# Network Analysis – Malware Compromise

## Contents

## Scenario

A SOC Analyst at Umbrella Corporation is going through SIEM alerts and sees the alert for connections to a known malicious domain. The traffic is coming from Sara's computer, an Accountant who receives a large volume of emails from customers daily. Looking at the email gateway logs for Sara's mailbox there is nothing immediately suspicious, with emails coming from customers. Sara is contacted via her phone and she states a customer sent her an invoice that had a document with a macro, she opened the email and the program crashed. The SOC Team retrieved a PCAP for further analysis.

## Pre-requisites

- Load kali
- Run sudo apt update && sudo apt -y upgrade > reboot
- Change network to host only instead of NAT – to restrict network so malware inside the pcap is contained within the VM
- Load Wireshark and open the pcap file

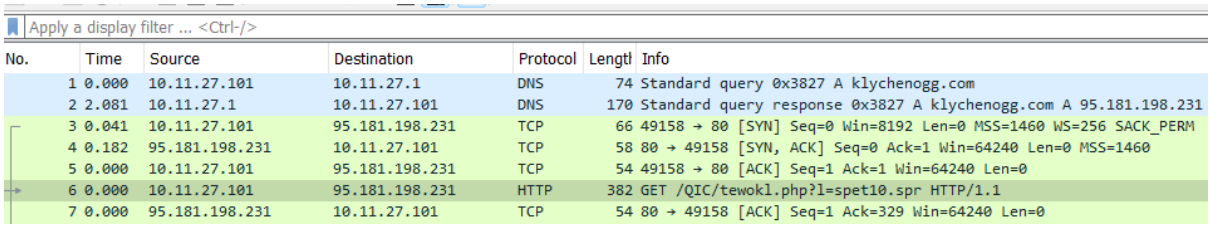
## Initial thoughts from scenario

- Known malicious domain, may be able to search malware sample on VirusTotal?
- Traffic from staff member's (Sara's) pc, be on the lookout for internal/private IP addresses in the pcap
- Nothing immediately suspicious in the inbox, meaning that it could be BEC, supply chain attack or spoofed to appear to be a real client contact
- Document with a macro, look out for MS Word attachments/documents in the pcap > clicking the macro caused the crash so the activity from the malware may be near this packet in the pcap?

## Challenge Questions

### What's the private IP of the infected host?

- Answer: 10.11.27.101

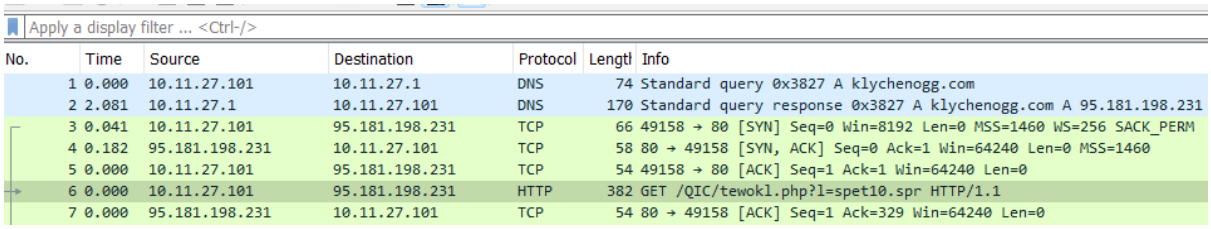



| No. | Time | Source | Destination | Protocol | Lengtl | Info |
|---|---|---|---|---|---|---|
| 1 | 0.000 | 10.11.27.101 | 10.11.27.1 | DNS | 74 | Standard query 0x3827 A klychenogg.com |
| 2 | 2.081 | 10.11.27.1 | 10.11.27.101 | DNS | 170 | Standard query response 0x3827 A klychenogg.com A 95.181.198.231 |
| 3 | 0.041 | 10.11.27.101 | 95.181.198.231 | TCP | 66 | 49158 → 80 [SYN] Seq=0 Win=8192 Len=0 MSS=1460 WS=256 SACK_PERM |
| 4 | 0.182 | 95.181.198.231 | 10.11.27.101 | TCP | 58 | 80 → 49158 [SYN, ACK] Seq=0 Ack=1 Win=64240 Len=0 MSS=1460 |
| 5 | 0.000 | 10.11.27.101 | 95.181.198.231 | TCP | 54 | 49158 → 80 [ACK] Seq=1 Ack=1 Win=64240 Len=0 |
| 6 | 0.000 | 10.11.27.101 | 95.181.198.231 | HTTP | 382 | GET /QIC/tewokl.php?l=spet10.spr HTTP/1.1 |
| 7 | 0.000 | 95.181.198.231 | 10.11.27.101 | TCP | 54 | 80 → 49158 [ACK] Seq=1 Ack=329 Win=64240 Len=0 |

- Loading the pcap and reviewing the first few packets, we immediately see a suspicious-looking GET request in packet 6
- We know from the scenario that an internal pc was compromised by the malware, and when malware is executed a common initial step is for it to beacon to a C2 server to download further scripts
    - Packet 6 shows the get request going to /QIC/tewok.php?l=spet10.spr, given that this is suspicious it could be an extra requirement triggered by the executed malware
- And since the source of this request/packet in the pcap is 10.11.27.101, we can confidently say that this is the private IP of the infected host

## What's the malware binary that the macro document is trying to retrieve?

- Answer: spet10.spr



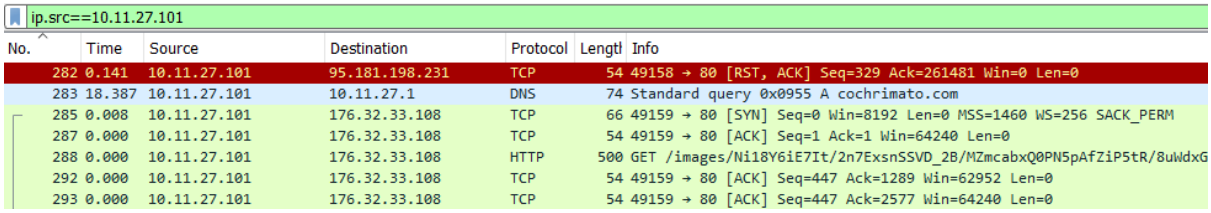
Apply a display filter ... <Ctrl-/>

| No. | Time | Source | Destination | Protocol | Length | Info |
|---|---|---|---|---|---|---|
| 1 | 0.000 | 10.11.27.101 | 10.11.27.1 | DNS | 74 | Standard query 0x3827 A klychenogg.com |
| 2 | 2.081 | 10.11.27.1 | 10.11.27.101 | DNS | 170 | Standard query response 0x3827 A klychenogg.com A 95.181.198.231 |
| 3 | 0.041 | 10.11.27.101 | 95.181.198.231 | TCP | 66 | 49158 → 80 [SYN] Seq=0 Win=8192 Len=0 MSS=1460 WS=256 SACK_PERM |
| 4 | 0.182 | 95.181.198.231 | 10.11.27.101 | TCP | 58 | 80 → 49158 [SYN, ACK] Seq=0 Ack=1 Win=64240 Len=0 MSS=1460 |
| 5 | 0.000 | 10.11.27.101 | 95.181.198.231 | TCP | 54 | 49158 → 80 [ACK] Seq=1 Ack=1 Win=64240 Len=0 |
| 6 | 0.000 | 10.11.27.101 | 95.181.198.231 | HTTP | 382 | GET /QIC/tewokl.php?l=spet10.spr HTTP/1.1 |
| 7 | 0.000 | 95.181.198.231 | 10.11.27.101 | TCP | 54 | 80 → 49158 [ACK] Seq=1 Ack=329 Win=64240 Len=0 |

- As identified in the previous task/question, packet 6 looks suspicious
- Downloads are commonly done using HTTP GET requests, this combined with the fact that the request originates from the victim pc/IP address confirms this is the malicious attachment

## From what domain HTTP requests with GET /images/ are coming from?

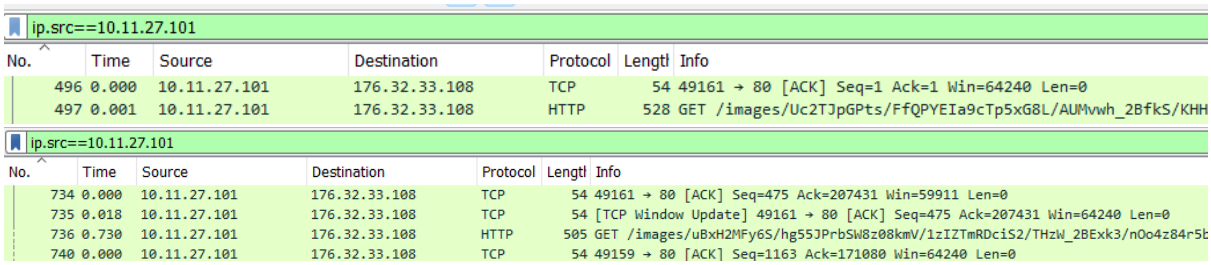
- Answer: cochrimato.com



ip.src==10.11.27.101

| No. | Time | Source | Destination | Protocol | Length | Info |
|---|---|---|---|---|---|---|
| 282 | 0.141 | 10.11.27.101 | 95.181.198.231 | TCP | 54 | 49158 → 80 [RST, ACK] Seq=329 Ack=261481 Win=0 Len=0 |
| 283 | 18.387 | 10.11.27.101 | 10.11.27.1 | DNS | 74 | Standard query 0x0955 A cochrimato.com |
| 285 | 0.008 | 10.11.27.101 | 176.32.33.108 | TCP | 66 | 49159 → 80 [SYN] Seq=0 Win=8192 Len=0 MSS=1460 WS=256 SACK_PERM |
| 287 | 0.000 | 10.11.27.101 | 176.32.33.108 | TCP | 54 | 49159 → 80 [ACK] Seq=1 Ack=1 Win=64240 Len=0 |
| 288 | 0.000 | 10.11.27.101 | 176.32.33.108 | HTTP | 500 | GET /images/Ni18Y6iE7It/2n7ExsnSSVD_2B/MZmcabxQ0PN5pAfZiP5tR/8uWdxG |
| 292 | 0.000 | 10.11.27.101 | 176.32.33.108 | TCP | 54 | 49159 → 80 [ACK] Seq=447 Ack=1289 Win=62952 Len=0 |
| 293 | 0.000 | 10.11.27.101 | 176.32.33.108 | TCP | 54 | 49159 → 80 [ACK] Seq=447 Ack=2577 Win=64240 Len=0 |

- We now know the victim IP address, so apply a filter of ip.src=10.11.27.101 to limit the pcap view to that of traffic leaving JUST that pc
- After the download of spet10.spr, the pcap shows a bunch of TCP traffic and ACK requests/packets which don't offer much information
- As we're looking for /images/ we can review the filtered view for HTTP GET requests with that in the Info column
    - The first request we find that matches this criteria is packet 288
    - When a HTTP GET request is performed, a DNS (Domain Name System) search is first performed to see if the requested domain is in the users browser cache; if not then a recursive search is performed to find the domain requested and return it
        - Knowing this, we can scroll up to view earlier packets in the filtered view for the DNS query that occurred shortly before the HTTP GET request for /images to find a potential domain of cochrimato.com
        - However this finding alone does not provide enough proof that this is the correct/desired domain



ip.src==10.11.27.101

| No. | Time | Source | Destination | Protocol | Length | Info |
|---|---|---|---|---|---|---|
| 496 | 0.000 | 10.11.27.101 | 176.32.33.108 | TCP | 54 | 49161 → 80 [ACK] Seq=1 Ack=1 Win=64240 Len=0 |
| 497 | 0.001 | 10.11.27.101 | 176.32.33.108 | HTTP | 528 | GET /images/Uc2TJpGPts/FfQPYEIa9cTp5xG8L/AUMvwh_2BfkS/KHH |

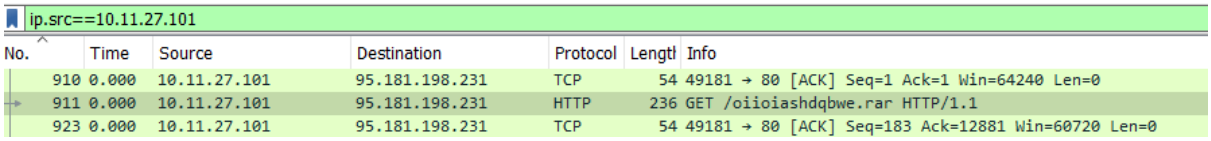
ip.src==10.11.27.101

| No. | Time | Source | Destination | Protocol | Length | Info |
|---|---|---|---|---|---|---|
| 734 | 0.000 | 10.11.27.101 | 176.32.33.108 | TCP | 54 | 49161 → 80 [ACK] Seq=475 Ack=207431 Win=59911 Len=0 |
| 735 | 0.018 | 10.11.27.101 | 176.32.33.108 | TCP | 54 | [TCP Window Update] 49161 → 80 [ACK] Seq=475 Ack=207431 Win=64240 Len=0 |
| 736 | 0.730 | 10.11.27.101 | 176.32.33.108 | HTTP | 505 | GET /images/uBxH2MFy6S/hg55JPrbSW8z08kmV/1zIZTmRDciS2/THzW_2BExk3/nOo4z84r5b |
| 740 | 0.000 | 10.11.27.101 | 176.32.33.108 | TCP | 54 | 49159 → 80 [ACK] Seq=1163 Ack=171080 Win=64240 Len=0 |

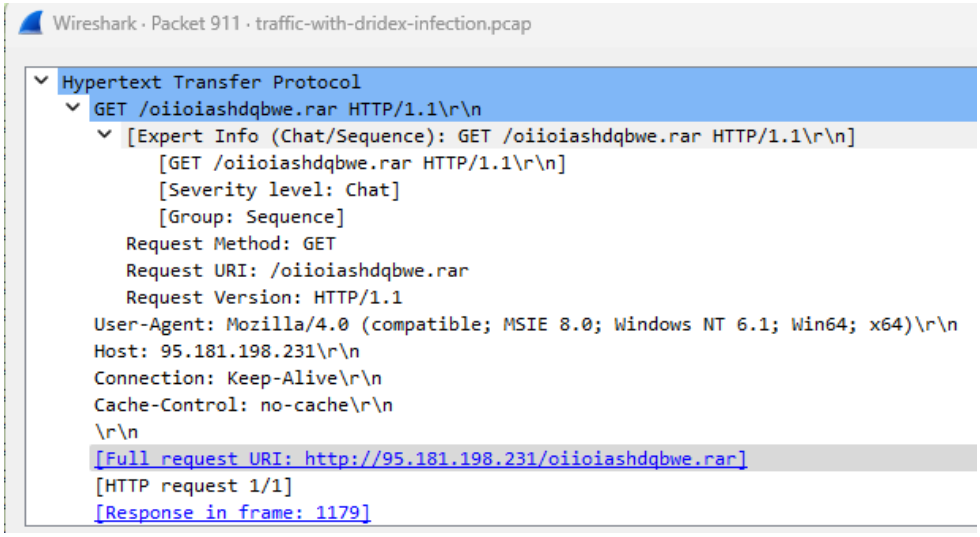
- Reviewing the filtered view further shows only two more HTTP GET requests to /images/…. which occur in packets 497 and 736
- This time when prior packets are reviewed, there are no DNS queries performed. This means that no recursive check was required because the requested domain was already stored in the users browser cache
    - Based on the first request, and this finding we can confidently say cochrimato.com is the desired domain

## The SOC Team found Dridex, a follow-up malware from Ursnif infection, to be the culprit. The customer who sent her the macro file is compromised. What's the full URL ending in .rar where Ursnif retrieves the follow-up malware from?

- Answer: http://95.181.198.231/oiioiashdqbwe.rar



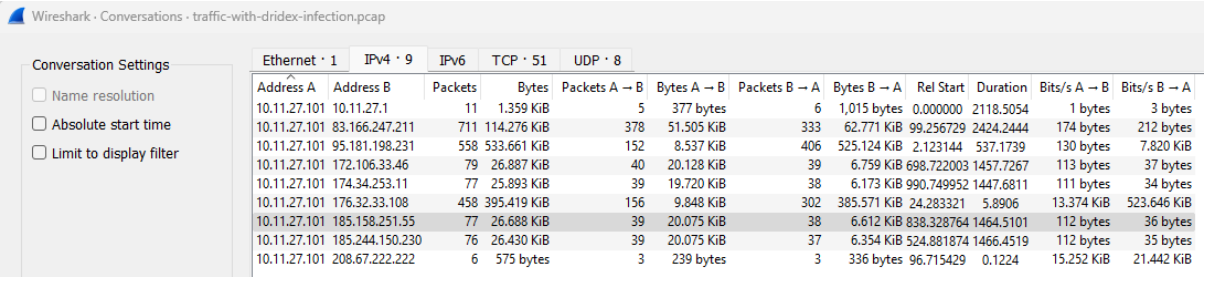

- Based on the question saying "… retrieves the follow-up malware from", this leads me to believe that the initial compromise malware will attempt to download another malware script ending in .rar
    - Meaning the pcap should be reviewed for another HTTP GET request, with a file (of nknown name) ending in .rar shown in the Info column of the packet view
- Keeping the pcap filtered to packets originating from the victim IP address, we start reviewing for HTTP GET requests matching the criteria stated above. This reveals packet 911 which requests /oiioiashdqbwe.rar
    - As there are no other packets or requests matching this criteria, we proceed with this file being the Dridex malware script/sample




- Double clicking on packet 911 opens a more detailed view of the packet
- Expand the HTTP dropdown and reviewing the contents reveals the full URI request to be http://95.181.198.231/oiioiashdqbwe.rar

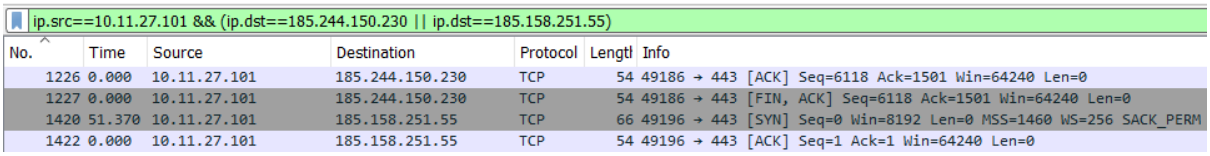
## What is the Dridex post-infection traffic IP addresses beginning with 185.?

- Answer: 185.244.150.230




| Address A | Address B | Packets | Bytes | Packets A → B | Bytes A → B | Packets B → A | Bytes B → A | Rel Start | Duration | Bits/s A → B | Bits/s B → A |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 10.11.27.101 | 10.11.27.1 | 11 | 1.359 KiB | 5 | 377 bytes | 6 | 1,015 bytes | 0.000000 | 2118.5054 | 1 bytes | 3 bytes |
| 10.11.27.101 | 83.166.247.211 | 711 | 114.276 KiB | 378 | 51.505 KiB | 333 | 62.771 KiB | 99.256729 | 2424.2444 | 174 bytes | 212 bytes |
| 10.11.27.101 | 95.181.198.231 | 558 | 533.661 KiB | 152 | 8.537 KiB | 406 | 525.124 KiB | 2.123144 | 537.1739 | 130 bytes | 7.820 KiB |
| 10.11.27.101 | 172.106.33.46 | 79 | 26.887 KiB | 40 | 20.128 KiB | 39 | 6.759 KiB | 698.722003 | 1457.7267 | 113 bytes | 37 bytes |
| 10.11.27.101 | 174.34.253.11 | 77 | 25.893 KiB | 39 | 19.720 KiB | 38 | 6.173 KiB | 990.749952 | 1447.6811 | 111 bytes | 34 bytes |
| 10.11.27.101 | 176.32.33.108 | 458 | 395.419 KiB | 156 | 9.848 KiB | 302 | 385.571 KiB | 24.283321 | 5.8906 | 13.374 KiB | 523.646 KiB |
| 10.11.27.101 | 185.158.251.55 | 77 | 26.688 KiB | 39 | 20.075 KiB | 38 | 6.612 KiB | 838.328764 | 1464.5101 | 112 bytes | 36 bytes |
| 10.11.27.101 | 185.244.150.230 | 76 | 26.430 KiB | 39 | 20.075 KiB | 37 | 6.354 KiB | 524.881874 | 1466.4519 | 112 bytes | 35 bytes |
| 10.11.27.101 | 208.67.222.222 | 6 | 575 bytes | 3 | 239 bytes | 3 | 336 bytes | 96.715429 | 0.1224 | 15.252 KiB | 21.442 KiB |

- Clearing all filters and opening Statistics > Conversations > IPv4 shows two IP addresses starting with 185. which are 185.158.251.55 and 185.244.150.230
    - As both IP addresses appear to have a similar amount of traffic to/from them there is nothing to distinguish which is the suspicious IP address at this moment




| No. | Time | Source | Destination | Protocol | Lengtl | Info |
|---|---|---|---|---|---|---|
| 1226 | 0.000 | 10.11.27.101 | 185.244.150.230 | TCP | 54 | 49186 → 443 [ACK] Seq=6118 Ack=1501 Win=64240 Len=0 |
| 1227 | 0.000 | 10.11.27.101 | 185.244.150.230 | TCP | 54 | 49186 → 443 [FIN, ACK] Seq=6118 Ack=1501 Win=64240 Len=0 |
| 1420 | 51.370 | 10.11.27.101 | 185.158.251.55 | TCP | 66 | 49196 → 443 [SYN] Seq=0 Win=8192 Len=0 MSS=1460 WS=256 SACK_PERM |
| 1422 | 0.000 | 10.11.27.101 | 185.158.251.55 | TCP | 54 | 49196 → 443 [ACK] Seq=1 Ack=1 Win=64240 Len=0 |

- Given the Conversations window shows the traffic originates from the victim IP for both the suspicious IP addresses, we can apply a combined filter for further investigation
    - This shows both IP addresses performing TCP and TLSv1.2 handshakes, but no further information to decipher which of the two is the one associated with Dridex
- Investigating both HTTPS certificates does
- I used Google to research for suspicious IP addresses related to Dridex malware and according to this link (https://www.cisa.gov/news-events/cybersecurity-advisories/aa19-339a), none of the known malicious IP addresses on this resource start with 185.X.X.X
    - This either means that BTLO changed the IP address so as to not use a real-world IP (very likely), or the above link is incorrect (very unlikely in my opinion)

| | | | | | | |
|---|---|---|---|---|---|---|
| 1214 | 0.061 | 10.11.27.101 | 185.244.150.230 | TLSv1.2 | 219 | Application Data |
| 1221 | 0.000 | 10.11.27.101 | 185.244.150.230 | TLSv1.2 | 1135 | Application Data |
| 1431 | 0.003 | 10.11.27.101 | 185.158.251.55 | TLSv1.2 | 219 | Application Data |
| 1439 | 0.000 | 10.11.27.101 | 185.158.251.55 | TLSv1.2 | 1135 | Application Data |
| 1566 | 0.001 | 10.11.27.101 | 185.244.150.230 | TLSv1.2 | 219 | Application Data |
| 1573 | 0.000 | 10.11.27.101 | 185.244.150.230 | TLSv1.2 | 1135 | Application Data |
| 1696 | 0.001 | 10.11.27.101 | 185.158.251.55 | TLSv1.2 | 219 | Application Data |
| 1703 | 0.000 | 10.11.27.101 | 185.158.251.55 | TLSv1.2 | 1135 | Application Data |
| 1839 | 0.003 | 10.11.27.101 | 185.244.150.230 | TLSv1.2 | 219 | Application Data |
| 1846 | 0.000 | 10.11.27.101 | 185.244.150.230 | TLSv1.2 | 1135 | Application Data |
| 1940 | 0.002 | 10.11.27.101 | 185.158.251.55 | TLSv1.2 | 219 | Application Data |
| 1947 | 0.000 | 10.11.27.101 | 185.158.251.55 | TLSv1.2 | 1135 | Application Data |

- At this stage, my last resort was to review when data was sent after TLSv1.2 handshakes had been established. This revealed that there were 5 packets of encrypted data sent to 185.158.251.55 and 6 sent to 185.244.150.230.
    - Based on this finding, I tentatively deduced that 185.244.150.230 is the suspicious IP address associated with Dridex.
- NOTE as of 14/9/2023:
    - Although I did find the correct IP address to complete this challenge, I am not 100% happy with my justification for this task

- I have logged a support ticket with BTLO for further clarification as to why 185.244.150.230 is correct IP address and will update this write-up once I have clarification