets.

Evading Firewalls



Evading Firewalls

Mitigations:

DPI — Deep Packet Inspection

- Whereas the conventional packet inspections read the metadata of the packet (mainly headers), Deep packet inspection reads the contents of a packet that is going through it in real time.
- Most DPI tools rely on a signatures database — if there is no signature relevant for ICMP messages, it won’t detect the ICMP Tunnel.
- Even if there is a relevant signature at the database, the operator should first configure it to be in an active mode.

5. HTTP Tunneling

Module 16

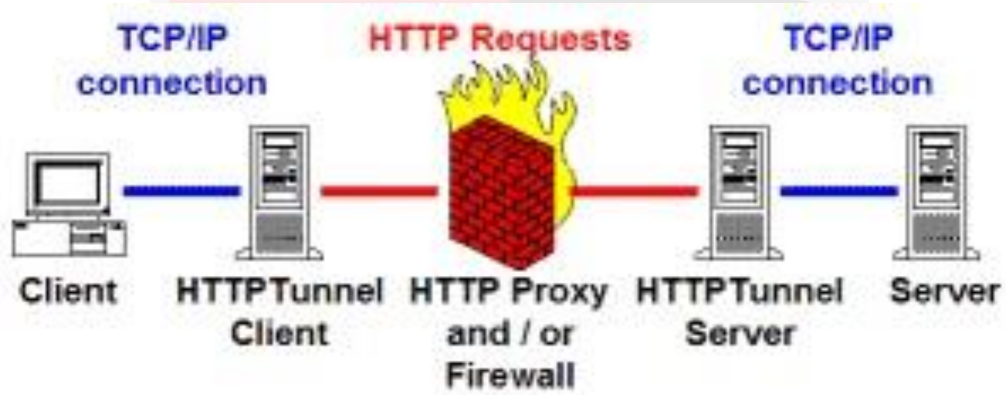
Evading Firewalls

- HTTP tunneling is used to create a network link between two computers including restrictions like firewalls, NATs and ACLs, etc.. The tunnel is created by an intermediary called a proxy server which is usually located in a DMZ.
- Tunneling can also allow communication using a protocol that normally wouldn’t be supported on the restricted network.
- HTTP tunneling performs protocol encapsulation, by enclosing data packets of one protocol (SOAP, JRMP, etc.) within HTTP Packets.
- The HTTP packets are then sent across the firewall as normal internet traffic.

Evading Firewalls

- The most common form of HTTP tunneling is the standardized HTTP CONNECT method.
- In this mechanism, the client asks an HTTP proxy server to forward the TCP connection to the desired destination. The server then proceeds to make the connection on behalf of the client. Once the connection has been established by the server, the proxy server continues to proxy the TCP stream to and from the client. Only the initial connection request is HTTP - after that, the server simply proxies the established TCP connection.
- This mechanism is how a client behind an HTTP proxy can access websites using SSL or TLS (i.e. HTTPS).

Evading Firewalls



Evading Firewalls

- The client connects to the proxy server and requests tunneling by specifying the port and the host computer it would like to connect to. The port is used to indicate the protocol being requested.[3]
  - **CONNECT streamline.t-mobile.com:22 HTTP/1.1**
  - **Proxy-Authorization: Basic encoded-credentials**
- If the connection was allowed and the proxy has connected to the specified host then the proxy will return a 2XX success response.[3]
  - **HTTP/1.1 200 OK**
- The client is now being proxied to the remote host. The client can communicate using any protocol accepted by the remote host. In the example below, the client is starting SSH communications, as hinted to, by the port number, in the initial CONNECT request.
  - SSH-2.0-OpenSSH_4.3\r\n
  - ...ggg

HACKING

Is an art, practised through a creative mind.