- Filename: eccouncil-ceh31250-v10-12-1-5-evading-ids-firewalls-and-honeypots
- Show Name: CEHv10 (312-50)
- Topic Name: Evading IDS, Firewalls & Honeypots
- Episode Name: Evading IDS, Firewalls & Honeypots Pt.5
- Description: In this episode, Adam and Wes discuss how to evade IDS, Firewalls & Honeypots.

==========

CEH v10 - Module 12 - Evading IDS, Firewalls & Honeypots

What is an intrusion? - what happens when an attacker bypasses or thwarts controls to gain access to resources.

Intrusion Detection System (IDS) - PASSIVE monitoring of activity looking for anomalies and alerting/notifying when they are found

Intrusion Prevention System (IPS) - ACTIVE monitoring of activity looking for anomalies and alerting/notifying AND taking action when they are found

Deployment Types: (HIDS & NIDS)

```
a. Host based - monitors activity on a single device/host by being installed
```

locally

```
b. Network based - monitors activity across a network using remote sensors that
```

report back to a central system. Often paired with a Security Information & Event Management (SIEM) system for analysis. Often Reverse ARP or Reverse DNS lookups are used to discover the source of attacks.

Knowledge & Behavior-Based Detection:

```
1. Knowledge Based (signature based | pattern matching) - Most common form of
```

detection. Uses a database of profiles, or signatures to assess all traffic against.

```
2. Behavior-Based (statistical | anomaly | heuristic) - Starts by creating a
```

baseline of behavior for the monitored system/network and then compares all traffic against that looking for deviations. Can be labeled an Artificially Intelligent (AI) or Expert system

Types of IDS alerts

True Positive (attack - alert)

False Positive (no attack - alert)

False Negative (attack - no alert)

True Negative (no attack - no alert)

What is a Firewall?

Firewalls are often seen as NAC devices. Use of rule sets to filter traffic can implement security policy. Several types of firewalls:

```
a. Stateful (Dynamic Packet Filtering - layer 3 + 4) vs. Stateless (Static
```

Packet Filtering - layer 3)

```
b. Deep Packet Inspection - layer 7

c. Proxy Firewall - mediates communications between untrusted and trusted
```

end-points (servers/hosts/clients). From an internal perspective, a proxy may forward traffic from known, internal client machines to untrusted hosts on the Internet, creating the illusion for the untrusted host that the traffic

originated from the proxy firewall, thus hiding the trusted internal client from potential attackers. To the user, it appears that he or she is communicating directly with the untrusted server.

Proxy Types:

```
Circuit-level proxy - creates a conduit through which a trusted host can
```

communicate with an untrusted one. This type of proxy does not inspect the data field that it forwards, which adds very little overhead to the communication between the user and untrusted server. The lack of application awareness also allows circuit-level proxies to forward any traffic to any TCP and UDP port. The disadvantage is that the data field will not be analyzed for malicious content.

```
Application-level proxy - relays the traffic from a trusted end-point running
```

a specific application to an untrusted end-point. They analyze the data field that they forward for various sorts of common attacks such as buffer overflows. Application-level proxies add processing overhead.

Multihomed firewall (dual-homed) - two or more network interfaces

Bastion host - endpoint that is exposed to the internet but has been hardened to withstand attacks

Screened Host - endpoint that is protected by a firewall

Demilitarized Zone (DMZ)

VPN

NAT

What is a Honeypot?

Honeypots/honeynets - Honeypot systems are decoy servers or systems setup to gather information regarding an attacker or intruder. Honeypots can be setup inside, outside or in the DMZ of a firewall design or even in all of the locations. In a sense, they are variants of standard Intrusion Detection Systems (IDS) but with more of a focus on information gathering and deception.

Two or more honeypots on a network form a honeynet. Honeynets and honeypots are usually implemented as parts of larger network intrusion detection systems.

A honeyfarm is a centralized collection of honeypots and analysis tools.

Types:

1. Low-interaction
2. Medium-interaction
3. High-interaction
4. Production
5. Research

SNORT... The Open Source IDS

SNORT has a rules engine that allows for customization of monitoring and detection capabilities.

Three available rule actions:

```
1. alert
2. pass
3. log
```

Three available IP protocols:

```
1. TCP
2. UDP
3. ICMP
```

What does a SNORT Rule look like?

alert icmp any any -> $HOME_NET any (msg:”ICMP test”; sid:1000001; rev:1; classtype:icmp-event;)

Shall we take a look at the syntax of this rule?

Rule Header = alert icmp any any -> $HOME_NET any

alert – Rule action. Snort will generate an alert when the set condition is met.

any – Source IP. Snort will look at all sources.

any – Source port. Snort will look at all ports.

-> – Direction. From source to destination.

$HOME_NET – Destination IP. We are using the HOME_NET value from the snort.conf file.

any – Destination port. Snort will look at all ports on the protected network.

Rule Options = (msg:”ICMP test”; sid:1000001; rev:1; classtype:icmp-event;)

msg:”ICMP test” – Snort will include this message with the alert.

sid:1000001 – Snort rule ID. Remember all numbers < 1,000,000 are reserved, this is why we are starting with 1000001 (you may use any number, as long as it’s greater than 1,000,000).

rev:1 – Revision number. This option allows for easier rule maintenance.

classtype:icmp-event – Categorizes the rule as an “icmp-event”, one of the predefined Snort categories. This option helps with rule organization.

Now you try one for yourself...

alert tcp 192.168.x.x any -> $HOME_NET 21 (msg:”FTP connection attempt”; sid:1000002; rev:1;)

How about...

alert tcp $HOME_NET 21 -> any any (msg:”FTP failed login”; content:”Login or password incorrect”; sid:1000003; rev:1;)

Anything unusual here??? Did you notice that we are setting the HOME_NET value as our source IP?

Why would we do this? Maybe because we will be looking for the outgoing FTP server responses?

How do I evade an IDS?

1. Insertion Attack - attacker forces the IDS to process invalid packets.

The attacker is able to exploit the IDS because it is not as "strict" in it's evaluation of packets as the endpoint system, allowing the attacker to exploit the IDS and force it to process the fake data.

2. Evasion - an endpoint accepts a packet that the IDS would normally reject. Typically executed via fragmentation of the attack packets to allow them to be moved through the IDS.

3. DoS Attack - overwhelming of one or more elements of the IDS ecosystem, forcing a failure.

4. Obfuscation - encoding the attack packets in such a way that the target is able to decode them, but the IDS is not.

    a. unicode
    b. polymorphic code
    c. encryption

d. path manipulation to cause signature mismatch

5. False Positive Generation Events - crafting of malicious packets designed to set off alarms with hope of distracting/overwhelming IDS and operators.
6. Session Splicing - Just another type of fragmentation attack.
7. Unicode Evasion - Use it and it messes things up !!
8. Fragmentation Attack - Used to push attack packets through the IDS without them being flagged as bad, allowing them to be received by the target and reassembled to form attack. Takes advantage of the differential in fragmentation timeout values between IDS and target.
9. Overlapping Fragments - generation of a series of tiny fragments with overlapping TCP sequence numbers. The target has to know how to reassemble the fragments, and depending on the O/S this behavior varies between use of the original fragments with a given offset (Earlier versions of Windows) and subsequent fragments with a given offset (CISCO)
10. Time-To-Live (TTL) Attack - requires the attacker to have inside knowledge of the target network to allow for the adjustment of the TTL values to control who gets what packets when
11. Invalid RST Packets - manipulation of the RST flag to trick IDS into ignoring the communication session with the target.
12. Urgency Flag - manipulation of the URG flag to cause the target and the IDS to have different sets of packets, because the IDS processes ALL packets irrespective of the URG flag, whereas the target will only process URG traffic.
13. Polymorphic Shellcode - blow up the pattern matching by constantly changing.
14. ASCII Shellcode - use ASCII characters to bypass pattern matching.
15. Application-Level Attacks - taking advantage of the compression used to transfer large files and hide attacks in compressed data, as it cannot be examined by the IDS.
16. Desynchronization - manipulating the TCP SYN to fool IDS into not paying attention to the sequence numbers of the illegitimate attack traffic, but rather, give it a false set of sequences to follow.
17. Encryption - using encryption to hide attack
18. Flooding - overwhelming the IDS

How do I evade a Firewall?

1. Firewalking - using TTL values to determine gateway ACL filters and allow for mapping of internal networks by analyzing IP packet responses.
2. Banner Grabbing - looking for FTP, telnet and web server banners
3. IP Address Spoofing - hijacking technique allowing attacker to masquerade as a trusted host.
4. Source Routing - allows the sender of a packet to partially or fully specify the route to be used.
5. Tiny Fragments - as described above, successful with firewalls when they ONLY CHECK for the TCP header info, allowing the fragmentation of the information across multiple packets to hide the true intention of the attack.
6. Use IP in place of a URL - may work depending on nature of filtering in place
7. Use Proxy Servers/Anonymizers - may work depending on nature of filtering in place
8. ICMP Tunneling - allows for the tunneling of a backdoor shell via the ICMP echo packets because the RFC (792) does not clearly define what kind of data goes in the data portion of the frame, allowing for attack traffic to be seen as acceptable when inserted. If firewalls do not examine the payload section of

the dataframe, they would let the data through, allowing the attack.

9. ACK Tunneling - use of the ACK flag to trick firewall into allowing packets, as many firewalls do not check ACK packets.
10. HTTP Tunneling - use of HTTP traffic to "hide" attacks.
11. SSH Tunneling - use of SSH to encrypt and send attack traffic.
12. MitM Attacks - use of DNS and routing manipulation to bypass firewalls.
13. XSS Attacks - allows for the exploitation of vulnerabilities around the processing of input parameters from the end user and the server responses in a web application. The attacker injects malicious HTML code into the website to force the bypassing of the firewall once executed.

How do I detect a Honeypot?

Probe services running on them; ports that show a service is available, but deny a three-way handshake may indicate that the system is a honeypot

Layer 7 - examine latency of responses from server

Layer 4 - examine the TCP Window size, looking for continuous acknowledgement of incoming packets even when the window size is set to 0

Layer 2 - If you are on the same network as the honeypot, look for MAC addresses in packets that indicate the presence of a "Black Hole"
0:0:f:ff:ff:ff

If honeypot is virtualized, look for the vendor assigned MAC address ranges as published by IEEE

If Honeypot is the Honeyd type, use time based TCP finger printing methods to detect

Detecting User-Mode Linux (UML) honeypot, analyze proc/mounts, proc/interuppts and proc/cmdline which would have UML specific settings and information

Detecting Sebek-based honeypots, Sebek will log eveything that is accessed via read() BEFORE sending to the network, causing congestion that can be an indicator

Detecting snort_inline honeypots, analyze the outgoing packets by capturing the snort_inline modified packets through another machine and analyzing them for evidence of the modifications

Detecting a Fake AP, fake AP's only send beacon frames but do not generate any fake traffic, allowing for network monitoring to identify them easily