DESIGN OF A REAL-TIME NETWORK IDPS USING WELL-KNOWN OPEN-SOURCE TOOLS

Jhonatan Parada Torres[1], Dr. Merlinda Drini[1,2]

[1]Queensborough Community College (CUNY), [2]Department of Engineering Technology

CUNY Research Scholars Program

Abstract

Intrusion Detection System (IDS) can be defined as the act of detecting actions that attempt to compromise the confidentiality, integrity or availability of a resource. Intrusion Detection and prevention system (IDPS) follows the same process of gathering and identifying data and behavior, with the added ability to block the activity.The purpose of this research is to design a real-time network IDPS using open-source tools. The first part is to explore and evaluate open-source tools available on the web and choose the best ones with the final design implementation (SNORT, SURICATA). The second part of this project is to research, utilize, and integrate the selected tools into real-time IDPSs. An IDS will be created to include such features as detecting an attack, incorporating rules for monitoring intrusions, and including approaches that would minimize false alarms, and ensure that the performance overhead is acceptable.

Introduction

In order to compare open-source IDS tools in a UNIX-based system, a virtual network consisting of three machines was created using the QEMU/KVM Virtual Machine Manager[7]. The first machine (virtual) had the Ubuntu 24.04.3 LTS operating system (OS); The second one had the Linux Mint 22.1 OS (host); And the last one had the Lubuntu 24.04.3 OS (virtual).

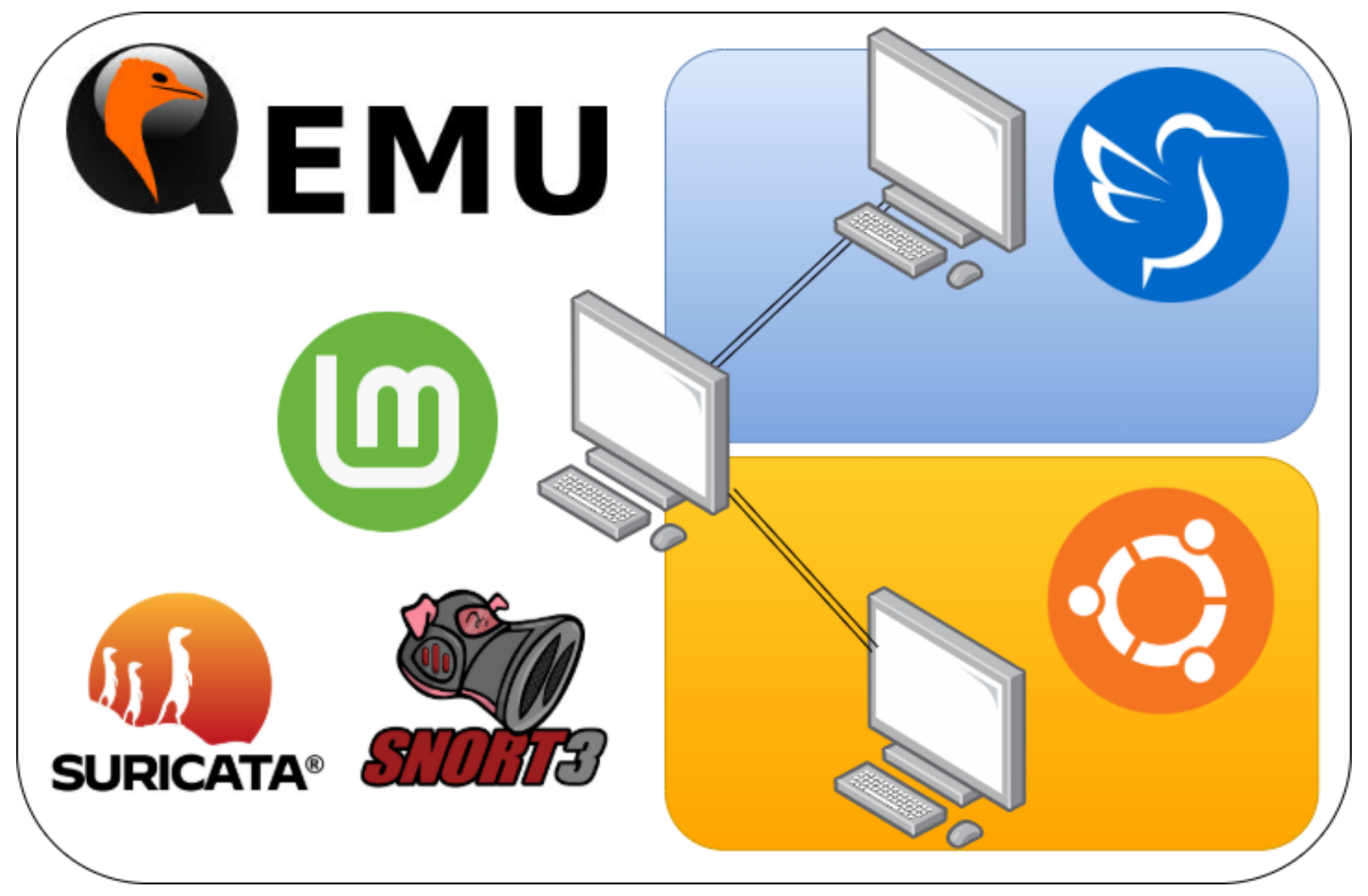
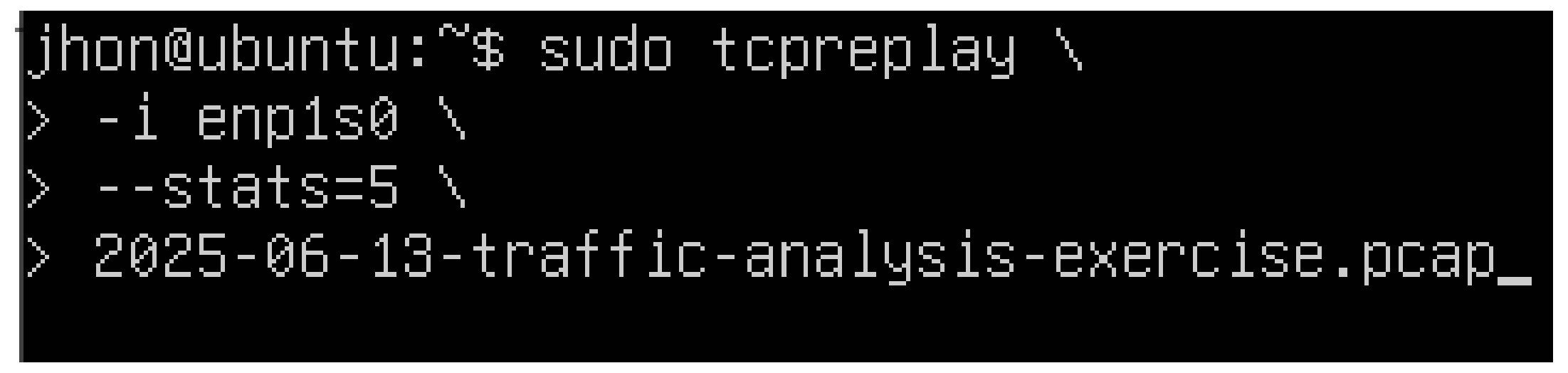


Fig. 1: QEMU/KVM Virutal Machines in Linux Mint Desktop

The Linux Mint machine was virtually placed in-line between the Ubuntu and Lubuntu machine acting as a router between them. Snort 3.3.1.0[9] and Suricata 7.0.3[4] was installed in Linux Mint and configured to monitor the traffic that flowed between the Ubuntu and Lubuntu systems.

Methods

Malware-Traffic-Analysis.net is a website that shares "pcaps and malware samples"[1]. Twenty-nine packet capture files (PCAPS) containing "infected traffic"[1] from 2020 to 2025 were downloaded to Ubuntu and then trasnmitted to Linux Mint using the Tcpreplay command-line tool[10] to see what and how many alerts would be generated by Snort and Suricata on the same traffic.



```
jhon@ubuntu:~$ sudo tcpreplay \
> -i enp1s0 \
> --stats=5 \
> 2025-06-13-traffic-analysis-exercise.pcap_
```

Fig. 2: Replaying Infected Traffic from Ubuntu using Tcpreplay

Snort 3.3.1.0 was installed and as encouraged by the Snort Team[8], the Talos lightSPD rules package was downloaded and passed to the local Snort configuration in Linux Mint. On the other hand, Suricata 7.0.3 was installed using the Advanced Packaging Tool[6] and no further configured with custom rules.

Results

In total, 362,192 packets or about 296.48 megabytes worth of network traffic was analyzed. Below is a table of the first ten PCAP samples, the number of network flows or "Conversations"[5] in each capture, and the number of alerts generated by Snort and Suricata.

PCAP Index	PCAP Name	Packets	Conversations	Suricata Alerts	Snort Alerts
1	2020-01-16-Lokibot-infection-traffic.pcap	295	54	103	59
2	2020-01-23-Ursnif-infection-with-Ursnif-variant-as-follow-up-malware.pcap	3424	119	59	111
3	2020-03-30-Kpot-infection-traffic.pcap	1866	12	2	1
4	2021-02-05-Spelevo-EK-sends-SmokeLoader.pcap	682	12	4	5
5	2021-08-05-AZORult-infection.pcap	5544	8	3	1
6	2022-01-06-TA551-IcedID-infection.pcap	3323	50	11	20
7	2022-06-08-SVCready-infection.pcap	10123	321	210	228
8	2022-07-01-SVCready-infection.pcap	22266	563	544	276
9	2022-08-31-IcedID-with-Cobalt-Strike-carved-and-sanitized.pcap	5692	443	221	1
10	2023-01-03-Rhadamanthys-Stealer-traffic.pcap	1156	5	3	7

After analyzing the 29 network traffic samples, the following observations were made: In 10 occasions, Snort created more alerts than Suricata; In 17 occasions, Suricata generated more alerts than Snort; And in 2 occasions, they generated the same number of alerts on the same traffic.

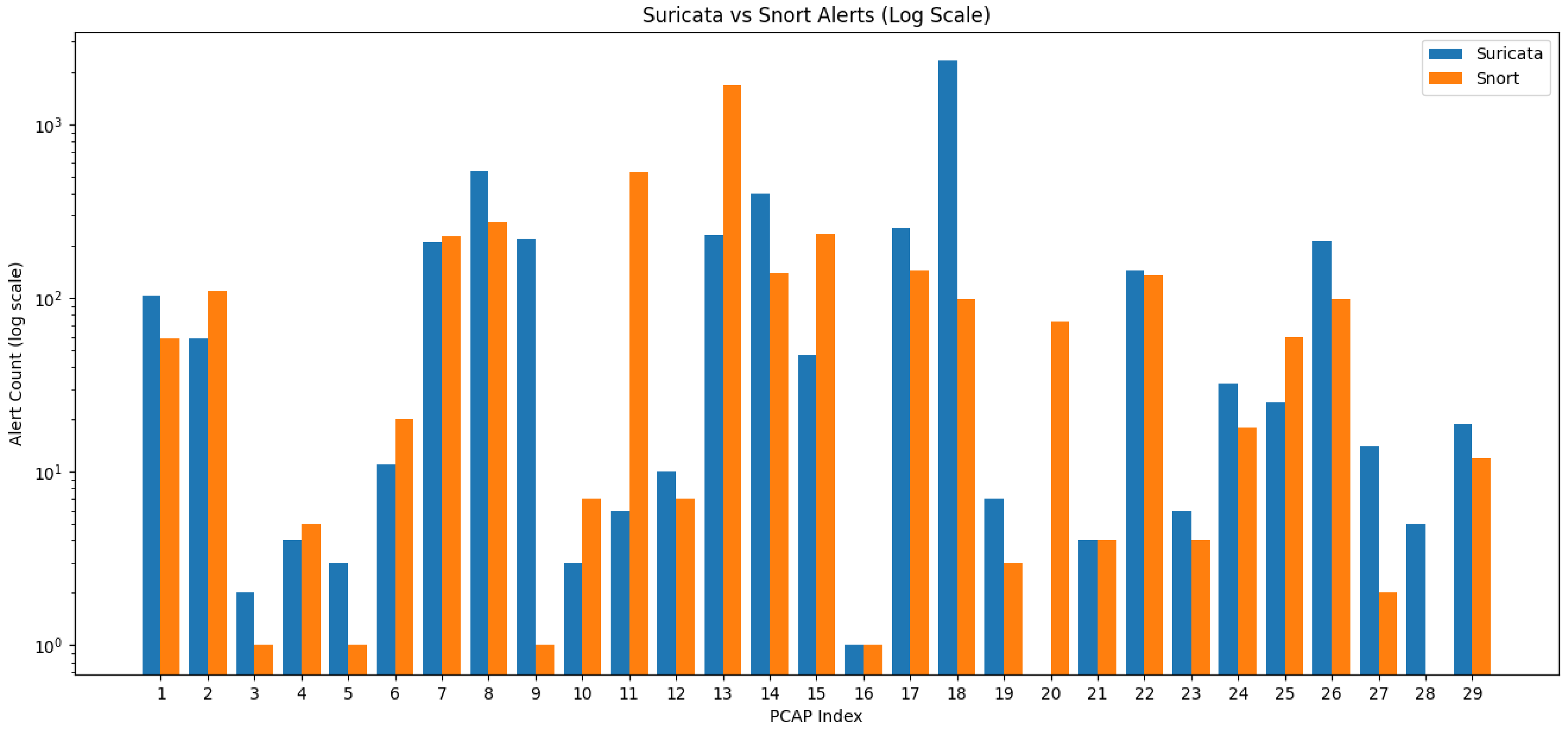


Fig. 3: Suricata vs Snort Alert Count on PCAP samples from Malware-Traffic-Analysis.net

By default, for each PCAP, Suricata generated a stats.log file[3] that included performance metrics such as the amount of memory operations per packet. This and the number of packets per PCAP were used to create the graphical visualization in Figure 4, which showed that memory operations per packet decreased as packet count increased.

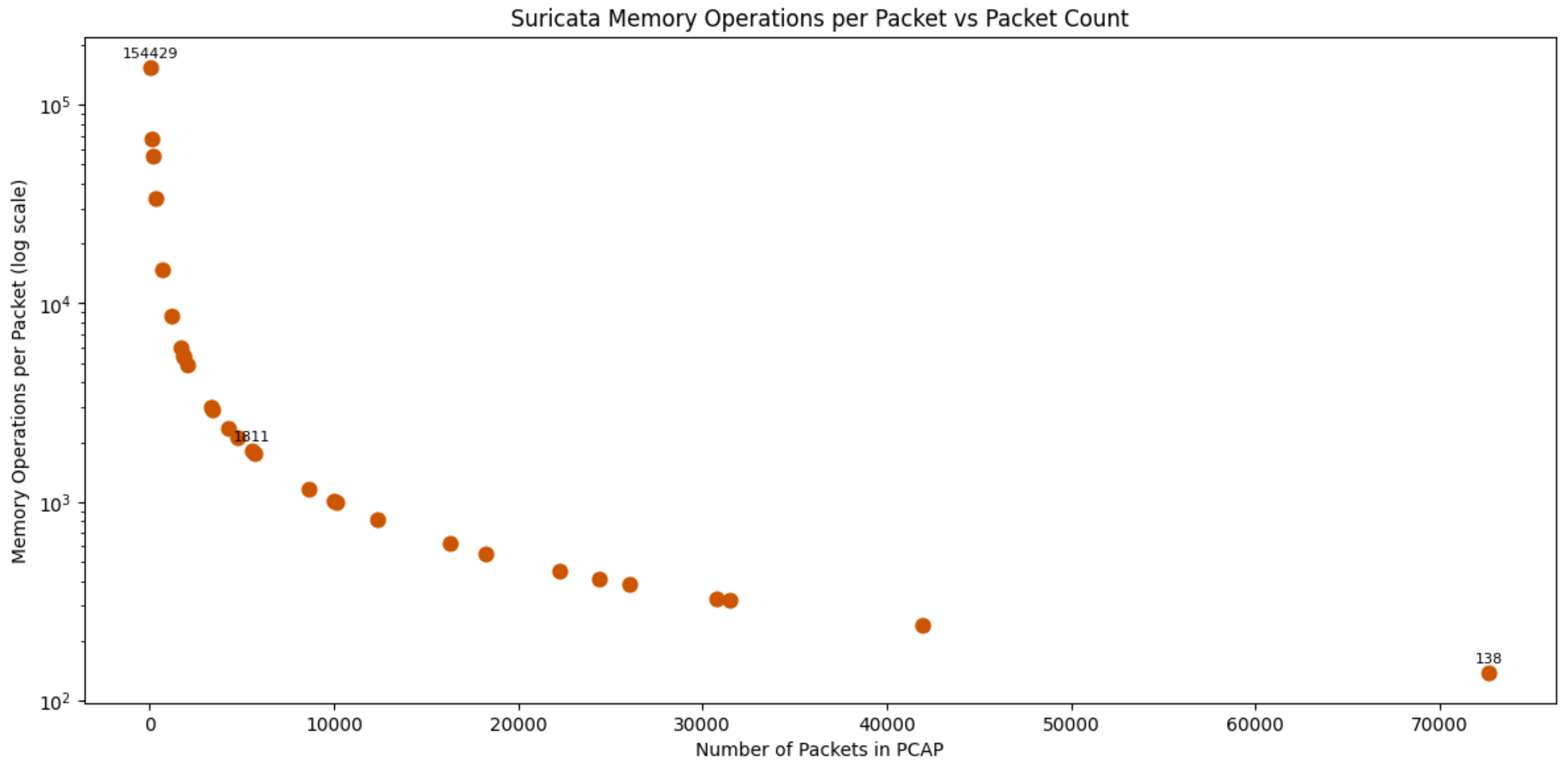


Fig. 4: Memory operatins per packet decreases as packet count increases.

Conclusions

Snort 3.3.1.0 and Suricata 7.0.3 showed they are both capable open-source IDS tools. We observed that each one generetad more alerts than the other in different scenarios. In figure 4, Suricata's built-in performance metrics revealed improved effiency as packet volume incresed, which shows a more transparent insight by default into its behavior. Despite Suricata's alert count being higher than Snort, that did not mean it was more effective than the former one. This is because Suricata used the Emerging Threat Open ruleset[2], while Snort used the lightSPD ruleset. Neither IDS proved to be superior to the other one in every case, but, due to Suricata's ease of installation, ready-to-use configuration, and visibility into its performance, it was selected for the second part of this research. For this reason, future work in this project will focus on expanding IDS and IDPS experimentation and development using Suricata.

Fig. 5: Suricata logo. Source: suricata.io/branding-images.

Acknowledgements

I would like to thank Dr. Merlinda Drini for her guidance in this research project, as well as the CUNY Research Scholars Program. I also thank the creators of Suricata and Snort for their publicly available documentation of their tools, and to the Malware-Traffic-Analysis.net project for proving free network traffic with malware to analyse.

References

[1] Brad Duncan. *Malware-Traffic-Analysis.net*. Accessed: 2025-11-17. 2025. URL: https://www.malware-traffic-analysis.net/.

[2] Suricata – Open Source Threat Detection Engine. *Emerging Threats Open Ruleset*. Accessed: 2025-11-20. 2025. URL: https://docs.suricata.io/en/latest/rules/intro.html/.

[3] Open Information Security Foundation. *12.1. Suricata.yaml — Suricata 9.0.0-dev documentation*. Accessed: 2025-11-20. 2025. URL: https://docs.suricata.io/en/latest/configuration/suricata-yaml.html.

[4] Open Information Security Foundation. *Suricata – Open Source Threat Detection Engine*. Accessed: 2025-11-17. 2025. URL: https://suricata.io.

[5] Kentik. *Network Traffic Analysis*. Accessed: 2025-11-20. 2025. URL: https://www.kentik.com/kentipedia/network-traffic-analysis/.

[6] Canonical Group Ltd. *Install and manage packages — Ubuntu Server documentation*. https://documentation.ubuntu.com/server/how-to/software/package-management. Accessed: 2025-11-18. 2025.

[7] QEMU Project. *QEMU: A Generic and Open Source Machine Emulator and Virtualizer*. Accessed: 2025-11-18. 2025. URL: https://www.qemu.org/.

[8] Talos. *End of Life Announcement for Multiple Versions of Snort 2 and Snort 3*. Accessed: 2025-11-18. 2025. URL: https://blog.snort.org/2025/.

[9] Martin Roesch & The Snort Team. *Snort – Open Source Network Intrusion Prevention System*. Accessed: 2025-11-17. 2025. URL: https://www.snort.org.

[10] Aaron Turner and Fred Klassen. *Tcpreplay - Pcap editing and replaying utilities*. Accessed: 2025-11-20. 2025. URL: https://tcpreplay.appneta.com/.