Prepared By: Gopi Gor

Date: 02/12/2024

Network Attacks Investigation

This report analyses 3 different network attacks and plays a role in incident reporting. This report has been created using Sumologic for lab purposes. Previously parsed logs have been used to run the scripts. Considering a role of an SOC analyst, a report has been prepared for Port Scan Detection, Slammer Worms and Zeus (C2) along with business implications.

Port Scan Detection

As an SOC analyst, to analyse port scan detection, I would first look at specific filters and protocols used in the PCAP files of Cloudshark. I would check the source and destination IPv4 and IPv6 addresses, and find out the details of the IP address. Then, I would look into the tcp filter to understand the sequences that the source is sending to the destination. I would look for incomplete SYN requests, which indicate a closed port or firewall. Furthermore, the **Len** factor can tell us more about if there is a payload or not. Sequence of port numbers could denote a stealth scan for incomplete SYN requests and thus I would examine each packet to investigate the TCP ports, UDP ports, length, src and destination IP and RST packets. A constant RST packet could mean the sender is trying to reset the connection in order to send malicious packets. The criticality I would assign to this alert would be **high** because the open ports scanning could reveal a lot of information to the attacker and knowledge about open ports could lead to unauthorized entry into the network. The dashboard most useful for this would be one that shows incomplete SYN requests, continuous RST requests and Len >0 related dashboards. Mainly, network activity dashboards would be the most useful.

The business impacts for this kind of alert would be that the attacker gets entry into the network and launches malicious payloads, thereby harming the complete organization or disruption of all services. The short-term risks can be security breaches if vulnerabilities are discovered after the port scanning. Long-term risks could be that an APT could be running for a long time without being noticed and thus can be ready to launch a big attack on the organization thereby completely exfiltrating data or exploiting vulnerabilities over a very long time.

To remediate this kind of alert, the first thing is place is the incident response plan, and the network access to be limited only to certain resources in the network. Furthermore, there has to be a team deployed for constant patch management in the network side to keep fixing vulnerabilities incase an attacker gets access to network ports and traffic.

Worms – Slammer

As an SOC Analyst, to investigate the alert for worms, specifically that of the Slammer, the Slammer worm tries to exploit vulnerabilities in the MS SQL database server and thus, I would first filter the PCAP to look for all UDP packets because slammer uses UDP packets for transmission. Then I would look for the source and destination IP, which would majorly have different sources but a common destination IP. After this, I would analyse the UDP packets and try to find a pattern with the payload, which would be 376 bytes, being the slammer payload size. Additionally, I would look for destination ports where the port is 1434, which slammer uses to exploit the vulnerability on the MS SQL server. The criticality I would assign to this is **high** because the worms have a tendency to spread themselves, so once they are in the system, it is more likely to spread very quickly. As for a dashboard, I would use dashboards which show UDP requests and packet details like destination ports, payload length and destination IP addresses.

The business implications of a worm in a network could be vastly different because they could cause major disruptions to the services being provided by the server, thereby resulting in loss of productivity of the server. The short-term risks could be service unavailability and DDoS resulting in complete network congestion during the early stages. Long-term risks include persistent security risks and a lot of challenges with regards to business continuity. The server and domains can be completed banned, and there could be compliance-based charges imposed on the organization too.

To remediate this alert type, the affected systems can be made isolated systems and then they can investigate forensically to understand the cause of the attack. Moreover, security patches could be based on the analysis and investigations made. The antivirus and security softwares should be updated and policies could be reinforced to understand all different tactics the slammer has used to infiltrate the network. Lastly, all compromised accounts, PC's and users may need to be newly created or updated in order to avoid future events like these.

Command & Control – Zeus

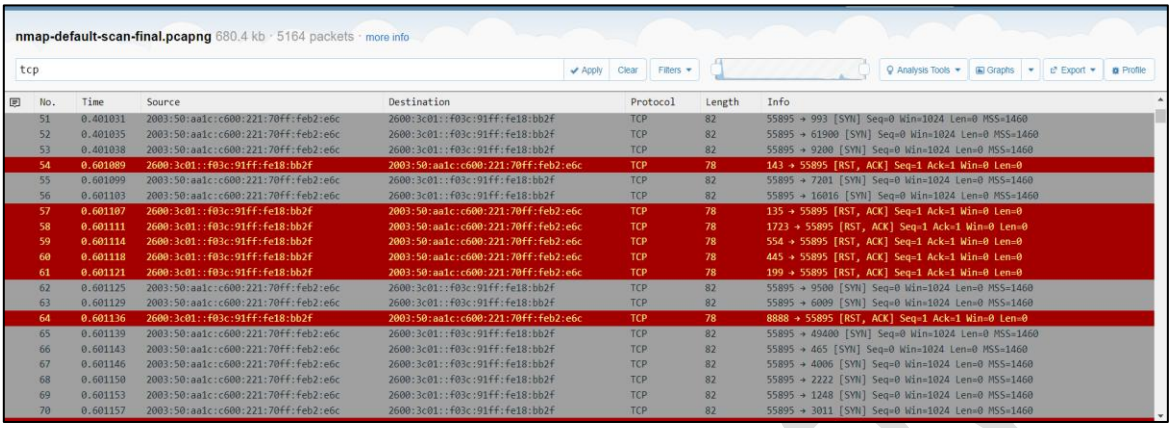
As an SOC Analyst, I would first look for investigation in the network traffic and irregular spikes in network traffic. Zeus botnets use a lot of HTTP and HTTPS traffic which may or may not have random user agent strings. I would also try to try to understand all random IP addresses an investigation on those IP addresses to understand the legitimacy of those IPs. The DNS traffic would show quite a lot of domain names if a Zeus botnet has been employed in the network. If we check the ports, there would be multiple random ports, where normal services don't run and they would be very high numbered ports. Additionally, a high payload in the len part could mean there is a malicious payload being injected into the system. The criticality I would assign to this alert type would be **high** due to the multiple botnets that can be employed if one part of the network gets filtrated. The dashboards that could be useful for investigating this alert type could be protocol-based dashboards like payloads, ssh, tls or https requests, packet analysis filters and timeline dashboards.

The Zeus botnet is designed to steal all kinds of sensitive data, so it could cost a lot of personal information to the organization along with financial loss. The short-term risks could be operational risk and service disruptions, along with stealing of personal data. In long-term risks, the data that was exfiltrated could be used to launch impersonation attacks or selling of data on the dark web. It could also cause loss of a competitive advantage for the organization.

For quick and long-term remediation, the network access should be immediately disabled to avoid any further botnets being created inside the network. The affected systems need to be completed wipes and sanitized to remove all the viruses that may have entered into the network. After this, network segmentation can be applied to keep all critical and sensitive servers on a local server to avoid any connections to the outside world. Lastly, continuous network monitoring needs to be implemented.

Appendix

1. Port Scanning

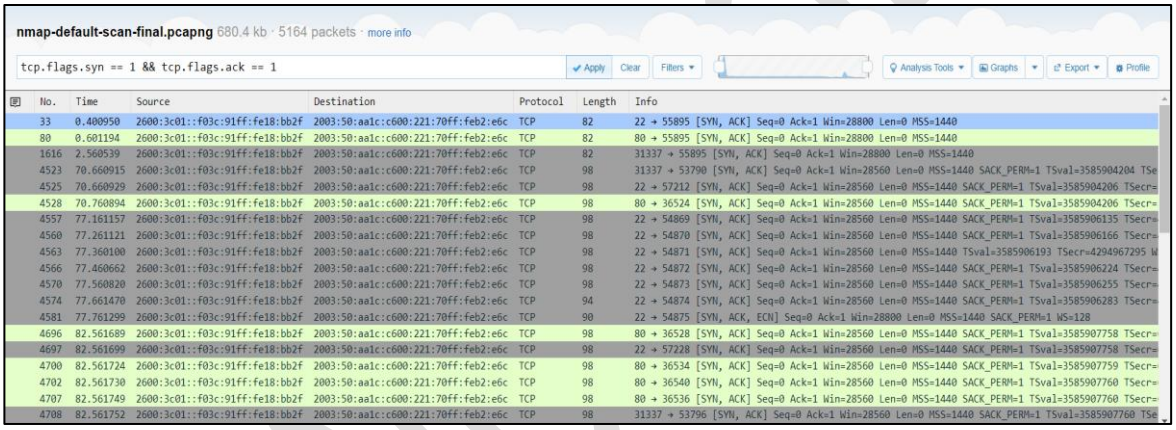


nmap-default-scan-final.pcapng 680.4 kb · 5164 packets · more info

tcp

No.	Time	Source	Destination	Protocol	Length	Info
51	0.401031	2003:50:aa1c:c600:221:70ff:feb2:e6c	2600:3c01::f03c:91ff:fe18:bb2f	TCP	82	55895 → 993 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
52	0.401035	2003:50:aa1c:c600:221:70ff:feb2:e6c	2600:3c01::f03c:91ff:fe18:bb2f	TCP	82	55895 → 61900 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
53	0.401038	2003:50:aa1c:c600:221:70ff:feb2:e6c	2600:3c01::f03c:91ff:fe18:bb2f	TCP	82	55895 → 9200 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
54	0.601089	2600:3c01::f03c:91ff:fe18:bb2f	2003:50:aa1c:c600:221:70ff:feb2:e6c	TCP	78	143 → 55895 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
55	0.601099	2003:50:aa1c:c600:221:70ff:feb2:e6c	2600:3c01::f03c:91ff:fe18:bb2f	TCP	82	55895 → 7201 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
56	0.601103	2003:50:aa1c:c600:221:70ff:feb2:e6c	2600:3c01::f03c:91ff:fe18:bb2f	TCP	82	55895 → 16016 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
57	0.601107	2600:3c01::f03c:91ff:fe18:bb2f	2003:50:aa1c:c600:221:70ff:feb2:e6c	TCP	78	135 → 55895 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
58	0.601111	2600:3c01::f03c:91ff:fe18:bb2f	2003:50:aa1c:c600:221:70ff:feb2:e6c	TCP	78	1723 → 55895 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
59	0.601114	2600:3c01::f03c:91ff:fe18:bb2f	2003:50:aa1c:c600:221:70ff:feb2:e6c	TCP	78	554 → 55895 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
60	0.601118	2600:3c01::f03c:91ff:fe18:bb2f	2003:50:aa1c:c600:221:70ff:feb2:e6c	TCP	78	445 → 55895 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
61	0.601121	2600:3c01::f03c:91ff:fe18:bb2f	2003:50:aa1c:c600:221:70ff:feb2:e6c	TCP	78	199 → 55895 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
62	0.601125	2003:50:aa1c:c600:221:70ff:feb2:e6c	2600:3c01::f03c:91ff:fe18:bb2f	TCP	82	55895 → 9500 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
63	0.601129	2003:50:aa1c:c600:221:70ff:feb2:e6c	2600:3c01::f03c:91ff:fe18:bb2f	TCP	82	55895 → 6009 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
64	0.601136	2600:3c01::f03c:91ff:fe18:bb2f	2003:50:aa1c:c600:221:70ff:feb2:e6c	TCP	78	8888 → 55895 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
65	0.601139	2003:50:aa1c:c600:221:70ff:feb2:e6c	2600:3c01::f03c:91ff:fe18:bb2f	TCP	82	55895 → 49400 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
66	0.601143	2003:50:aa1c:c600:221:70ff:feb2:e6c	2600:3c01::f03c:91ff:fe18:bb2f	TCP	82	55895 → 465 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
67	0.601146	2003:50:aa1c:c600:221:70ff:feb2:e6c	2600:3c01::f03c:91ff:fe18:bb2f	TCP	82	55895 → 4006 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
68	0.601150	2003:50:aa1c:c600:221:70ff:feb2:e6c	2600:3c01::f03c:91ff:fe18:bb2f	TCP	82	55895 → 2222 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
69	0.601153	2003:50:aa1c:c600:221:70ff:feb2:e6c	2600:3c01::f03c:91ff:fe18:bb2f	TCP	82	55895 → 1248 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
70	0.601157	2003:50:aa1c:c600:221:70ff:feb2:e6c	2600:3c01::f03c:91ff:fe18:bb2f	TCP	82	55895 → 3011 [SYN] Seq=0 Win=1024 Len=0 MSS=1460

Fig. 1. TCP to notice open ports from destination.

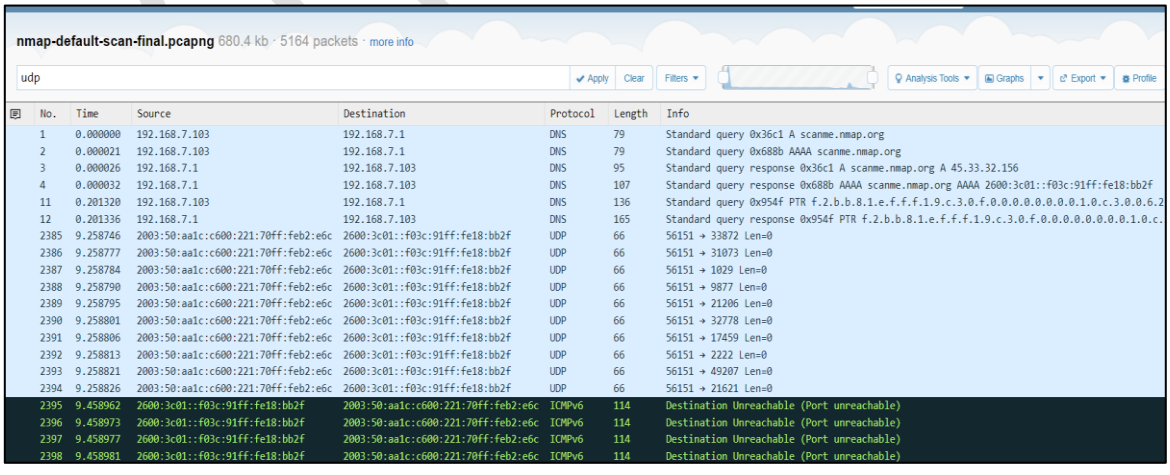


nmap-default-scan-final.pcapng 680.4 kb · 5164 packets · more info

tcp.flags.syn == 1 && tcp.flags.ack == 1

No.	Time	Source	Destination	Protocol	Length	Info
33	0.400950	2600:3c01::f03c:91ff:fe18:bb2f	2003:50:aa1c:c600:221:70ff:feb2:e6c	TCP	82	22 → 55895 [SYN, ACK] Seq=0 Ack=1 Win=28800 Len=0 MSS=1440
80	0.601194	2600:3c01::f03c:91ff:fe18:bb2f	2003:50:aa1c:c600:221:70ff:feb2:e6c	TCP	82	80 → 55895 [SYN, ACK] Seq=0 Ack=1 Win=28800 Len=0 MSS=1440
1616	2.560539	2600:3c01::f03c:91ff:fe18:bb2f	2003:50:aa1c:c600:221:70ff:feb2:e6c	TCP	82	31337 → 55895 [SYN, ACK] Seq=0 Ack=1 Win=28800 Len=0 MSS=1440
4523	70.660915	2600:3c01::f03c:91ff:fe18:bb2f	2003:50:aa1c:c600:221:70ff:feb2:e6c	TCP	98	31337 → 53790 [SYN, ACK] Seq=0 Ack=1 Win=28560 Len=0 MSS=1440 SACK_PERM=1 TSval=3585904204 TSe
4525	70.660929	2600:3c01::f03c:91ff:fe18:bb2f	2003:50:aa1c:c600:221:70ff:feb2:e6c	TCP	98	22 → 57212 [SYN, ACK] Seq=0 Ack=1 Win=28560 Len=0 MSS=1440 SACK_PERM=1 TSval=3585904206 TSecr=
4528	70.760894	2600:3c01::f03c:91ff:fe18:bb2f	2003:50:aa1c:c600:221:70ff:feb2:e6c	TCP	98	80 → 36524 [SYN, ACK] Seq=0 Ack=1 Win=28560 Len=0 MSS=1440 SACK_PERM=1 TSval=3585904206 TSecr=
4557	77.161157	2600:3c01::f03c:91ff:fe18:bb2f	2003:50:aa1c:c600:221:70ff:feb2:e6c	TCP	98	22 → 54869 [SYN, ACK] Seq=0 Ack=1 Win=28560 Len=0 MSS=1440 SACK_PERM=1 TSval=3585906135 TSecr=
4560	77.261121	2600:3c01::f03c:91ff:fe18:bb2f	2003:50:aa1c:c600:221:70ff:feb2:e6c	TCP	98	22 → 54870 [SYN, ACK] Seq=0 Ack=1 Win=28560 Len=0 MSS=1440 SACK_PERM=1 TSval=3585906166 TSecr=
4563	77.360100	2600:3c01::f03c:91ff:fe18:bb2f	2003:50:aa1c:c600:221:70ff:feb2:e6c	TCP	98	22 → 54871 [SYN, ACK] Seq=0 Ack=1 Win=28560 Len=0 MSS=1440 TSval=3585906193 TSecr=4294967295 W
4566	77.460662	2600:3c01::f03c:91ff:fe18:bb2f	2003:50:aa1c:c600:221:70ff:feb2:e6c	TCP	98	22 → 54872 [SYN, ACK] Seq=0 Ack=1 Win=28560 Len=0 MSS=1440 SACK_PERM=1 TSval=3585906224 TSecr=
4570	77.560820	2600:3c01::f03c:91ff:fe18:bb2f	2003:50:aa1c:c600:221:70ff:feb2:e6c	TCP	98	22 → 54873 [SYN, ACK] Seq=0 Ack=1 Win=28560 Len=0 MSS=1440 SACK_PERM=1 TSval=3585906255 TSecr=
4574	77.661470	2600:3c01::f03c:91ff:fe18:bb2f	2003:50:aa1c:c600:221:70ff:feb2:e6c	TCP	94	22 → 54874 [SYN, ACK] Seq=0 Ack=1 Win=28560 Len=0 MSS=1440 SACK_PERM=1 TSval=3585906283 TSecr=
4581	77.761299	2600:3c01::f03c:91ff:fe18:bb2f	2003:50:aa1c:c600:221:70ff:feb2:e6c	TCP	90	22 → 54875 [SYN, ACK, ECN] Seq=0 Ack=1 Win=28800 Len=0 MSS=1440 SACK_PERM=1 WS=128
4696	82.561689	2600:3c01::f03c:91ff:fe18:bb2f	2003:50:aa1c:c600:221:70ff:feb2:e6c	TCP	98	80 → 36528 [SYN, ACK] Seq=0 Ack=1 Win=28560 Len=0 MSS=1440 SACK_PERM=1 TSval=3585907758 TSecr=
4697	82.561699	2600:3c01::f03c:91ff:fe18:bb2f	2003:50:aa1c:c600:221:70ff:feb2:e6c	TCP	98	22 → 57228 [SYN, ACK] Seq=0 Ack=1 Win=28560 Len=0 MSS=1440 SACK_PERM=1 TSval=3585907758 TSecr=
4700	82.561724	2600:3c01::f03c:91ff:fe18:bb2f	2003:50:aa1c:c600:221:70ff:feb2:e6c	TCP	98	80 → 36534 [SYN, ACK] Seq=0 Ack=1 Win=28560 Len=0 MSS=1440 SACK_PERM=1 TSval=3585907759 TSecr=
4702	82.561730	2600:3c01::f03c:91ff:fe18:bb2f	2003:50:aa1c:c600:221:70ff:feb2:e6c	TCP	98	80 → 36540 [SYN, ACK] Seq=0 Ack=1 Win=28560 Len=0 MSS=1440 SACK_PERM=1 TSval=3585907760 TSecr=
4707	82.561749	2600:3c01::f03c:91ff:fe18:bb2f	2003:50:aa1c:c600:221:70ff:feb2:e6c	TCP	98	80 → 36536 [SYN, ACK] Seq=0 Ack=1 Win=28560 Len=0 MSS=1440 SACK_PERM=1 TSval=3585907760 TSecr=
4708	82.561752	2600:3c01::f03c:91ff:fe18:bb2f	2003:50:aa1c:c600:221:70ff:feb2:e6c	TCP	98	31337 → 53796 [SYN, ACK] Seq=0 Ack=1 Win=28560 Len=0 MSS=1440 SACK_PERM=1 TSval=3585907760 TSe

Fig. 2. TCP to set ack and syn flags as 1 to show complete connections



nmap-default-scan-final.pcapng 680.4 kb · 5164 packets · more info

udp

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	192.168.7.103	192.168.7.1	DNS	79	Standard query 0x36c1 A scanme.nmap.org
2	0.000021	192.168.7.103	192.168.7.1	DNS	79	Standard query 0x688b AAAA scanme.nmap.org
3	0.000026	192.168.7.1	192.168.7.103	DNS	95	Standard query response 0x36c1 A scanme.nmap.org A 45.33.32.156
4	0.000032	192.168.7.1	192.168.7.103	DNS	107	Standard query response 0x688b AAAA scanme.nmap.org AAAA 2600:3c01::f03c:91ff:fe18:bb2f
11	0.201320	192.168.7.103	192.168.7.1	DNS	136	Standard query 0x954f PTR f.2.b.b.8.1.e.f.f.f.1.9.c.3.0.f.0.0.0.0.0.0.0.0.1.0.c.3.0.0.6.2
12	0.201336	192.168.7.1	192.168.7.103	DNS	165	Standard query response 0x954f PTR f.2.b.b.8.1.e.f.f.f.1.9.c.3.0.f.0.0.0.0.0.0.0.0.1.0.c.
2385	9.258746	2003:50:aa1c:c600:221:70ff:feb2:e6c	2600:3c01::f03c:91ff:fe18:bb2f	UDP	66	56151 → 33872 Len=0
2386	9.258777	2003:50:aa1c:c600:221:70ff:feb2:e6c	2600:3c01::f03c:91ff:fe18:bb2f	UDP	66	56151 → 31073 Len=0
2387	9.258784	2003:50:aa1c:c600:221:70ff:feb2:e6c	2600:3c01::f03c:91ff:fe18:bb2f	UDP	66	56151 → 1029 Len=0
2388	9.258790	2003:50:aa1c:c600:221:70ff:feb2:e6c	2600:3c01::f03c:91ff:fe18:bb2f	UDP	66	56151 → 9877 Len=0
2389	9.258795	2003:50:aa1c:c600:221:70ff:feb2:e6c	2600:3c01::f03c:91ff:fe18:bb2f	UDP	66	56151 → 21206 Len=0
2390	9.258801	2003:50:aa1c:c600:221:70ff:feb2:e6c	2600:3c01::f03c:91ff:fe18:bb2f	UDP	66	56151 → 32778 Len=0
2391	9.258806	2003:50:aa1c:c600:221:70ff:feb2:e6c	2600:3c01::f03c:91ff:fe18:bb2f	UDP	66	56151 → 17459 Len=0
2392	9.258813	2003:50:aa1c:c600:221:70ff:feb2:e6c	2600:3c01::f03c:91ff:fe18:bb2f	UDP	66	56151 → 2222 Len=0
2393	9.258821	2003:50:aa1c:c600:221:70ff:feb2:e6c	2600:3c01::f03c:91ff:fe18:bb2f	UDP	66	56151 → 49207 Len=0
2394	9.258826	2003:50:aa1c:c600:221:70ff:feb2:e6c	2600:3c01::f03c:91ff:fe18:bb2f	UDP	66	56151 → 21621 Len=0
2395	9.458962	2600:3c01::f03c:91ff:fe18:bb2f	2003:50:aa1c:c600:221:70ff:feb2:e6c	ICMPv6	114	Destination Unreachable (Port unreachable)
2396	9.458973	2600:3c01::f03c:91ff:fe18:bb2f	2003:50:aa1c:c600:221:70ff:feb2:e6c	ICMPv6	114	Destination Unreachable (Port unreachable)
2397	9.458977	2600:3c01::f03c:91ff:fe18:bb2f	2003:50:aa1c:c600:221:70ff:feb2:e6c	ICMPv6	114	Destination Unreachable (Port unreachable)
2398	9.458981	2600:3c01::f03c:91ff:fe18:bb2f	2003:50:aa1c:c600:221:70ff:feb2:e6c	ICMPv6	114	Destination Unreachable (Port unreachable)

Fig. 3. UDP to check to check if ports are open now.

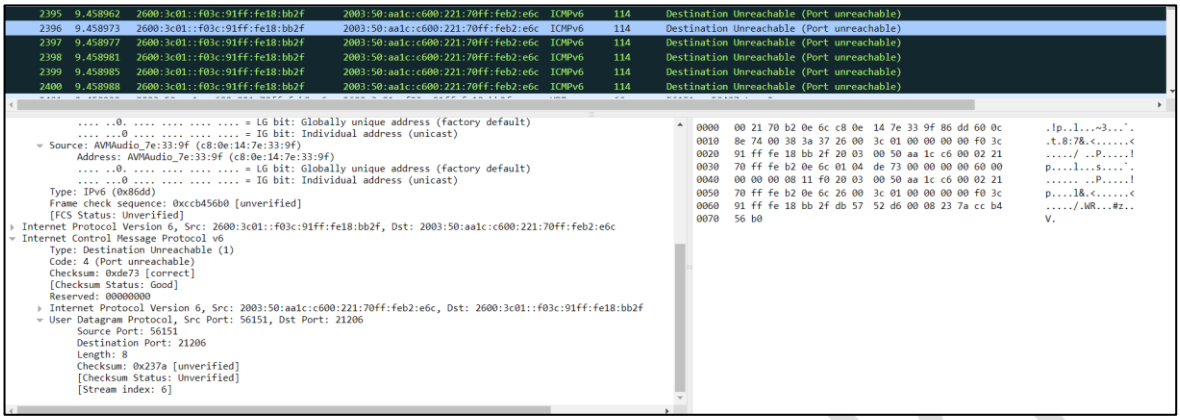


Fig. 4. UDP packets to see that ports are not reachable and same src and dest.IP.

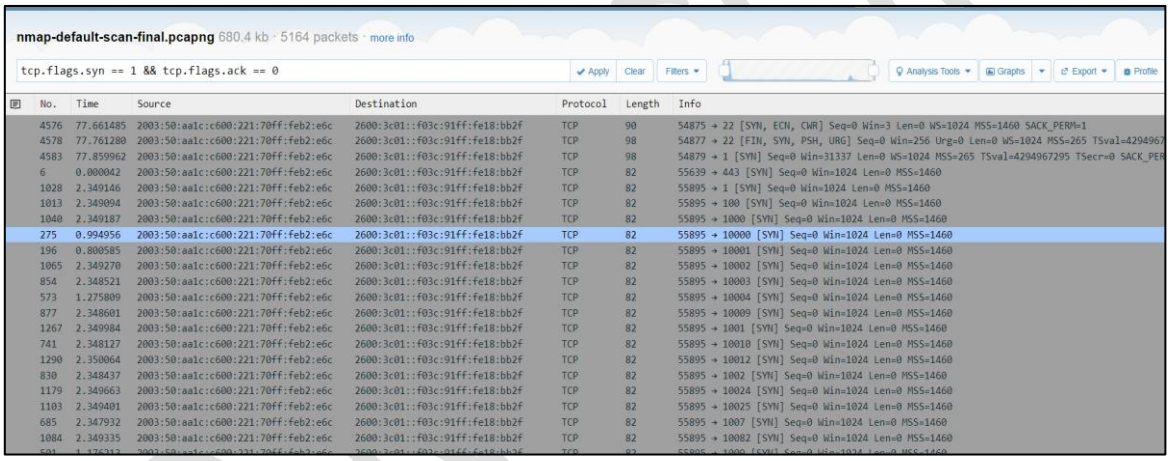


Fig. 5. TCP flags to set syn to 1 and ack to 0 for incomplete transmission

2. Worms - Slammer

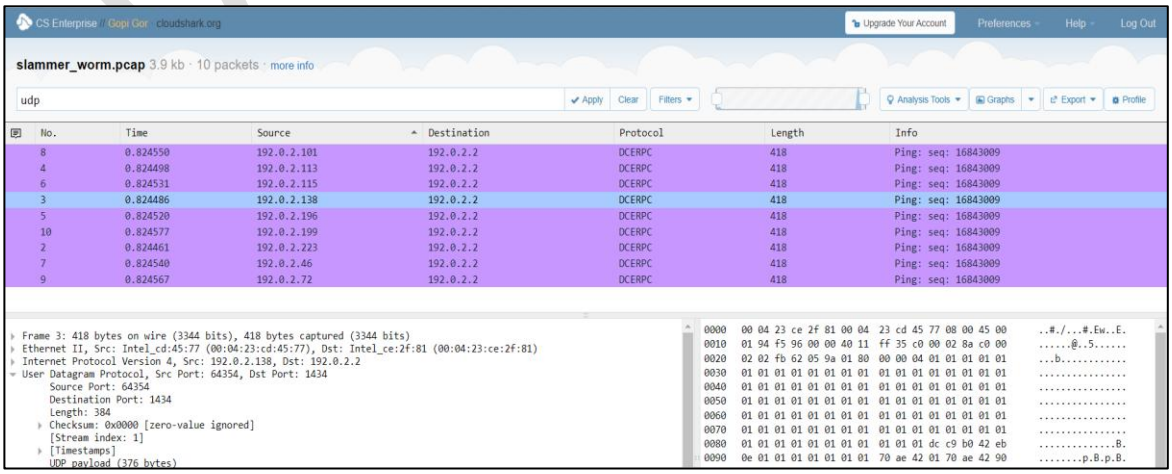


Fig. 6. UDP showing dest. port of 1434 to exploit MS SQL DB vulnerabilities

3. C&C – Zeus Botnet

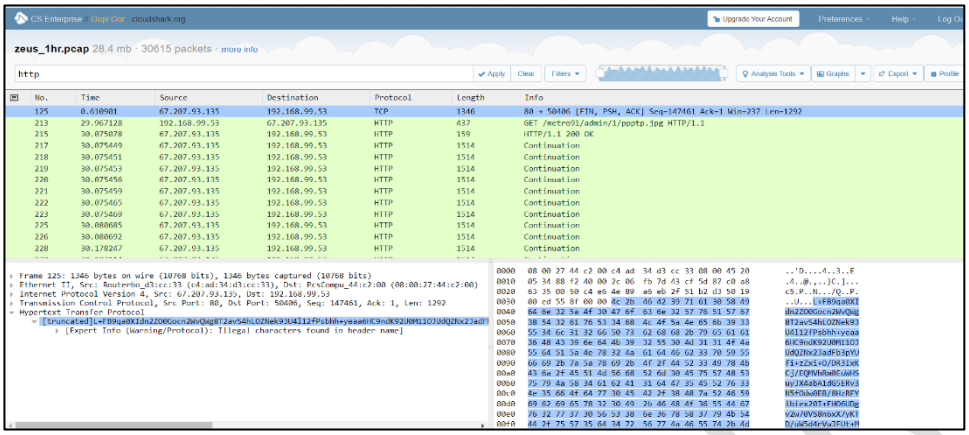


Fig. 7. UDP to check to check if ports are open now.

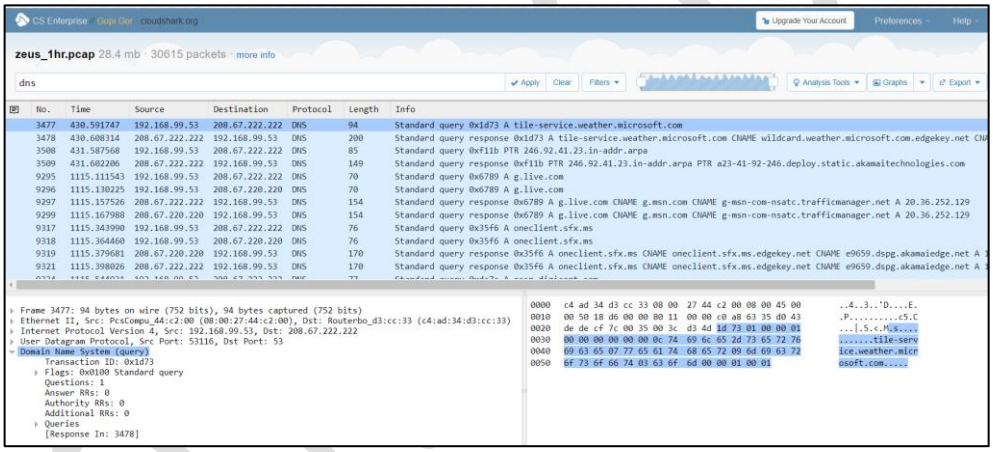


Fig. 8. UDP to check to check if ports are open now.

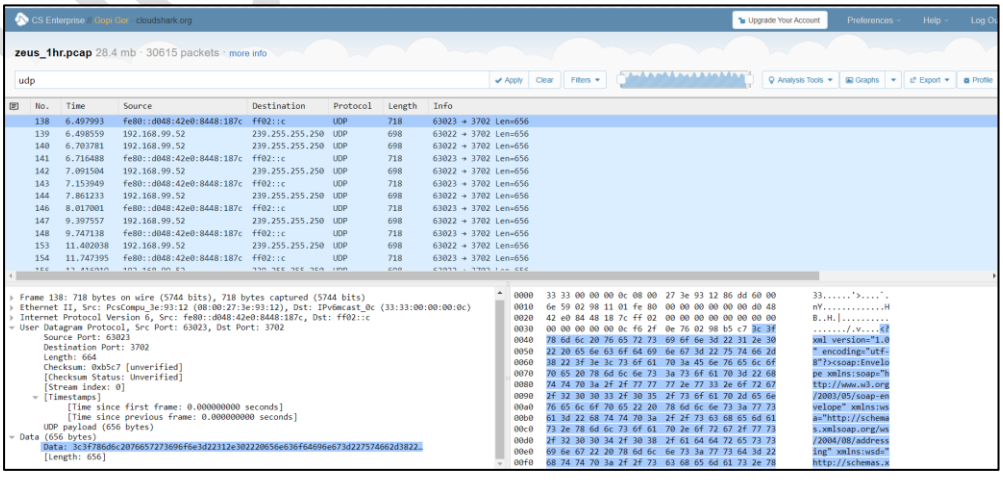


Fig. 9. UDP to check to check if ports are open now.

References

1. https://www.crowdstrike.com/cybersecurity-101/malware/trojan-zeus-malware/
2. https://cloud.ibm.com/docs/codeengine?topic=codeengine-ts-app-toomanyports
3. https://www.geeksforgeeks.org/slammer-worm-in-information-security/
4. https://www.csoonline.com/article/566849/sql-slammer-16-years-later-four-modern-day-scenarios-that-could-be-worse.html
5. https://security.stackexchange.com/questions/59418/how-to-detect-the-so-called-undetectable-gameover-zeus