

Exploring Snort Capabilities

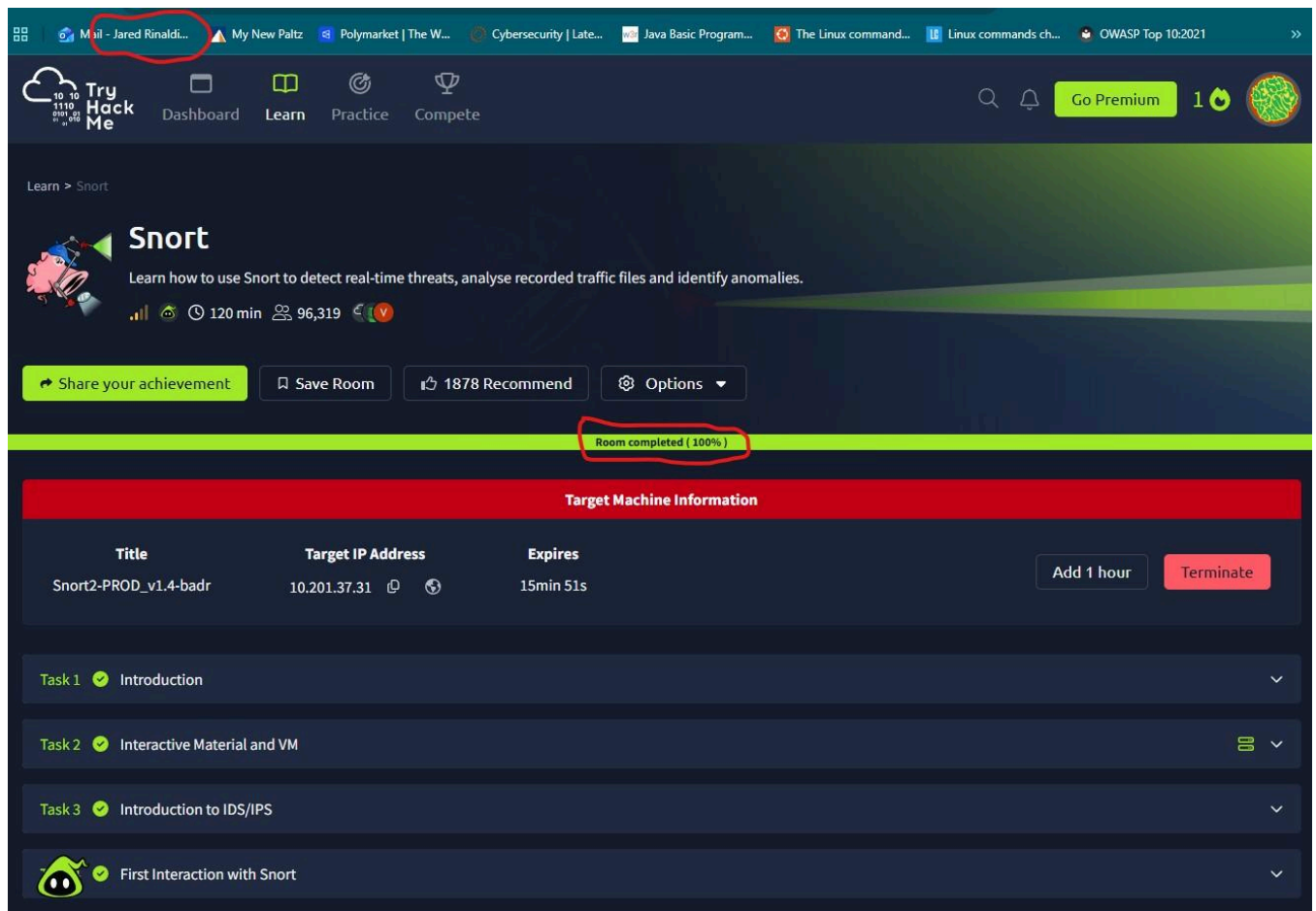
Researcher: Jared Rinaldi

GitHub: [GitHub.com/PyronicGreen](https://github.com/PyronicGreen)

November 3, 2025

Abstract: Snort is an open source network intrusion detection and prevention system (NIDS/NIPS) made by Cisco. It is capable of performing real-time traffic analysis and packet logging on IP networks. Additionally, it can perform protocol analysis, content searching and matching, as well as detect a variety of attacks and probes. It can be used as a straight packet sniffer like tcpdump, a packet logger (useful for network traffic debugging) or as a full-blown network intrusion prevention system. In this lab, I will install Snort on my Ubuntu Virtual Machine. Snort will then be used to detect my network traffic, which will be saved to a snort log file where it can be analyzed.

TryHackMe Snort Assignment:



The screenshot shows the TryHackMe interface for the 'Snort' assignment. The top navigation bar includes links to 'Dashboard', 'Learn', 'Practice', and 'Compete'. The assignment page features a 'Snort' title, a description, and a progress bar indicating 'Room completed (100%)'. Below this, there is a table for 'Target Machine Information' and a list of tasks.

Title	Target IP Address	Expires
Snort2-PROD_v1.4-badr	10.201.37.31	15min 51s

Tasks:

- Task 1: Introduction
- Task 2: Interactive Material and VM
- Task 3: Introduction to IDS/IPS
- First Interaction with Snort

Assignment 6

Snort Configuration

Step 1: Update the System

```
rinaldij@JaredRinaldi:/var/log/snort$ sudo apt update
[sudo] password for rinaldij:
Hit:1 http://us.archive.ubuntu.com/ubuntu focal InRelease
Hit:2 http://security.ubuntu.com/ubuntu focal-security InRelease
Reading package lists... Done
Building dependency tree
Reading state information... Done
All packages are up to date.
rinaldij@JaredRinaldi:/var/log/snort$ sudo apt upgrade -y
Reading package lists... Done
Building dependency tree
Reading state information... Done
Calculating upgrade... Done
0 upgraded, 0 newly installed, 0 to remove and 0 not upgraded.
rinaldij@JaredRinaldi:/var/log/snort$
```

Step 2: Confirm Snort Installation

```
rinaldij@JaredRinaldi: ~
rinaldij@JaredRinaldi:~$ snort -V

  ,,-_
o"  )~
  '""

-*> Snort! <*-
Version 2.9.7.0 GRE (Build 149)
By Martin Roesch & The Snort Team: http://www.snort.org/contact#team
Copyright (C) 2014 Cisco and/or its affiliates. All rights reserved.
Copyright (C) 1998-2013 Sourcefire, Inc., et al.
Using libpcap version 1.9.1 (with TPACKET_V3)
Using PCRE version: 8.39 2016-06-14
Using ZLIB version: 1.2.11

rinaldij@JaredRinaldi:~$
```

Step 5: Test Snort Configuration

```
Snort successfully validated the configuration!
Snort exiting
rinaldij@JaredRinaldi:~$
```

Steps 6 and 7: Download and View Snort Logs

I ran the command `sudo snort -c /etc/snort/snort.conf -i enp0s3`. I let the command run for around 15 seconds, and then stopped it with CTRL+c. A log file was generated called `snort.log`. I checked this file using the `tail` and `nano` commands,

but it did not seem to contain any log entries. I believe this is due to my system being isolated from the rest of the network due to it being a virtual machine, and therefore there is no inbound network traffic.

```
rinaldij@JaredRinaldi:/var/log/snort$ ls
snort.log
rinaldij@JaredRinaldi:/var/log/snort$ tail snort.log
rinaldij@JaredRinaldi:/var/log/snort$
```

Step 8: Running Snort as a Daemon

```
rinaldij@JaredRinaldi:/var/log/snort$ sudo snort -D -c /etc/snort/snort.conf -i
enp0s3
Spawning daemon child...
My daemon child 3697 lives...
Daemon parent exiting (0)
rinaldij@JaredRinaldi:/var/log/snort$
```

When I ran the command for this step, I did not see “Snort” running as one of the processes. As this looked to be the wrong outcome, I did a little research and discovered that VirtualBox’s default network configuration is set to NAT, which prevents the virtual host from getting any inbound traffic.

The fix was to shut down VirtualBox and switch the network type to “Bridged Connection.” After doing this and re-starting my Ubuntu machine, I went back and repeated step 6, believing that now Snort would have some network traffic to work with. This proved to be accurate. Right away, the output made it clear that Snort was analyzing packets. This meant my virtual network was no longer isolated.

```
rinaldij@JaredRinaldi: ~  
^C*** Caught Int-Signal  
=====
```

Run time for packet processing was 19.29073 seconds	
Snort processed 177 packets.	
Snort ran for 0 days 0 hours 0 minutes 19 seconds	
Pkts/sec:	9

```
=====
```

Memory usage summary:	
Total non-mmapped bytes (arena):	45948928
Bytes in mapped regions (hblkhd):	13574144
Total allocated space (uordblks):	40402256
Total free space (fordblks):	5546672
Topmost releasable block (keepcost):	19392

```
=====
```

Packet I/O Totals:	
Received:	184
Analyzed:	177 (96.196%)
Dropped:	0 (0.000%)
Filtered:	0 (0.000%)
Outstanding:	7 (3.804%)
Injected:	0

```
=====
```

Breakdown by protocol (includes rebuilt packets):	
Eth:	177 (100.000%)
VLAN:	0 (0.000%)
IP4:	31 (17.514%)
Frag:	0 (0.000%)
ICMP:	0 (0.000%)
UDP:	28 (15.819%)
TCP:	0 (0.000%)

The snort.log file now contained data entries (step 7).

Returning to Step 8, the command `sudo snort -D -c /etc/snort/snort.conf -i enp0s3` was again run.

```
rinaldij@JaredRinaldi: ~  
rinaldij@JaredRinaldi:~$ sudo snort -D -c /etc/snort/snort.conf -i enp0s3  
Spawning daemon child...  
My daemon child 2693 lives...  
Daemon parent exiting (0)  
rinaldij@JaredRinaldi:~$
```

This time, “snort” did show up as a process when the `top` command was run.


```
rinaldij@JaredRinaldi: ~  
top - 16:04:01 up 29 min, 1 user, load average: 0.02, 0.05, 0.07  
Tasks: 193 total, 1 running, 192 sleeping, 0 stopped, 0 zombie  
%Cpu(s): 0.7 us, 0.5 sy, 0.0 ni, 98.7 id, 0.0 wa, 0.0 hi, 0.2 si, 0.0 st  
MiB Mem : 1963.8 total, 244.9 free, 893.3 used, 825.6 buff/cache  
MiB Swap: 2048.0 total, 2048.0 free, 0.0 used. 910.4 avail Mem
```

PID	USER	PR	NI	VIRT	RES	SHR	S	%CPU	%MEM	TIME+	COMMAND
2157	rinaldij	20	0	4193476	341488	126468	S	0.7	17.0	0:27.00	gnome-s+
807	root	20	0	1874776	45148	32276	S	0.3	2.2	0:03.22	contain+
1955	rinaldij	20	0	262636	78008	46852	S	0.3	3.9	0:16.40	Xorg
2083	rinaldij	20	0	221172	2708	2256	S	0.3	0.1	0:03.68	VBoxCli+
2693	root	20	0	568116	148936	5176	S	0.3	7.4	0:00.02	snort
2699	rinaldij	20	0	11984	3860	3224	R	0.3	0.2	0:00.38	top
1	root	20	0	168028	11556	8176	S	0.0	0.6	0:01.46	systemd
2	root	20	0	0	0	0	S	0.0	0.0	0:00.00	kthreadd
3	root	0	-20	0	0	0	I	0.0	0.0	0:00.00	rcu_gp
4	root	0	-20	0	0	0	I	0.0	0.0	0:00.00	rcu_par+
5	root	0	-20	0	0	0	I	0.0	0.0	0:00.00	slub_fl+
6	root	0	-20	0	0	0	I	0.0	0.0	0:00.00	netns
8	root	0	-20	0	0	0	I	0.0	0.0	0:00.00	kworker+
10	root	0	-20	0	0	0	I	0.0	0.0	0:00.00	mm_perc+
11	root	20	0	0	0	0	S	0.0	0.0	0:00.00	rcu_tas+
12	root	20	0	0	0	0	S	0.0	0.0	0:00.00	rcu_tas+

To stop all snort processes from running, I ran the command `sudo pkill -TERM snort`. I used the `pgrep` command to confirm no Snort processes were running.

```
rinaldij@JaredRinaldi:~$ sudo pkill -TERM snort  
rinaldij@JaredRinaldi:~$ pgrep snort || echo "no snort processes running!"  
no snort processes running!  
rinaldij@JaredRinaldi:~$
```

Conclusion: In this lab, I successfully installed Snort and used it in IDS mode to monitor traffic and log alerts. Using the “-D” option in the snort command, I was able to configure Snort to run as a background process and monitor incoming and outgoing network traffic. Snort is a useful tool and I anticipate using it for future home lab purposes.