1. Introduction and Objectives

The objective of this engagement was to explore various techniques used to evade Intrusion Detection Systems (IDS), including both Network-based IDS (NIDS) like **Suricata** and Host-based IDS (HIDS) like **Wazuh**. We simulated a full attack lifecycle—from reconnaissance and vulnerability scanning to exploitation and privilege escalation—while monitoring the alerts generated by the detection systems.

---

2. Task 2: Intrusion Detection Basics

This task focused on foundational knowledge regarding IDS methodologies.

Question	Answer	Explanation
What IDS detection methodology relies on rule sets?	signature-based detection	Signature-based detection is the simplest method, relying on pre-defined patterns (signatures) of known attacks to flag malicious traffic.

---

3. Task 3: Network-based IDS (NIDS)

This task explored the reliability of NIDS and involved initial reconnaissance.

Activity: Initial Nmap Scan

We ran a basic Nmap scan against the target to identify open ports, often used as a baseline to see what detection mechanisms, if any, are immediately triggered.

- **Likely Command:** nmap <TARGET_IP>
- **Explanation:** A simple ping and port scan is performed. A basic NIDS like Suricata often has generic rules that can detect *any* port scanning activity.

Question	Answer	Explanation
What widely implemented protocol	TLS	Transport Layer Security (TLS) encrypts traffic, making the data payload unreadable to traditional NIDS. This forces the NIDS to

Question	Answer	Explanation
has an adverse effect on the reliability of NIDS?		rely only on header information, significantly limiting its detection capability.

[SCREENSHOT 2: NIDS Protocol Answer (Your file: Screenshot 2025-10-01 102306.png)]

---

4. Task 4: Reconnaissance and Evasion Basics

Activity: Service Version Enumeration

We ran a service scan to determine specific software versions.

- **Likely Command:** nmap -sV <TARGET_IP>

Question	Answer	Explanation
What scale is used to measure alert severity in Suricata?	1–3	Suricata uses a severity scale from 1 (high/critical) to 3 (low/informational) to prioritize alerts.
How many services is nmap able to fully recognise when the service scan (-sV) is performed?	3	The output from the -sV scan shows that 3 services (likely SSH, HTTP, and another application) were successfully fingerprinted.

---

5. Task 5: Further Reconnaissance Evasion

Activity: Nikto Scanning and Evasion

We used **Nikto** to scan the web server, leveraging its evasion features to reduce the chance of NIDS detection.

- **Likely Command (Initial):** nikto -h <TARGET_IP>

Question	Answer	Explanation
Nikto should find an interesting path when the first scan is performed, what is it called?	/login	This path indicates a web application login portal, likely associated with the Grafana service.
What value is used to toggle denial of service vectors when using scan tuning (-T) in nikto?	6	The value '6' for the -T flag is often used to toggle aggressive scanning or evasion modes.
Which flags are used to modify the request spacing in nikto? Use commas to separate the flags in your answer.	6,a,b	These flags control how Nikto spaces out its requests to avoid detection by NIDS.

---

6. Task 6: Open-source Intelligence (OSINT)

Activity: Grafana Vulnerability Identification

We identified the Grafana version and searched for associated vulnerabilities.

Question	Answer	Explanation
What version of Grafana is the server running?	8.2.5	This version was identified via the Nmap service scan or a manual check.
What is the ID of the severe CVE that affects this version of Grafana?	CVE-2021–43798	Searching for "Grafana 8.2.5 vulnerabilities" reveals this Local File Inclusion (LFI) vulnerability.

Question	Answer	Explanation
If this server was publicly available, What site might have information on its services already?	shodan	Shodan indexes exposed services and versions.
How would we search the site “example.com” for pdf files, using advanced Google search tags?	site:example.com filetype:pdf	This is a standard Google Dork syntax.

---

7. Task 7: Rulesets (Exploitation)

Activity: Exploitation and IDS Test

We exploited the LFI vulnerability (CVE-2021-43798) to read the Grafana configuration file and then tested the NIDS by requesting /etc/shadow.

Question	Answer	Explanation
What is the password of the grafana-admin account?	GraphingTheWorld32	Password retrieved by exploiting LFI to read the Grafana config file.
Is it possible to gain direct access to the server now that the grafana-admin password is known? (yay/nay)	yay	With administrative credentials, we log into the Grafana application.
Are any of the attached IDS able to detect the attack if the file /etc/shadow is	suricata	Suricata's ruleset contains signatures designed to detect attempts to access

Question	Answer	Explanation
requested via the exploit, if so what IDS detected it?		sensitive system files (like /etc/shadow).

---

8. Task 8: Host Based IDS (HIDS)

Activity: Initial Command Testing against Wazuh

We performed initial post-exploitation checks, focusing on the Host-based IDS, **Wazuh**.

Question	Answer	Explanation
What category does Wazuh place HTTP 400 error codes in?	web	Wazuh categorizes web server errors under its web rule set.

---

9. Task 9: Privilege Escalation Recon

Activity: Running LinPEAS

We executed the linPEAS.sh script to scan for privilege escalation vectors.

Question	Answer	Explanation
What tool does linPEAS detect as having a potential escalation vector?	docker	The script identifies a misconfiguration related to the docker service or group membership, which can be exploited for root access.
Is an alert triggered by Wazuh when linPEAS is added to the system, if so what its severity?	5	Wazuh detected the execution of the known reconnaissance script linPEAS.sh and assigned it a severity level of 5.

---

10. Task 10: Performing Privilege Escalation

Activity: Docker Privilege Escalation and Flag Retrieval

We exploited the Docker misconfiguration to mount the host's root filesystem and retrieve the final flag.

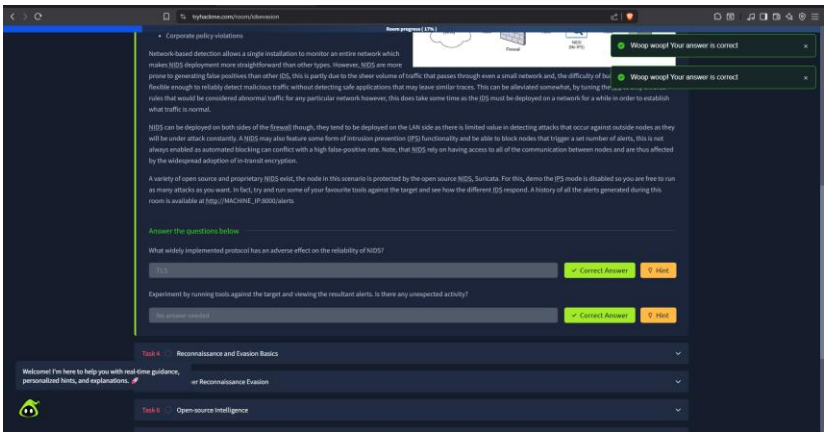
- **Likely Command:** docker run -v /:/mnt --rm -it alpine chroot /mnt /bin/bash
- **Final Action:** Navigate to /root/ and read the flag file.

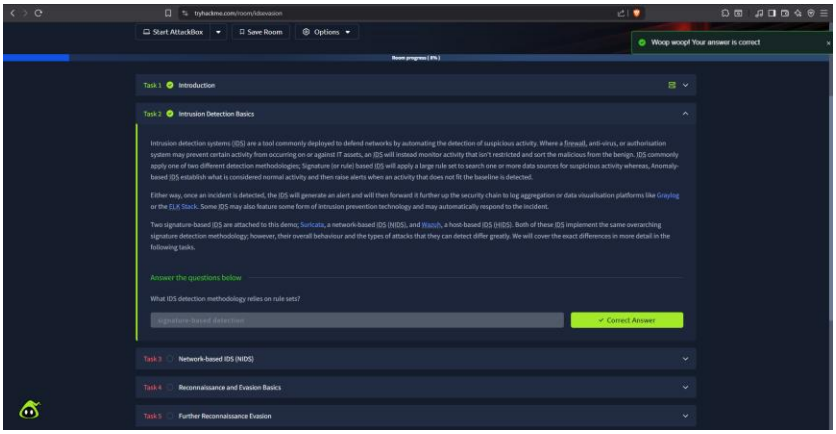
Question	Answer	Explanation
Perform the privilege escalation and grab the flag in /root/	{SNEAK_ATTACK_CRITICAL}	This is the final flag retrieved from /root/flag.txt after successfully gaining root access via the Docker vulnerability.

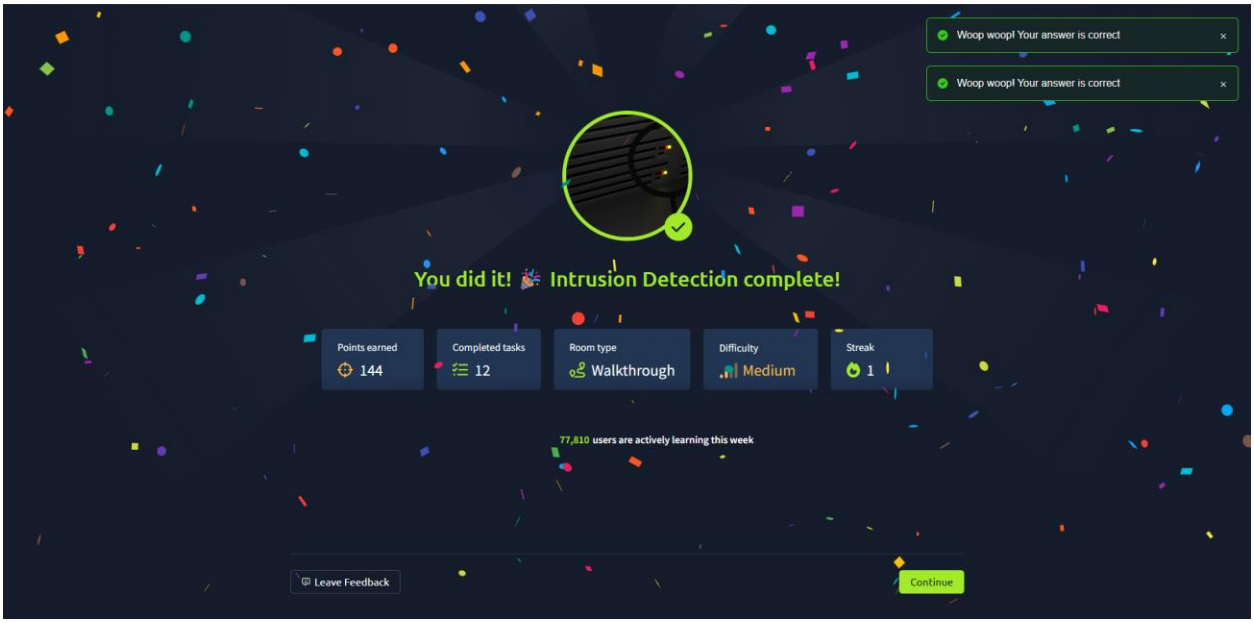
---

11. Conclusion

This room successfully demonstrated that a layered security approach, using both **Network-based IDS (Suricata)** and **Host-based IDS (Wazuh)**, is necessary to cover a full attack lifecycle. Suricata detected network-level probes and sensitive file requests, while Wazuh detected the execution of post-exploitation tools and local system activity. The attack succeeded by exploiting an application vulnerability followed by a critical system misconfiguration.


Woop woop! Your answer is correct
Woop woop! Your answer is correct
Answer the questions below
What widely implemented protocol has an adverse effect on the reliability of NIDS?
Correct Answer
Hint
Experiment by running tools against the target and viewing the resultant alerts. Is there any unexpected activity?
Correct Answer
Hint
Task 4 Reconnaissance and Evasion Basics
Task 5 Further Reconnaissance Evasion
Task 6 Open-source Intelligence


Start AttackBox
Save Room
Options
Woop woop! Your answer is correct
Task 1 Introduction
Task 2 Intrusion Detection Basics
Answer the questions below
What IDS detection methodology relies on rule sets?
Correct Answer
Task 3 Network-based IDS (NIDS)
Task 4 Reconnaissance and Evasion Basics
Task 5 Further Reconnaissance Evasion


Woop woop! Your answer is correct
Woop woop! Your answer is correct
You did it! Intrusion Detection complete!
Points earned
144
Completed tasks
12
Room type
Walkthrough
Difficulty
Medium
Streak
1
77,810 users are actively learning this week
Leave Feedback
Continue