Mahmudur Rahman
Snort Lab Updated (11-05-2025)

Network Intrusion Detection System (Snort IDS Lab)

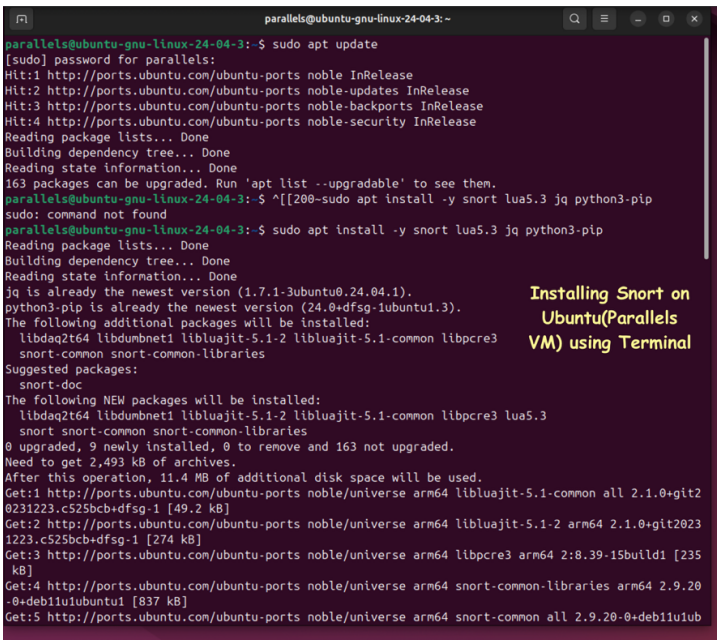
Ubuntu (Parallels VM) | Python | hping3 | Snort 2.9.20 | 2025

- Deployed and configured **Snort IDS** on Ubuntu within a virtualized Parallels environment to detect malicious traffic and perform network-level threat analysis.
- Wrote and tuned **custom local.rules** for detecting SSH brute force and port-scanning attempts with detection filters and unique SIDs.
- Used **hping3** to simulate attack traffic and validate detection logic on loopback and enp0s5 interfaces.
- Captured, analyzed, and visualized Snort alerts to confirm detection accuracy for both inbound and outbound scans.
- Created a self-contained test lab with **Python automation scripts** (generate_scan.py, generate_ssh_bruteforce.py) and alert-parsing utilities for reproducible IDS testing.
- Verified Snort's **rule engine initialization**, preprocessor configuration, and packet decoding pipeline, confirming successful packet capture and live alert generation.

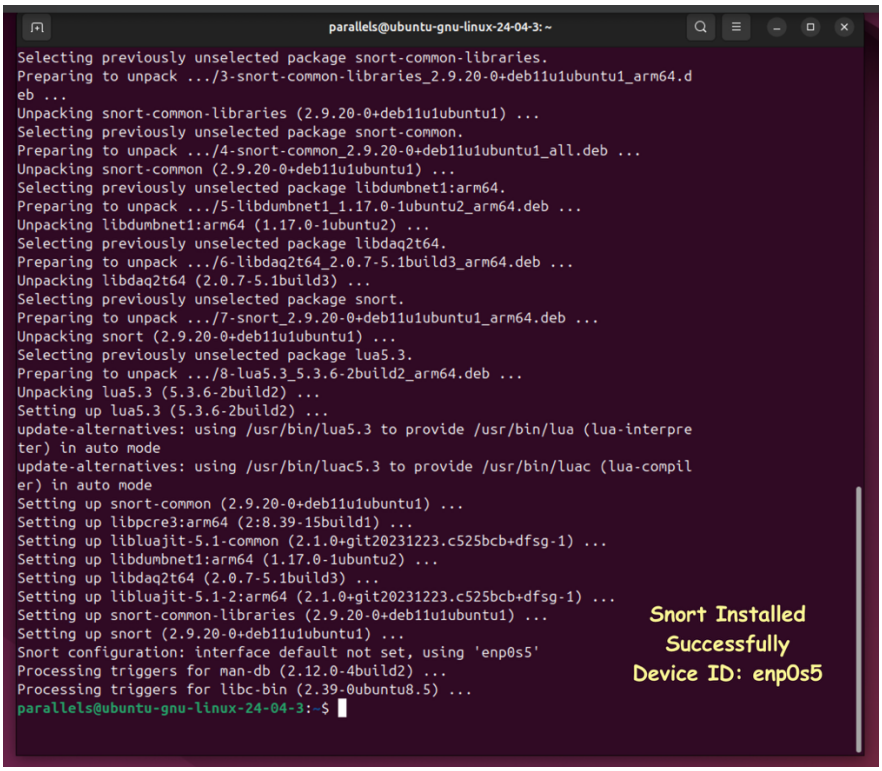
Key Skills: Intrusion Detection Systems (IDS), Snort, Network Security, Packet Analysis, Python Automation, hping3, Linux Administration, Cyber Defense Lab Setup

Screenshots Showcase

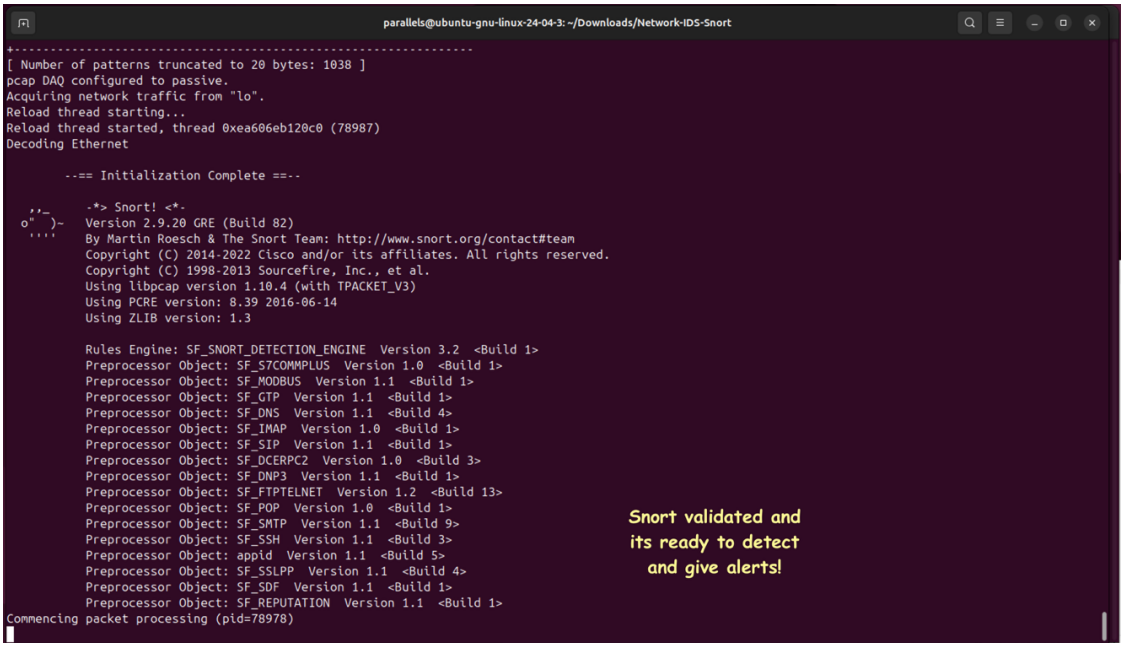
1. *Installing Snort on Ubuntu (Parallels VM) using Terminal*



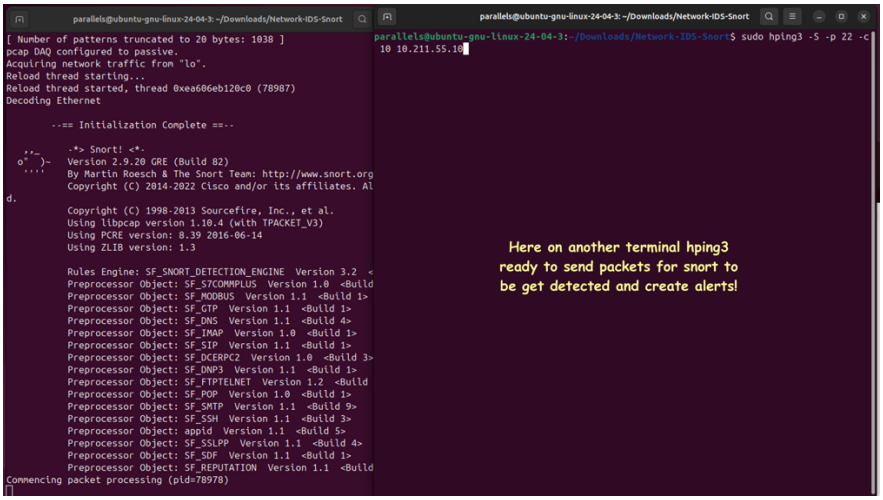
2. *Snort successfully installed and configured — interface enp0s5 identified*



3. *Snort validated configuration: initialized preprocessors and detection engine*



4. *Running Snort on loopback interface — ready to detect and generate alerts*



5. *Simulated SSH brute-force traffic with hping3 scripts; Snort generated real-time alerts in console view*