TASK 8 – Intrusion Detection

Room : https://tryhackme.com/room/idsevasion

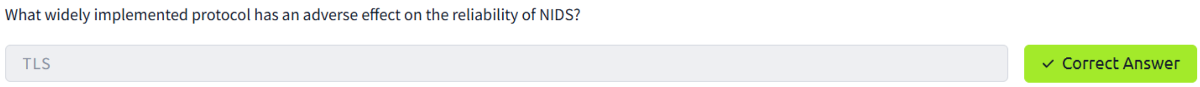
Task 1 - Introduction

I set up the TryHackMe machine and registered a user on the web interface so any alerts would match my session.
I noted the target IP and the open web ports (for example, port 3000) and checked the alerts page at http://MACHINE_IP:8000/alerts.
This made sure the environment was ready for the IDS experiments.

Task 2 – Intrusion Detection Basics

Intrusion Detection Systems (IDS) watch network or host activity and flag anything that looks suspicious instead of blocking it outright.
There are two main ways they work: signature-based (rules that match known bad stuff) and anomaly-based (learns normal behaviour and alerts on odd activity).
This demo uses Suricata (network IDS) and Wazuh (host IDS) to show how different signatures catch different types of attacks and how alerts get sent to tools like Graylog or the ELK Stack.

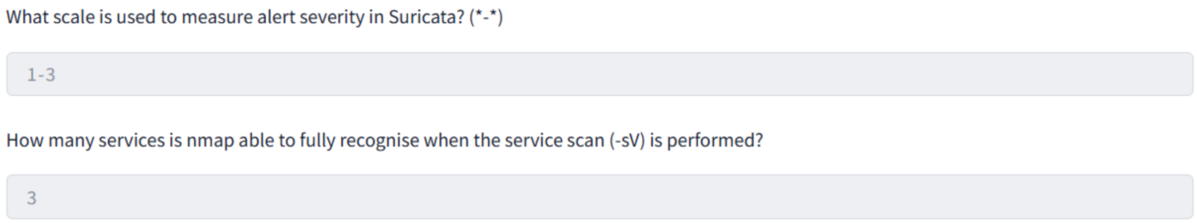
Task 3- Network Based IDS


What widely implemented protocol has an adverse effect on the reliability of NIDS?

TLS

Task 4 - Reconnaissance and Evasion Basics

I tried simple evasion by changing HTTP headers and running a SYN stealth scan (nmap -sV 10.201.94.189).
Changing the User-Agent cut down some signature alerts, while the SYN scan made less application-layer noise but still triggered network-level indicators.

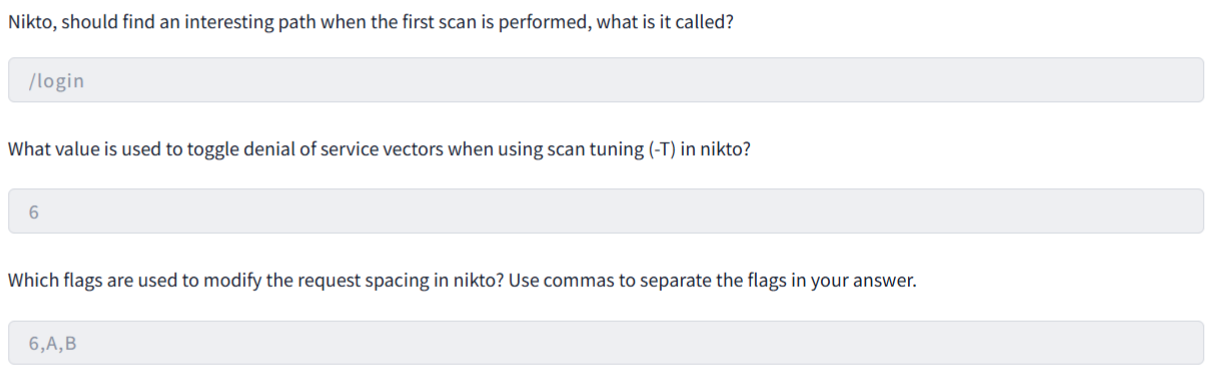

What scale is used to measure alert severity in Suricata? (*-*)

1-3

How many services is nmap able to fully recognise when the service scan (-sV) is performed?

3

Task 5 - Further Reconnaissance Evasion



Nikto, should find an interesting path when the first scan is performed, what is it called?

/login

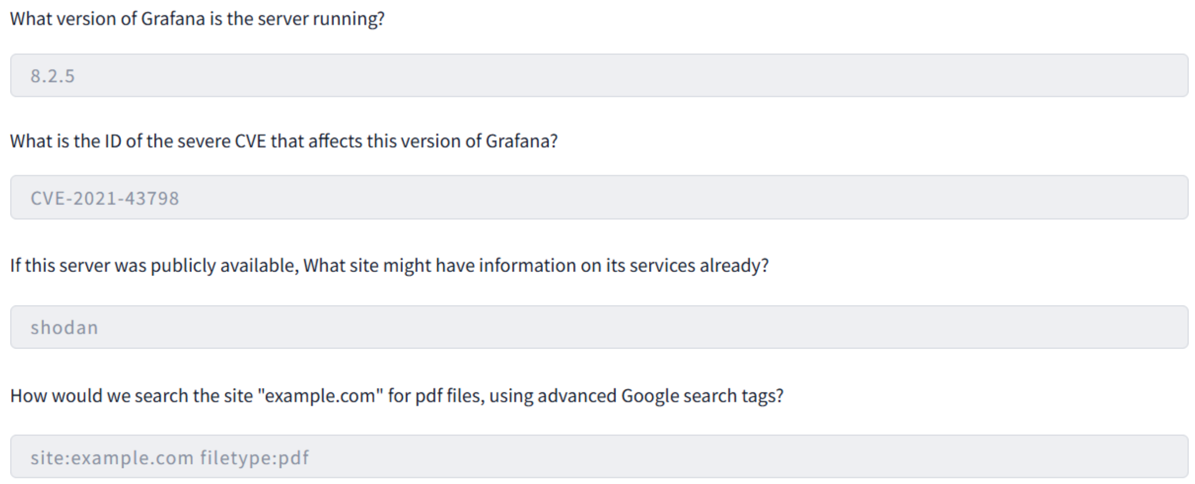
What value is used to toggle denial of service vectors when using scan tuning (-T) in nikto?

6

Which flags are used to modify the request spacing in nikto? Use commas to separate the flags in your answer.

6,A,B

Task 6 – Open Source Intelligence



What version of Grafana is the server running?

8.2.5

What is the ID of the severe CVE that affects this version of Grafana?

CVE-2021-43798

If this server was publicly available, What site might have information on its services already?

shodan

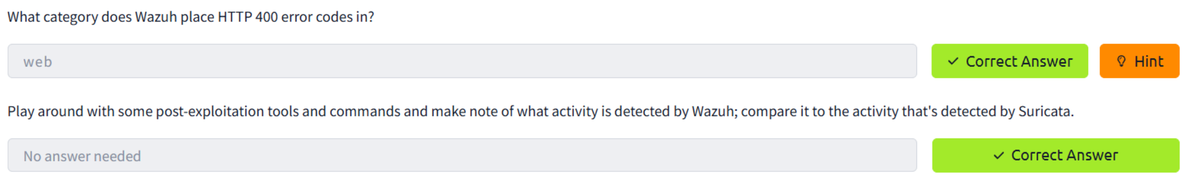
How would we search the site "example.com" for pdf files, using advanced Google search tags?

site:example.com filetype:pdf

Task 7 – Rulesets

I downloaded the exploit script and ran it against the service on port **3000** to read files as the service user.
I then checked the IDS alerts at **MACHINE_IP:8000/alerts** and inspected Suricata's rule hits.
Suricata showed alerts that matched Emerging Threats signatures for this exploit, so the attack was logged — but I noted rule limitations and possible gaps in coverage.

Task 8 – Host Based IDS



What category does Wazuh place HTTP 400 error codes in?

web

Play around with some post-exploitation tools and commands and make note of what activity is detected by Wazuh; compare it to the activity that's detected by Suricata.

No answer needed

Task 9 – Privilage Escalation Recon

I checked my current permissions with sudo -l, groups, and cat /etc/group to see what I could do and who else had higher access.

I ran **linPEAS** to hunt for SUID binaries, world-writable files, and leaked credentials — it found several promising escalation vectors.
Suricata (the NIDS) didn't see the local commands, but the HIDS/file-integrity alerts fired for the linPEAS activity and any added files.
I also noticed that downloading tools with wget would be visible to the NIDS unless the traffic was TLS-protected or copied by hand.

Task 10 – Performing Privilage Escalation

I used the grafana-admin account to run a Docker container that mounted the host filesystem (docker run -it --entrypoint=/bin/bash -v /:/mnt/ ...) and got a shell into the host.
From inside the container I edited /mnt/etc/sudoers to add grafana-admin ALL=(ALL) NOPASSWD: ALL, gaining root privileges.
Wazuh (HIDS) raised file-integrity alerts for the modified files, while Suricata (NIDS) didn't see the local changes.

Task 11 – Establishing Persistence

I tried persistence methods like adding a cron job and tweaking an init script, then watched the IDS alerts.
Wazuh flagged the file changes and the new process, showing host-level detection of my persistence attempts.

Conclusion

This CTF showed that IDS tools can catch a lot — network sensors spotted scans and exploits, and host sensors noticed file changes and persistence.
But each has gaps (network tools miss local actions, host tools can be noisy), so the best defense is layers: keep systems patched, harden Docker/IoT, use DDoS/CDN protections, and share alerts quickly with your team.