

Malware Command and Control Channels

- a journey into Darkness -

By Brad Woodberg

- Emerging Threats Product Manager / Proofpoint



Agenda

- C2 Intro and Background (7 mins)
- Modern C2 Techniques (6 mins)
- Case Studies (15 mins)
- Predictions for C2 (5 mins)
- Defense (10 mins)
- Wrap Up (2 mins)

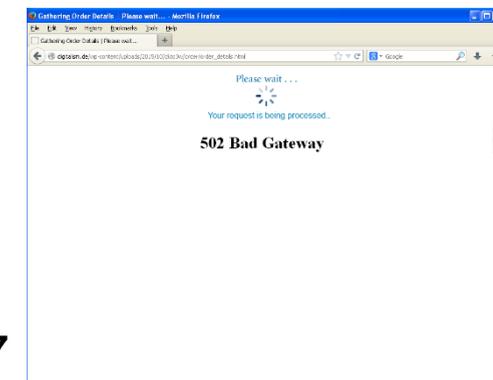
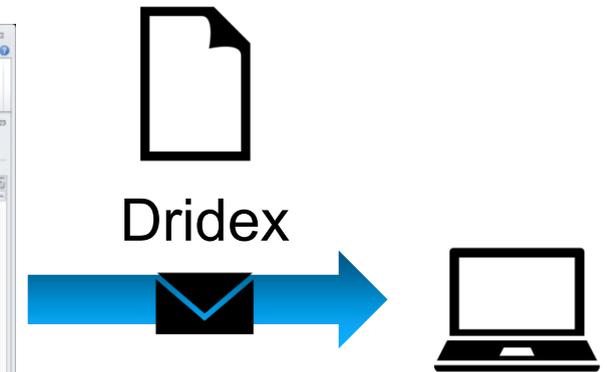
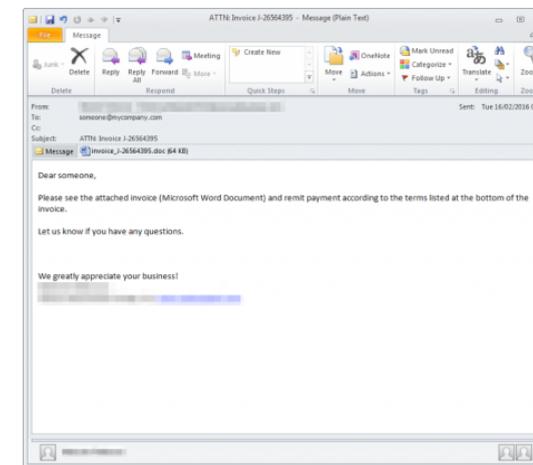
Why Command & Control?

- Vulnerabilities, Exploits, and Malware grab the headlines and analyst focus
- While very interesting, it is also very noisy, many exploits fail, very FP prone.
- If you can effectively detect C2 activity, you have a high fidelity indicator that an asset is actually compromised.
- With C2, the tables are turned on attackers, they go on defense, and we go on offense.

T-0: Initial Infection

- Modern malware is delivered in one of two ways:
 - Executable Content: Binary executables, embedded executable content like macros typically through web or email channels on the network.
 - Exploit Driven: An exploit against a software vulnerability such as those against Flash, PDF, Java, Office Docs, Browsers, and other network enabled applications.

- Regardless of how modern malware compromises a system, it is rarely autonomous.



Angler EK

T-1: Rough Landings

Initial malware execution may occur under non-ideal scenarios:

- Malware may land on a non-target asset
- Malware may not have sufficient privileges when it executes
- Malware may be delivered in pieces to evade detection / fit into buffers
- Malware may require payload before it is malicious (e.g. TinyLoader)
- Malware may require coordination with C2 for operating instructions before it takes action (e.g. Crypto Ransomware waiting to receive a key)

Enter Command and Control

T-2: Escalation

- Complete malware breach by acquiring additional executables, payloads, and configurations.
 - May be as simple as a word doc downloading an EXE (e.g. Dridex),
 - Or as complex as a dropper downloading an entirely new malware (e.g. Tinyloader / AbaddonPoS)
- Escalation stage is often carried out by contacting C2 Infrastructure
- This communication often leverages different infrastructure, protocols, and methods than the initial infection.
 - Often because infection infrastructure is rented, and C2 is managed by a different actor.

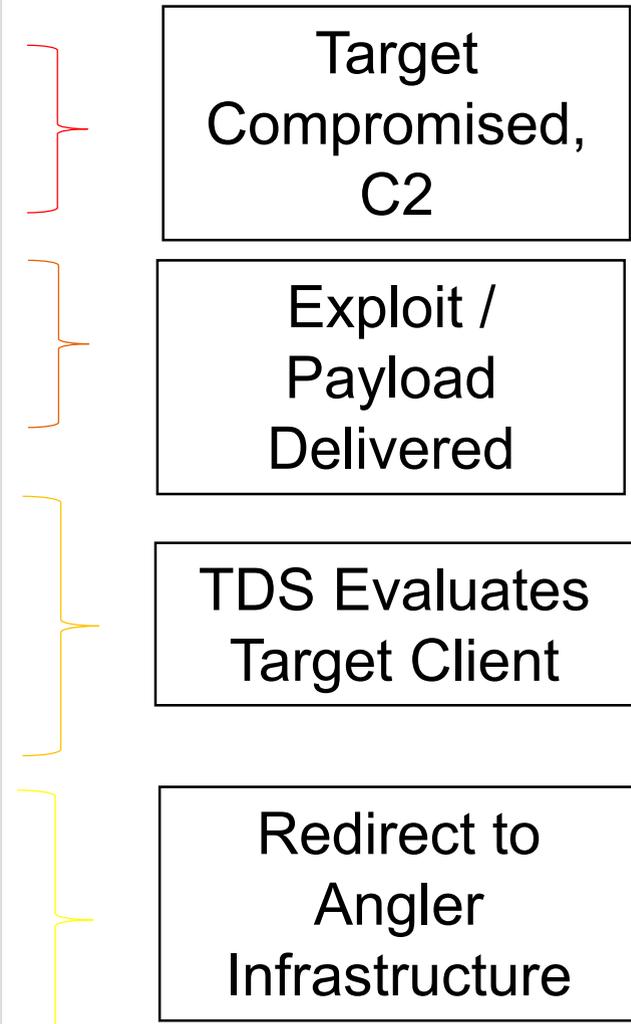
Initial Infection in Action: Angler Exploit Kit



Sample: 26907326de17c8c3f17c13bf32f61810

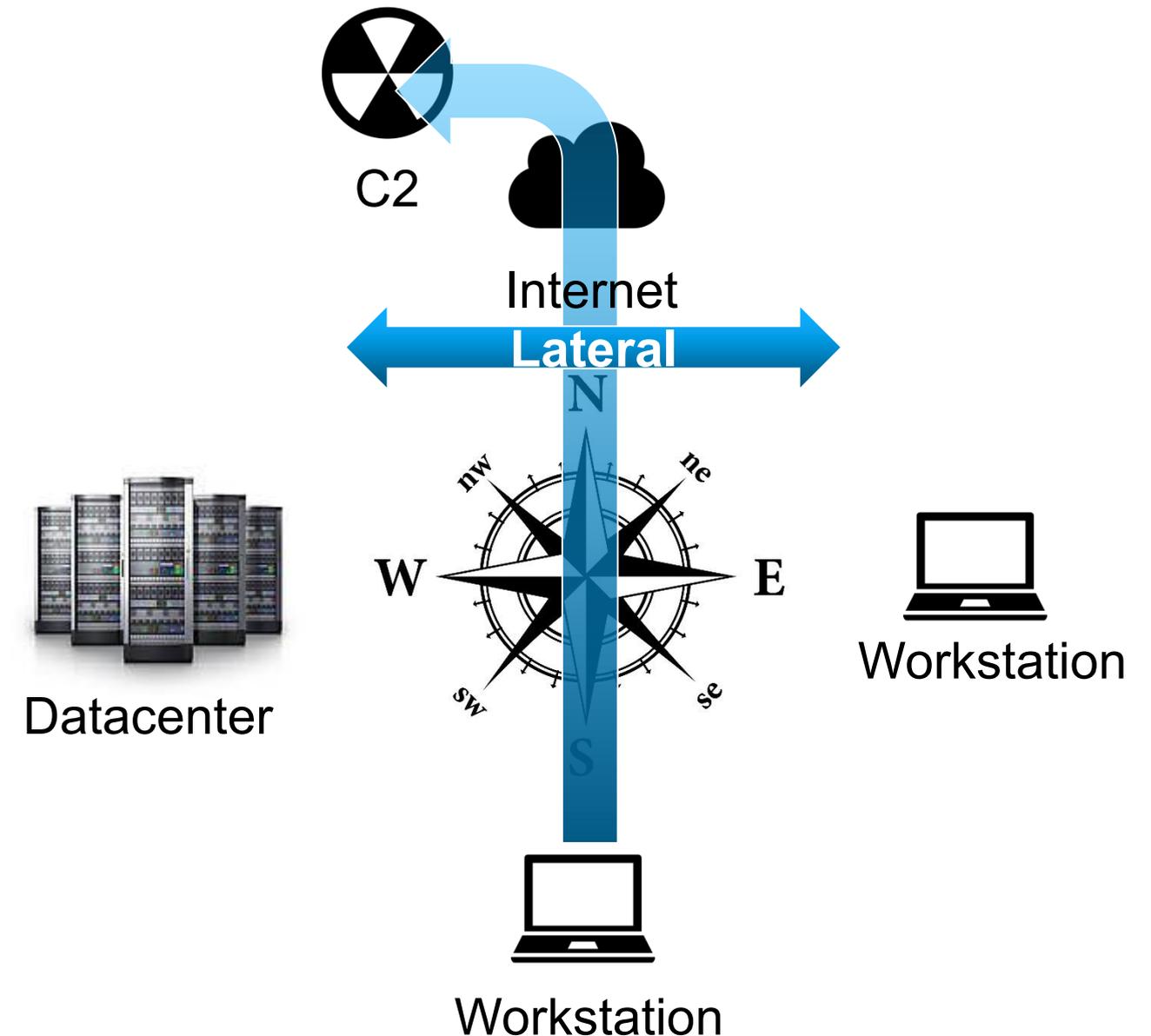
Alerts **Connections** DNS HTTP

Date	Sid	Signature	Rev	SrcIP	SrcPort	DstIP	DstPort
2016-05-17	2819805	ETPRO TROJAN CryptXXX CnC Beacon	3	private	49198	144.76.82.19	443
2016-05-17	2819805	ETPRO TROJAN CryptXXX CnC Beacon	3	private	49197	144.76.82.19	443
2016-05-17	2820097	ETPRO DELETED CryptXXX 2.06 Checkin	1	private	49197	144.76.82.19	443
2016-05-17	2816933	ETPRO CURRENT_EVENTS Angler EK Apr 07 2016	2	5.39.35.232	80	private	49193
2016-05-17	2811284	ETPRO CURRENT_EVENTS Angler or Nuclear EK Flash Exploit M2	2	5.39.35.232	80	private	49185
2016-05-17	2820164	ETPRO CURRENT_EVENTS Angler EK Payload May 10 2016 M2 T1	2	5.39.35.232	80	private	49185
2016-05-17	2811284	ETPRO CURRENT_EVENTS Angler or Nuclear EK Flash Exploit M2	2	5.39.35.232	80	private	49183
2016-05-17	2816933	ETPRO CURRENT_EVENTS Angler EK Apr 07 2016	2	5.39.35.232	80	private	49185
2016-05-17	2816933	ETPRO CURRENT_EVENTS Angler EK Apr 07 2016	2	5.39.35.232	80	private	49183
2016-05-17	2014726	ET POLICY Outdated Windows Flash Version IE	82	private	49183	5.39.35.232	80
2016-05-17	2816933	ETPRO CURRENT_EVENTS Angler EK Apr 07 2016	2	5.39.35.232	80	private	49183
2016-05-17	2816941	ETPRO CURRENT_EVENTS Angler EK Flash Exploit URI Struct Apr 07 IE	3	private	49183	5.39.35.232	80
2016-05-17	2815888	ETPRO CURRENT_EVENTS Possible Angler EK Landing Jan 21 M3	3	5.39.35.232	80	private	49178
2016-05-17	2816511	ETPRO CURRENT_EVENTS Angler EK Landing Mar 02 2016 M1 T1	2	5.39.35.232	80	private	49178
2016-05-17	2816932	ETPRO CURRENT_EVENTS Angler EK Landing with URI Primer Apr 06	2	5.39.35.232	80	private	49178
2016-05-17	2816933	ETPRO CURRENT_EVENTS Angler EK Apr 07 2016	2	5.39.35.232	80	private	49178
2016-05-17	2022772	ET CURRENT_EVENTS Evil Redirector Leading to EK Apr 28 2016	3	72.167.3.128	80	private	49163



Lateral Infection vs. C2

- Lateral Infection is not the same as C2!
- Lateral Infection focuses on Three Phases:
 - Introsection: Local device scanning
 - Network Scanning: mapping the network for potential targets and pivot points.
 - Exploit and Spread: Compromise other assets.
 - LI typically involves using native networking protocols to scan and spread within an organization (e.g. Locky using SMB to infect file shares)
- Lateral Infection is typically East / West by definition vs. North / South



Lateral Infection vs. C2 Continued

- > C2 is typically North / South
- > C2 will be less likely to be native enterprise networking protocols (e.g. AD protocols) and instead HTTP/SSL, custom application stacks, or outright custom channels such as encrypted channels.
- > C2 is often more evasive than LI
 - This is primarily because with C2 the attacker controls both sides of the communication, where with LI they only control the client!

Exfiltration

- This phase is where the malware delivers on its intended purpose
- Exfiltrated data often includes stealing intellectual property, exposing attributes of a target network, or larger escalation of an attack.
- May or may not leverage the standard C2 infrastructure including control channels, C2 servers &c.
- May be possible to fingerprint activity heuristically

The screenshot displays a network traffic analysis tool interface. The top section shows a list of network packets with columns for No., Time, Source, Destination, Protocol, Length, and Info. The bottom section shows a detailed view of a specific request and response, including headers and body content.

No.	Time	Source	Destination	Protocol	Length	Info
53	89.788441	192.168.78.25	198.105.254.11	HTTP	772	POST /upload/_dispatch.php HTTP/1.1 (application/x-www-form-urlencoded)
55	89.848311	198.105.254.11	192.168.78.25	HTTP	494	HTTP/1.1 301 Moved Permanently (text/html)
66	91.203880	192.168.78.25	198.105.254.11	HTTP	570	POST /upload/_dispatch.php HTTP/1.1
70	91.261809	198.105.254.11	192.168.78.25	HTTP	494	HTTP/1.1 301 Moved Permanently (text/html)
80	92.378867	192.168.78.25	192.42.116.41	HTTP	572	POST /upload/_dispatch.php HTTP/1.1
84	92.476604	192.42.116.41	192.168.78.25	HTTP	206	HTTP/1.1 200 OK
88	92.536732	192.168.78.25	198.105.254.11	HTTP	572	POST /upload/_dispatch.php HTTP/1.1
92	92.595736	198.105.254.11	192.168.78.25	HTTP	494	HTTP/1.1 301 Moved Permanently (text/html)
101	93.691644	192.168.78.25	198.105.254.11	HTTP	570	POST /upload/_dispatch.php HTTP/1.1
105	93.752331	198.105.254.11	192.168.78.25	HTTP	494	HTTP/1.1 301 Moved Permanently (text/html)
118	94.837520	192.168.78.25	198.105.254.11	HTTP	572	POST /upload/_dispatch.php HTTP/1.1
122	94.893161	198.105.254.11	192.168.78.25	HTTP	494	HTTP/1.1 301 Moved Permanently (text/html)
130	95.972402	192.168.78.25	198.105.254.11	HTTP	570	POST /upload/_dispatch.php HTTP/1.1
134	96.029925	198.105.254.11	192.168.78.25	HTTP	494	HTTP/1.1 301 Moved Permanently (text/html)
144	97.900159	192.168.78.25	5.196.70.240	HTTP	570	POST /upload/_dispatch.php HTTP/1.1
148	98.321007	5.196.70.240	192.168.78.25	HTTP	671	HTTP/1.1 200 OK (application/octet-stream)

User-Agent: Mozilla/4.0 (compatible; MSIE 7.0; Windows NT 6.1; Trident/5.0; SLCC2; .NET CLR 2.0.50727; .NET CLR 3.5.30729; .NET CLR 3.0.30729; Media Center PC 6.0; .NET4.0C; .NET4.0E)\r\n
Host: gccxquuylioxoip.pw\r\n
Content-Length: 718\r\n
Connection: Keep-Alive\r\n
\r\n
[Full request URI: http://gccxquuylioxoip.pw/upload/_dispatch.php]
[HTTP request 1/1]
[Response in frame: 55]
HTML Form URL Encoded: application/x-www-form-urlencoded
Form item: "dyws" = "r...x...j...u"
Form item: "YjKRb" = "q5JRYu!!L...tj...j3"
Form item: "jMsRnD" = "...LI...CTW...=1..."
Form item: "BzBDFQ" = "0...R...T...fC...-V...@b..."
Form item: "uDRZrHs" = "...4...i...]"x'sJ0rJ9"
Form item: "dJTn" = "...;T6b..._...l&e..."
Form item: "IUMvMyc" = "...G...q...m...up4...9...T...i6@..."
Form item: "DcehnCj" = "h...M...i.../...|4...-...|..."
Form item: "XSull" = "...|d...-...5<...w"

0000 08 00 27 3b 7d 34 a0 00 27 3e 08 8e 08 00 45 00 ...;4.. '>...E.
0010 02 f6 0a d4 40 00 80 06 19 f7 c0 a8 4e 19 c6 69 ...@... ..N..i
0020 fe 0b c0 0e 00 50 79 73 8e 00 04 c8 a4 df 50 18 ...PysP.
0030 fa f0 3e 11 00 00 64 79 77 73 3d 25 31 36 25 45 ...>...dy ws=%16E
0040 43 25 45 33 25 31 44 25 32 32 25 32 41 25 38 36 C%3%10% 22%2A%80

Targeted vs. Crimeware

- At a high level we can categorize malware into two families, Crimeware and Targeted.
 - Crimeware: This is malware that is often general purpose and widely distributed. Often as part of exploit kits and mass mailing campaigns.
 - Targeted: This is malware that is custom built to target individual organizations or a small subset of targets often within a specific vertical.
- Under Targeted Malware there is a third category which is Targeted Espionage which is typically much more advanced.

Crimeware vs. Targeted

Crimeware:

- General Purpose
- Widely distributed
- Go to greater lengths to evade detection from a protocol perspective
- Yet quite chatty on C2 channels

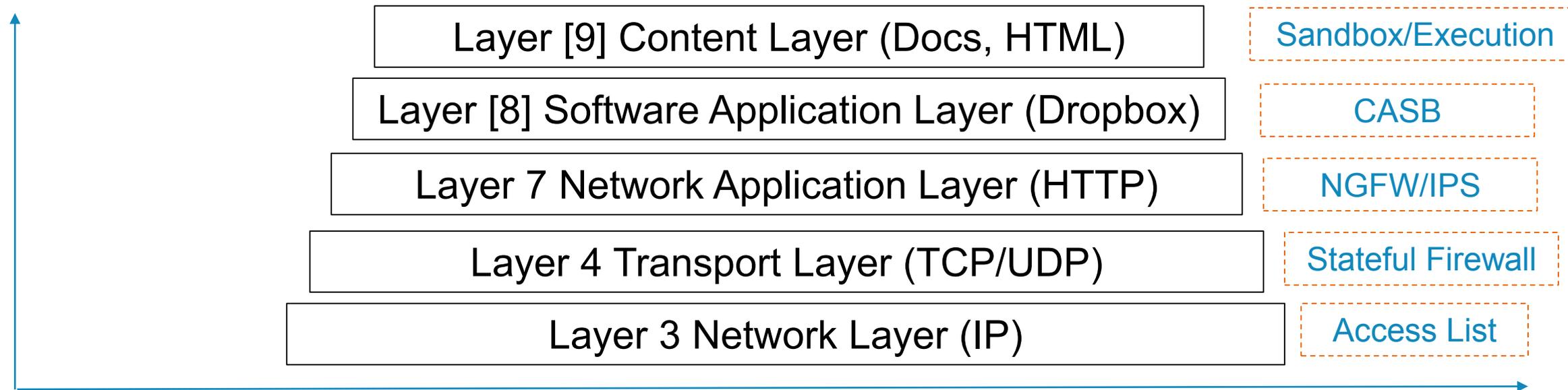
Targeted:

- Highly selective victims
- Will be custom built to navigate individual networks, common platforms.
- Often does not go to great lengths from an obfuscation perspective

Targeted Espionage:

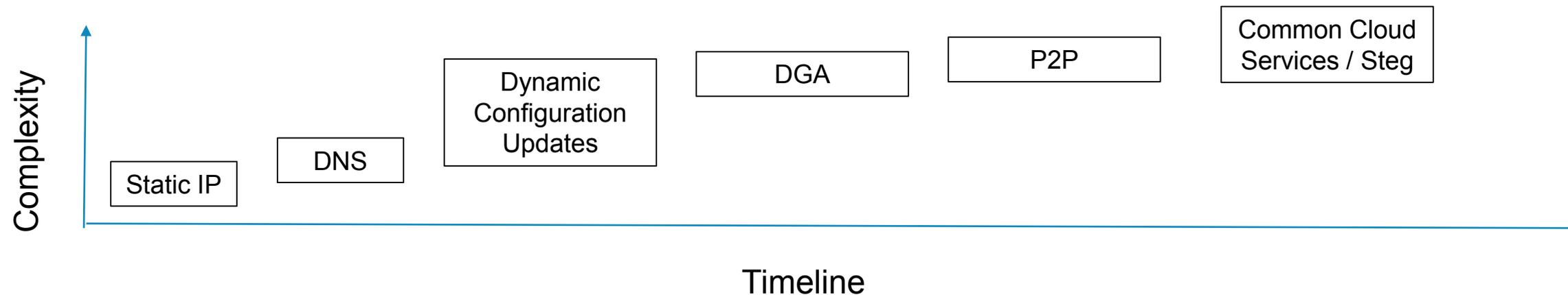
- Most exotic form of malware
- Far more sophisticated than traditional targeted.
- May lack network based C2 channels altogether.
- May leverage insiders as well as covert HW to bridge air gaps.

Cat and Mouse



- Enter the Stateful Firewall which leveraged Layer 7 payload inspection (similar to IPS) to identify applications
 - Early malware attacks used fixed non-standard ports to communicate e.g. Back Orifice (1998)
- Malware noted that keeping explicit strings in the payload would be easy to identify (e.g. GhostRat). The same early malware heavily leveraged IP options like Bits for a /TUPLE/SCYPIE structure
 - Early forward proxy heavily leveraged IP options like Bits for a /TUPLE/SCYPIE structure
- To evade NGFWs, malware shifted to leverage steganographic techniques to hide data in legitimate files, such as
 - NGFWs and other deep inspection tools were shifted to leverage steganographic techniques to hide data in legitimate files, such as
- Finally, malware just took advantage of the fact that stateful firewalls don't look beyond the L4 header to allow traffic to communicate out of the network.
 - At the same time, the malware just took advantage of the fact that stateful firewalls don't look beyond the L4 header to allow traffic to communicate out of the network.
- In addition to the advanced obfuscation, malware has gone to great lengths to hide itself in legitimate, cloud applications.

C2 Hosting



- Early days C2 infrastructure was very fixed. Similar to traditional computing, it was physical machines in data centers with static IP's.
- While DNS was prominent, domain names for malware would not change very quickly.
- Configuration Updates via CNC
- This weak link made for a great target for vendors providing defense mechanisms. So malware evolved as well to domain generation algorithms (DGA's) which could quickly cycle through generated domain names to eliminate single points of failure. E.g. Conficker
- The issue with DGA's is that the algorithm can be reverse engineered, and it still relies on DNS. Enter P2P Mechanisms like GameOver-Zeus
- To offset the potential disruptions for DGA's, malware started leveraging common cloud services which enterprises are adverse to blocking as they may serve a business function.

C2 - Counter Defense Techniques

- Attackers think economically, want their malware to last as long as possible thus bringing the most ROI.
- Botnet authors utilize several counter detection techniques to ensure the viability of their malware.
 - Filter who can connect (e.g. IP filtering to eliminate non-targets, researchers and sandboxing tools.)
 - Secret Handshakes: E.g. leverage custom TCP stacks or special low level handshakes that only illicit responses if correct handshake is used (e.g. Poison Ivy)
 - Encryption: Predefined SSL Certificates embedded in malware for authenticating client/servers
 - Steganography: Hiding in plain sight, exceptionally difficult to detect, looks like standard legitimate apps and traffic.

Case Studies

- Now that we've covered the background and evolution, let's take a look at actual malware C2 channels to reinforce our examples.
- Note that there are often a great many variants for each malware and some leverage different communication than the mainstream samples which we will cover.

Gh0stRAT



- Basic C2 Protocol
- Common strains support a basic non-encoded string in the PCAP.
- ‘Gh0st’ string in initial payload to identify malware
- Non-Standard Port easily filterable

The screenshot shows a Wireshark interface with a packet list table and a packet details pane. The packet list table shows several TCP packets between source IP 192.168.103.10 and destination IP 60.251.73.126. Packet 12 is selected, showing details for Ethernet II, Internet Protocol Version 4, and Transmission Control Protocol (Seq=1, Ack=1, Len=195). The packet bytes pane shows the raw data in hex and ASCII, with the ASCII portion containing the string "b.o._G.. /..g..E..".

No.	Time	Source	Destination	Protocol	Length	Info
9	9.850912	192.168.103.10	60.251.73.126	TCP	62	1026 → 3002 [SYN] Seq=0 Win=16384 Len=0 MSS=1460 SACK_PERM=1
10	10.039054	60.251.73.126	192.168.103.10	TCP	62	3002 → 1026 [SYN, ACK] Seq=0 Ack=1 Win=8192 Len=0 MSS=1460 SACK_PERM=1
11	10.040020	192.168.103.10	60.251.73.126	TCP	60	1026 → 3002 [ACK] Seq=1 Ack=1 Win=17520 Len=0
12	10.045809	192.168.103.10	60.251.73.126	TCP	249	1026 → 3002 [PSH, ACK] Seq=1 Ack=1 Win=17520 Len=195
13	10.234160	60.251.73.126	192.168.103.10	TCP	76	3002 → 1026 [PSH, ACK] Seq=1 Ack=196 Win=64240 Len=22
14	10.401554	192.168.103.10	60.251.73.126	TCP	60	1026 → 3002 [ACK] Seq=196 Ack=23 Win=17498 Len=0

Frame 12: 249 bytes on wire (1992 bits), 249 bytes captured (1992 bits)
Ethernet II, Src: AsustekC_8f:a0:67 (00:11:2f:8f:a0:67), Dst: 62:ad:6f:ef:5f:47 (62:ad:6f:ef:5f:47)
Internet Protocol Version 4, Src: 192.168.103.10, Dst: 60.251.73.126
Transmission Control Protocol, Src Port: 1026 (1026), Dst Port: 3002 (3002), Seq: 1, Ack: 1, Len: 195
Source Port: 1026
Destination Port: 3002
[Stream index: 0]

```
0000 62 ad 6f ef 5f 47 00 11 2f 8f a0 67 08 00 45 00  b.o._G.. /..g..E.  
0010 00 eb 00 09 40 00 80 06 4b d8 c0 a8 67 0a 3c fb  ....@... K...g.<.  
0020 49 7e 04 02 0b ba b0 d0 5d 2e f8 11 b5 b1 50 18  I~..... ].....P.  
0030 44 70 6c 59 00 00 c3 00 00 00 f8 00 00 00 47 68  DpLY.... ..Gh  
0040 30 73 74 78 9c 4b 63 60 60 98 03 c4 ac 40 cc 08  0stx.Kc` `...@..  
0050 c4 1a 5c 0c 0c 4c 40 3a 38 b5 a8 2c 33 39 55 21  ..\..L@: 8...,39U!  
0060 20 31 39 5b c1 98 a1 5e 02 28 76 86 9d 81 61 87  19[...^(v...a.  
0070 87 10 03 08 e4 b3 31 30 d8 b8 09 31 48 30 48 31  .....10 ...1H0H1  
0080 58 78 03 c5 5e 4e a8 d9 f1 6a 62 cd 7f 20 d8 0a  Xx..^N.. .jb.. ..  
0090 a4 23 7f 4f aa 89 01 62 10 1f a4 3e 63 7d 6b cd  .#.0...b ...>c}k.  
00a0 17 a0 5e 0f a0 da e6 75 ad 35 20 be 05 10 7f 78  ..^....u .5 ....x  
00b0 25 c4 70 e0 b3 10 43 dd 15 07 06 a0 91 0c 05 a7  %.p...C. ....  
00c0 8e 82 31 33 48 13 23 23 c3 6f 0e 06 06 43 20 b3  ..13H.## .o...C .  
00d0 14 28 06 03 a7 96 6f b8 03 32 f4 c0 8a 74 ae 82  .(....o. .2...t..  
00e0 d2 a4 9c cc 64 dd d4 44 0b 63 33 f3 54 73 06 62  ....d..D .c3.Ts.b  
00f0 40 35 d0 70 00 e4 c4 40 55 @5.p...@ U
```

PoisonIvy

- Unknown Encrypted, 256 Byte handshake
- Does not contain explicit strings in handshake which are easy to key on.
- Available since 2005, still very popular and little changed despite being in the wild so long.
- 256 Byte Handshake is exchanged in a CHAP like sequence. Client sends a hello which allows the server to prevent it from communicating with an unknown client.
- The server will only accept the client communication if it has been encrypted with the right password.

No.	Time	Source	Destination	Protocol	Length	Info
6	0.029001	172.16.3.46	172.16.1.1	TCP	310	1037 → 3460 [PSH, ACK] Seq=1 Ack=1 Win=65535 Len=256
7	0.029028	172.16.1.1	172.16.3.46	TCP	54	3460 → 1037 [ACK] Seq=1 Ack=257 Win=6432 Len=0
8	60.392080	172.16.1.1	172.16.3.46	TCP	54	3460 → 1037 [FIN, ACK] Seq=1 Ack=257 Win=6432 Len=0
9	60.392213	172.16.3.46	172.16.1.1	TCP	60	1037 → 3460 [ACK] Seq=257 Ack=2 Win=65535 Len=0
10	60.392321	172.16.3.46	172.16.1.1	TCP	60	1037 → 3460 [FIN, ACK] Seq=257 Ack=2 Win=65535 Len=0
11	60.392340	172.16.1.1	172.16.3.46	TCP	54	3460 → 1037 [ACK] Seq=2 Ack=258 Win=6432 Len=0
14	90.469389	172.16.3.46	185.32.221.46	TCP	62	1038 → 3460 [SYN] Seq=0 Win=65535 Len=0 MSS=1460 SACK_PERM=1
15	90.643309	185.32.221.46	172.16.3.46	TCP	54	3460 → 1038 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
16	91.110604	172.16.3.46	185.32.221.46	TCP	62	[TCP Spurious Retransmission] 1038 → 3460 [SYN] Seq=0 Win=65535 Len=0 MSS=1460 SACK_PERM=1
17	91.284695	185.32.221.46	172.16.3.46	TCP	54	3460 → 1038 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
18	91.766846	172.16.3.46	185.32.221.46	TCP	62	[TCP Spurious Retransmission] 1038 → 3460 [SYN] Seq=0 Win=65535 Len=0 MSS=1460 SACK_PERM=1
19	91.941027	185.32.221.46	172.16.3.46	TCP	54	3460 → 1038 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
22	121.939892	172.16.3.46	172.16.1.1	TCP	62	1039 → 3460 [SYN] Seq=0 Win=65535 Len=0 MSS=1460 SACK_PERM=1
23	121.939982	172.16.1.1	172.16.3.46	TCP	62	3460 → 1039 [SYN, ACK] Seq=0 Ack=1 Win=5840 Len=0 MSS=1460 SACK_PERM=1

▸ Frame 6: 310 bytes on wire (2480 bits), 310 bytes captured (2480 bits)
▸ Ethernet II, Src: HewlettP_47:bd:24 (00:08:02:47:bd:24), Dst: Supermic_f8:de:82 (00:30:48:f8:de:82)
▸ Internet Protocol Version 4, Src: 172.16.3.46, Dst: 172.16.1.1
▸ Transmission Control Protocol, Src Port: 1037 (1037), Dst Port: 3460 (3460), Seq: 1, Ack: 1, Len: 256

▲ Data (256 bytes)
Data: de36a7able4b718367483754be28dc864ec357fc326e5e31...
[Length: 256]

```
0000 00 30 48 f8 de 82 00 08 02 47 bd 24 08 00 45 00 .0H.... .G.$..E.
0010 01 28 16 f8 40 00 80 06 86 88 ac 10 03 2e ac 10 .(.@... ..
0020 01 01 04 0d 0d 84 c8 54 12 40 e7 c4 b1 e4 50 18 .....T @....P.
0030 ff ff 14 81 00 00 de 36 a7 ab 1e 4b 71 83 67 48 .....6 ...Kq.gH
0040 37 54 be 28 dc 86 4e c3 57 fc 32 6e 5e 31 c7 d4 7T.(.N.W.2n^1..
0050 2a 6a b2 54 3f a4 04 04 83 32 85 c1 93 33 65 b5 *j.T?... .2...3e.
0060 9e 64 a8 09 ab 63 56 7f af 7a 1f ad b7 64 e8 31 .d...cV. .z...d.1
0070 fb 56 5d 6c d1 df f3 e0 8e bc 48 6f 89 a0 3a 50 .V]l.... ..Ho...P
0080 d8 60 a1 dd 74 be 5b 55 97 a1 b1 a2 aa 7a 50 ae .`.t.[U .....zP.
0090 15 91 33 3d 85 99 9d 5b a9 1c 74 5b a2 5d e7 be ..3=...[ ..t[.]..
00a0 a0 95 31 70 14 a4 df e3 93 e0 d5 7c 07 fe 0a 29 ..1p.... ...]...
00b0 ec 3f 54 80 2c a7 59 70 5d 4e fa 81 56 e5 b5 f6 .?T.,.Yp ]N..V...
00c0 ec d3 21 8f f0 0d 8d 50 c0 94 ad 82 c9 ff 12 d1 ..!....P .....
00d0 ff 4f f8 32 50 7b 56 9d cc 0f b1 a6 d2 47 3f db .0.2P{V. ....G?.
00e0 21 c3 68 42 37 b1 c1 b3 bf a0 92 80 d4 c7 99 2d !.hb7... ..
00f0 3a 9f 62 94 54 90 37 08 06 ed b8 9b d6 6f ce cb .:b.T.7. ....o..
0100 30 ec 81 0d 8a 47 e7 28 4a f3 1b 77 30 d0 36 e4 0....G.( ).w0.6.
0110 c5 99 6f 03 ca 7e 96 ac 95 9c 3a d9 44 07 d3 45 .o...~... :.D..E
0120 3d 1e 5c b6 e8 2c 32 4a f3 d4 5b 4a 09 07 4e 7e =.\.,2] ..[J..N~
0130 27 61 de df 20 e2 'a... .
```

NanoLocker

- Some malware leverage common network utilities and infrastructure to embed C2 functionality
- NanoLocker leverages ICMP to ping a hardcoded address 52.91.55.122 with an ICMP payload of the ransomware Bitcoin address. It will also send follow up payloads of the number of files encrypted on the system.

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	10.0.2.15	52.91.55.122	IPv4	74	ICMP (1)
2	0.079069	52.91.55.122	10.0.2.15	IPv4	74	ICMP (1)

▶ Frame 1: 74 bytes on wire (592 bits), 74 bytes captured (592 bits)
▶ Ethernet II, Src: CadmusCo_d1:0a:c1 (08:00:27:d1:0a:c1), Dst: RealtekU_12:35:02 (52:54:00:12:35:02)
▶ Internet Protocol Version 4, Src: 10.0.2.15, Dst: 52.91.55.122
▶ Data (40 bytes)
Data: 0800475c020004006162636465666768696a6b6c6d6e6f7071727374757677616263646566676869
[Length: 40]

```
0000 52 54 00 12 35 02 08 00 27 d1 0a c1 08 00 45 00  RT..5... '.....E.
0010 00 3c 00 4c 00 00 80 01 c2 91 0a 00 02 0f 34 5b  .<.L.... .....4[
0020 37 7a 08 00 47 5c 02 00 04 00 61 62 63 64 65 66  7z..G\.. .abcdef
0030 67 68 69 6a 6b 6c 6d 6e 6f 70 71 72 73 74 75 76  ghijklmn opqrstuv
0040 77 61 62 63 64 65 66 67 68 69                    wabcdefg hi
```

GameOver/Zeus

- GameOver / Zeus attempted to obfuscate its activities by leveraging P2P protocols to avoid single points of failure similar to how traditional P2P filesharing services work (loosely based on Kademlia DHT techniques)
- Zeus leveraged basic rolling XOR for packet payloads to make signature based IDS difficult. UDP Payloads
 - Emphasizes the point that often times the malware authors will just attempt to stay one step ahead of security solutions rather than implement the most state of the art attacks.

The image shows a Wireshark network traffic capture. The top pane displays a list of packets with columns for No., Time, Source, Destination, Protocol, Length, and Info. The bottom pane shows a detailed view of a selected packet (Frame 7), including the Ethernet II header, Internet Protocol Version 4 header, and User Datagram Protocol header. The payload is shown in hexadecimal and ASCII, demonstrating a rolling XOR obfuscation technique.

No.	Time	Source	Destination	Protocol	Length	Info
7	36.789199	192.168.4.5	79.46.183.210	RTP	114	PT=Unassigned, SSRC=0x2CF2453A, Seq=40604, Time=431093951, Mark[Malformed Packet]
8	51.814595	192.168.4.5	66.136.148.79	UDP	114	29579 → 19925 Len=72
10	66.814572	192.168.4.5	74.41.224.134	BT-uTP	114	uTorrent Transport Protocol Type: Unknown 69[Malformed Packet]
12	81.845902	192.168.4.5	31.135.144.242	UDP	114	29579 → 11571 Len=72
14	96.877045	192.168.4.5	178.59.240.41	SKYPE	114	NAT repeat Unk: 4
16	111.877009	192.168.4.5	82.107.220.127	SKYPE	114	NAT info Unk: 4
18	126.892576	192.168.4.5	117.194.224.243	UDP	114	29579 → 28318 Len=72
19	141.908162	192.168.4.5	46.36.116.1	RTP	114	PT=Unassigned, SSRC=0x468FB450, Seq=44206, Time=2280578286, Mark
20	156.923636	192.168.4.5	79.9.10.57	UDP	114	29579 → 23939 Len=72

Frame 7: 114 bytes on wire (912 bits), 114 bytes captured (912 bits)
Ethernet II, Src: CisTechn_10:20:04 (00:20:18:10:20:04), Dst: 00:ff:a1:00:06:50 (00:ff:a1:00:06:50)
Internet Protocol Version 4, Src: 192.168.4.5, Dst: 79.46.183.210
User Datagram Protocol, Src Port: 29579 (29579), Dst Port: 11134 (11134)

```
0000 00 ff a1 00 06 50 00 20 18 10 20 04 08 00 45 00  ....P.  . . . .E.
0010 00 64 00 7c 00 00 80 11 6e 5f c0 a8 04 05 4f 2e  .d.|.... n_....O.
0020 b7 d2 73 8b 2b 7e 00 50 59 50 9f 9e 9e 9c 19 b1  ..s.+~.P YP.....
0030 f8 bf 2c f2 45 3a b0 b7 4e 13 66 da 11 3c d3 4a  ., .E:.. N.f.<.J
0040 ca 61 3c bc 1c a8 f7 f7 73 24 26 6d 46 83 63 72  .a<..... s$&mF.cr
0050 9a ef e3 4e bc 7d d6 2a ba 92 51 4d 02 77 0b 0c  ...N.).* ..Qm.w..
0060 50 91 26 ff 10 90 54 59 cf 3e 3e 3e 3e 3e 3e 3e  P.&...TY .>>>>>>
0070 3e 3e                                     >>
```

Dridex using Pastebin as C2

- Virtually any cloud service can be used for C2. In this example Pastebin is leveraged.
- While sites like Pastebin might be simple to turn off, Twitter, Amazon, and Facebook may have legitimate business purposes.
- Malware may hide in comments, images, video and uploaded content.

Sample: ce181f45efb519504e54fed5daa45cc7

MD5 ce181f45efb519504e54fed5daa45cc7 SHA256 N/A
Submission Date 2015-08-11 17:38:02 File Size N/A
Type PCAP VirusTotal 17/57

Alerts Connections DNS HTTP

Date	Sid	Signature	Rev	SrcIP	SrcPort	DstIP	DstPort
2015-08-11	2021621	ET TROJAN Possible Dridex SSL Cert Aug 12 2015	6	94.23.110.45	443	private	49442
2015-08-11	2021621	ET TROJAN Possible Dridex SSL Cert Aug 12 2015	6	195.154.184.240	1443	private	49433
2015-08-11	2812390	ETPRO TROJAN Possible Dridex Exe Command in Pastebin Title	2	190.93.240.15	80	private	49432
2015-08-11	2812389	ETPRO TROJAN Possible Dridex Open Command in Pastebin Title	2	190.93.240.15	80	private	49432
2015-08-11	2014520	ET INFO EXE - Served Attached HTTP	6	185.14.29.178	80	private	49431
2015-08-11	2021078	ET INFO SUSPICIOUS Dotted Quad Host MZ Response	2	185.14.29.178	80	private	49431
2015-08-11	2014520	ET INFO EXE - Served Attached HTTP	6	185.14.29.178	80	private	49431
2015-08-11	2021078	ET INFO SUSPICIOUS Dotted Quad Host MZ Response	2	185.14.29.178	80	private	49431
2015-08-11	10000029	FILE ET magio PE32	2	185.14.29.178	80	private	49431
2015-08-11	2000419	ET POLICY PE EXE or DLL Windows file download	22	185.14.29.178	80	private	49431
2015-08-11	2021244	ET TROJAN Dridex Download June 10 2015	2	185.14.29.178	80	private	49431
2015-08-11	2812388	ETPRO TROJAN Possible Dridex 0 byte POST to Pastebin	3	private	49430	190.93.240.15	80

ToR as a C2 Channel

- After an initial infection, malware hops to TOR2Web a clientless TOR implementation for C2 Activity
- TOR allows botnet operators to evade communication snooping in intermediate systems.

Sample: eef89c15b2625a8614d8c898fb802e04

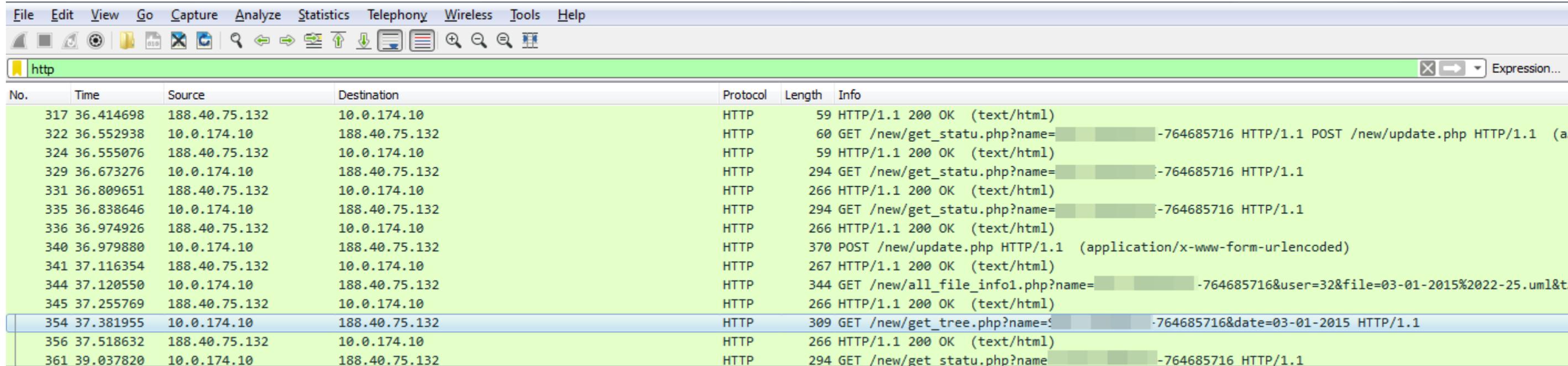
MD5 eef89c15b2625a8614d8c898fb802e04
Submission Date 2015-02-10 17:03:21
Type PE32 executable (GUI) Intel 80386, for MS Windows

SHA256 c026e9528b880d62e886c837494da9d6fc3ed90374f69c5496de63086eb9f575
File Size 48592
VirusTotal 47/54

Alerts Connections DNS HTTP

Date	Sid	Signature	Rev	SrcIP	SrcPort	DstIP	DstPort
2015-10-11	2018879	ET POLICY onion.cab tor2web .onion Proxy domain in SNI	1	private	49380	188.138.122.22	443
2015-10-11	2018876	ET POLICY onion.cab .onion Proxy DNS lookup	1	private	53212	8.8.8.8	53
2015-10-11	2020358	ET TROJAN Critroni Variant .onion Proxy Domain	1	private	53212	8.8.8.8	53
2015-10-11	2018876	ET POLICY onion.cab .onion Proxy DNS lookup	1	private	53212	8.8.8.8	53
2015-10-11	2020358	ET TROJAN Critroni Variant .onion Proxy Domain	1	private	53212	8.8.8.8	53
2015-10-11	2015576	ET POLICY DNS Query to .onion proxy Domain (tor2web)	6	private	62661	8.8.8.8	53
2015-10-11	2020358	ET TROJAN Critroni Variant .onion Proxy Domain	1	private	62661	8.8.8.8	53
2015-10-11	2015576	ET POLICY DNS Query to .onion proxy Domain (tor2web)	6	private	62661	8.8.8.8	53
2015-10-11	2020358	ET TROJAN Critroni Variant .onion Proxy Domain	1	private	62661	8.8.8.8	53
2015-10-11	2808413	ETPRO POLICY telize.com IP lookup	2	private	49386	46.19.37.108	80
2015-10-11	2019925	ET TROJAN Win32/Dalexix.A Possible SSL Cert (cargol.cat)	2	217.149.7.213	443	private	49354
2015-10-11	2019924	ET TROJAN Win32/Dalexix.A Possible SSL Cert (ppc.cba.pl)	2	85.17.73.180	443	private	49353

AridViper



The screenshot shows a network traffic analysis tool interface with a menu bar (File, Edit, View, Go, Capture, Analyze, Statistics, Telephony, Wireless, Tools, Help) and a toolbar. The main display area shows a list of HTTP traffic entries. The selected entry (No. 354) is highlighted in blue. The table columns are No., Time, Source, Destination, Protocol, Length, and Info.

No.	Time	Source	Destination	Protocol	Length	Info
317	36.414698	188.40.75.132	10.0.174.10	HTTP	59	HTTP/1.1 200 OK (text/html)
322	36.552938	10.0.174.10	188.40.75.132	HTTP	60	GET /new/get_statu.php?name=[REDACTED]-764685716 HTTP/1.1 POST /new/update.php HTTP/1.1 (a
324	36.555076	188.40.75.132	10.0.174.10	HTTP	59	HTTP/1.1 200 OK (text/html)
329	36.673276	10.0.174.10	188.40.75.132	HTTP	294	GET /new/get_statu.php?name=[REDACTED]-764685716 HTTP/1.1
331	36.809651	188.40.75.132	10.0.174.10	HTTP	266	HTTP/1.1 200 OK (text/html)
335	36.838646	10.0.174.10	188.40.75.132	HTTP	294	GET /new/get_statu.php?name=[REDACTED]-764685716 HTTP/1.1
336	36.974926	188.40.75.132	10.0.174.10	HTTP	266	HTTP/1.1 200 OK (text/html)
340	36.979880	10.0.174.10	188.40.75.132	HTTP	370	POST /new/update.php HTTP/1.1 (application/x-www-form-urlencoded)
341	37.116354	188.40.75.132	10.0.174.10	HTTP	267	HTTP/1.1 200 OK (text/html)
344	37.120550	10.0.174.10	188.40.75.132	HTTP	344	GET /new/all_file_info1.php?name=[REDACTED]-764685716&user=32&file=03-01-2015%2022-25.uml&t
345	37.255769	188.40.75.132	10.0.174.10	HTTP	266	HTTP/1.1 200 OK (text/html)
354	37.381955	10.0.174.10	188.40.75.132	HTTP	309	GET /new/get_tree.php?name=[REDACTED]-764685716&date=03-01-2015 HTTP/1.1
356	37.518632	188.40.75.132	10.0.174.10	HTTP	266	HTTP/1.1 200 OK (text/html)
361	39.037820	10.0.174.10	188.40.75.132	HTTP	294	GET /new/get_statu.php?name=[REDACTED]-764685716 HTTP/1.1

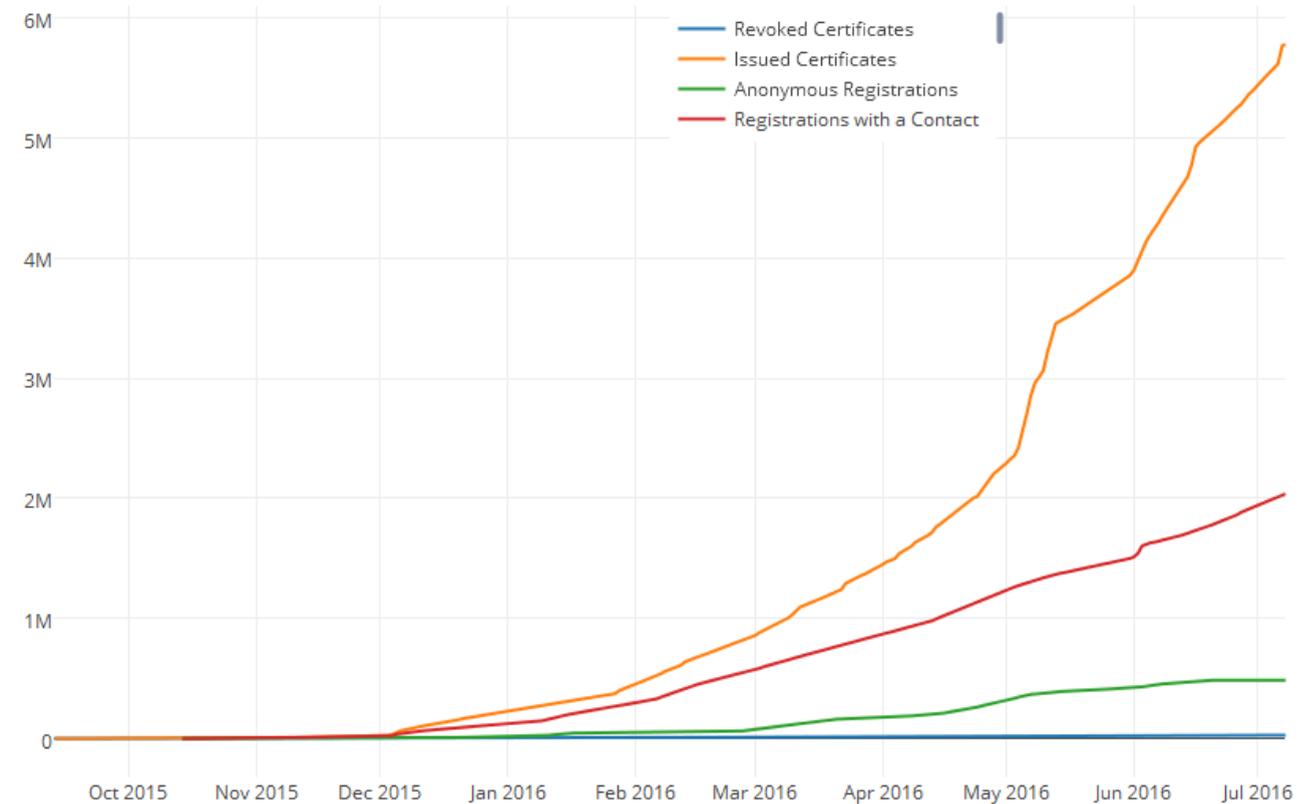
- Targeted malware which leverages basic HTTP over standard ports to blend in.
- This stream is composed of initial client registration to C2 server, along with post registration activity to validate interesting files on the system.
- Arid Viper originally focused on Israeli targets

Trends and Projections

> Encryption:

- Let's Encrypt could be huge game changer for malware
- Previously cost/overhead was high for SSL, Let's Encrypt eliminates this limitation.
- Won't impact state sponsored or targeted attacks much, but will impact Crimeware heavily.

Daily Activity



Source: Let's Encrypt: <https://letsencrypt.org/stats/>

Advanced Steg ** Recorded Demo **

> Steganography

- Hiding in plain site really is a powerful covert channel.
- Attackers may choose to take techniques which are not computationally difficult to generate, but are computationally difficult to detect, especially in real time network streams.
- Sky is the limit, this could be a very interesting topic for future discussions all on it's own.



Leveraging Cloud Apps

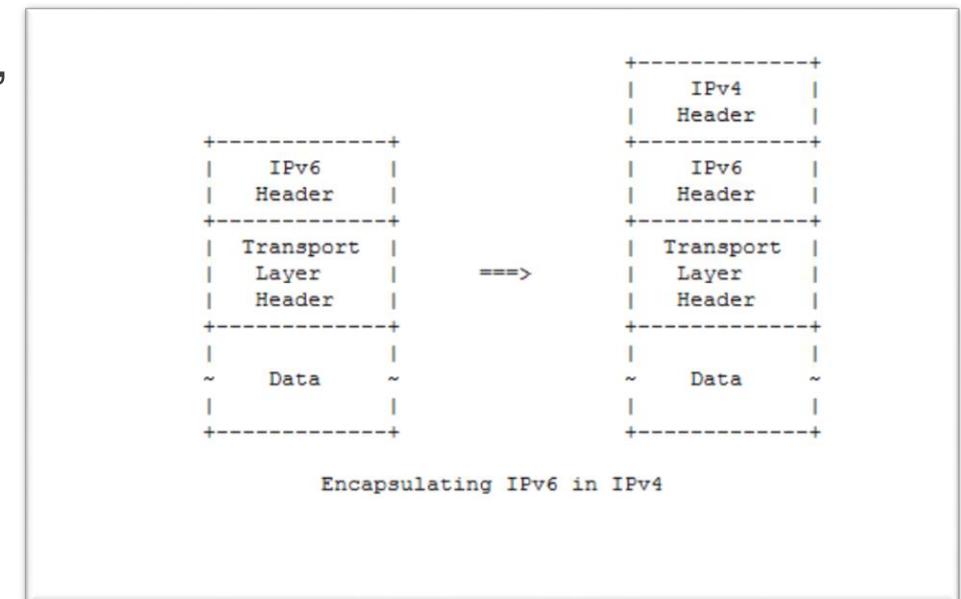
> Hiding C2 in Cloud Apps

- This is likely to be a continuing trend. It helps to solve the attacker challenge of hosting and potential blacklisting of standalone C2 infrastructure by overlaying it on top of cloud applications which often have business legitimacy.
- This makes it harder to detect and harder for organizations to take action on because they cannot block these apps.
- Puts the onus on Cloud providers to detect malicious activity. The effectiveness will vary widely depending on how invested these providers are.
- Cloud apps can be deployed with little more than an email address, often free compute infrastructure for attackers!

IPv6

> IPv6

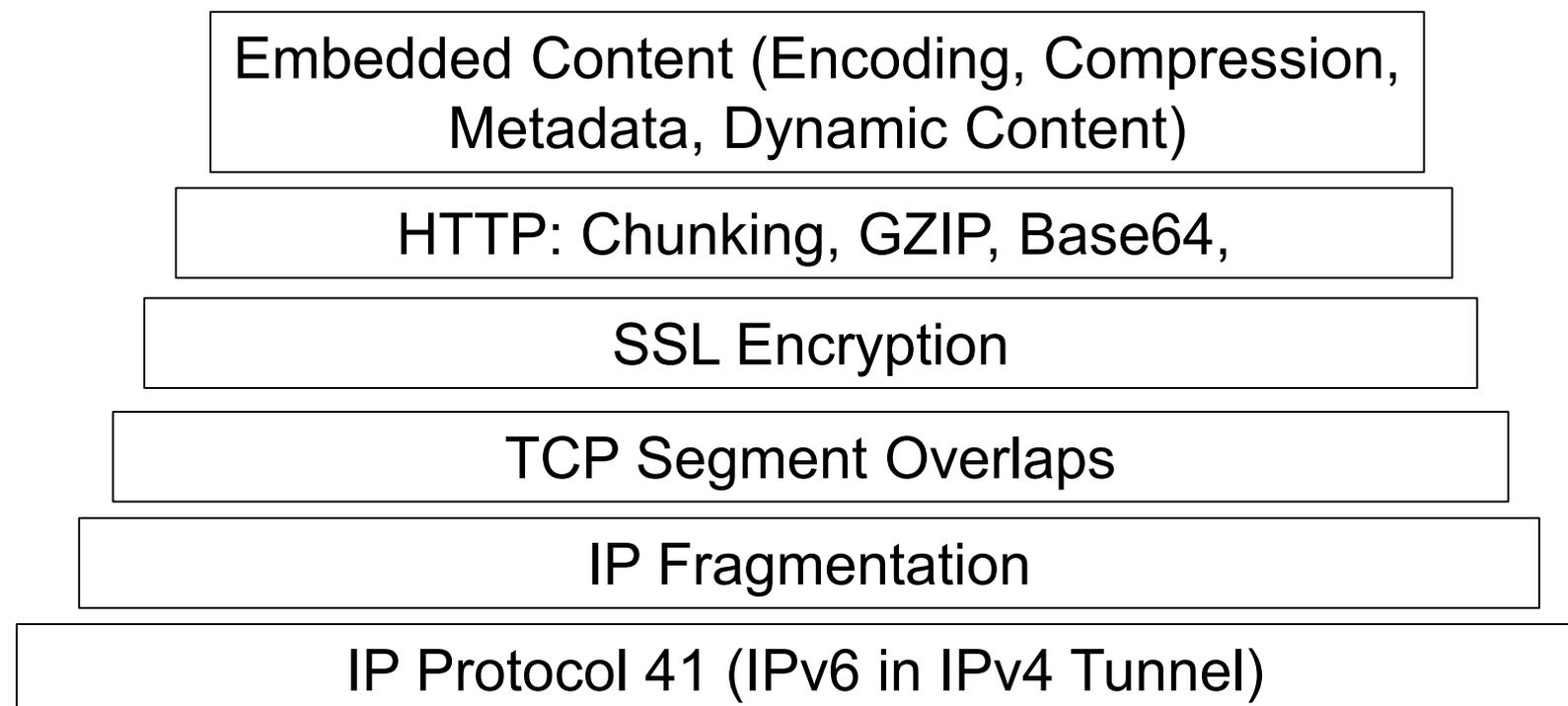
- Today IPv4 is still the predominate routed protocol on the internet, particularly outside of APAC and universities. This is changing
- IPv6 presents a big challenge because of the massive number of IPv6 addresses. We will be looking to do more blacklisting based on networks rather than IP's.
- IPv6 also may expose weaknesses in security software that does not support it yet or has underlying flaws and vulnerabilities.
- It is enabled by default in virtually every modern OS! Including IPv6inV4 Tunneling



Layered Evasions

> Layered Evasions

- Stacking numerous evasions from the IP level up the chain into the application layer to try to evade malicious activity detection by trying to fool detection capabilities (similar to traditional IDS layering evasion techniques).



C2 Detection Is Critical!

- High fidelity Indicator
- May prevent malware from successfully executing
- May prevent escalation to attack other hosts inside/outside the network
- May prevent sensitive data from making it out
- Makes more hoops for the attacker to jump through and therefore more opportunities to make a mistake.

Defense Mechanisms Phase 1

- Eliminate the Known Bad
 - Block access to known bad IP's, countries
 - Block Access to Malicious Domains

- Minimize the attack surface
 - Restrict FW/NGFW to least privilege including
 - Restrict Firewall Ports!, no ~~any any any~~ policy
 - Block unnecessary / undesirable L7 applications with an NGFW
 - Block unknown / unknown encrypted applications at the FW level with NGFW
 - Block queries to known/suspicious DNS domains

Defense Mechanisms Phase 2

> Fingerprint Known Malware

- Where possible, identify malware with both pattern matching and behavioral identification from a high fidelity source. If you can accurately identify malware itself, then you can have a higher degree of confidence of an infection.
- Especially if you can identify the malware by it's C2 channel

```
alert http $HOME_NET any -> $EXTERNAL_NET any (msg:"ET TROJAN Ransomware Locky CnC Beacon 21 May"; flow:established,to_server; content:"POST"; http_method; content:"/_dispatch.php"; fast_pattern; content:"www-form-urlencoded|0d 0a|"; http_header; content:"|0d 0a|x-requested-with|3a 20|XMLHttpRequest|0d 0a|"; http_header; pcre:"/^[0-9a-zA-Z=%-]{0,48}(?:%[A-F0-9]{2}){4}/Psi"; reference:md5,6f8987e28fed878d88858a943e7c6e7c; classtype:trojan-activity; sid:2022952; rev:2;)
```

> SSL Interception

- SSL Interception is an increasingly important function if it can be leveraged.
- It allows you to not only inspect encrypted streams, but also breaks any malware that uses predefined certificates.
- If you cannot do SSL interception, you can at least look at the network streams to try to fingerprint the certificate or identify anomalous SSL protocols.

Defense Mechanisms Phase 3

> SSL Interception

- SSL Interception is an increasingly important function if it can be leveraged.
- It allows you to not only inspect encrypted streams, but also breaks any malware that uses predefined certificates.
- If you cannot do SSL interception, you can at least look at the network streams to try to fingerprint the certificate or identify anomalous SSL protocols

> Known SSL Certs

- Where possible, use IDS or other technology to detect known malicious SSL certs which provide high fidelity indicators of an attack (even if SSL MiTM isn't possible)

```
alert tls $EXTERNAL_NET any -> $HOME_NET any (msg:"ET TROJAN ABUSE.CH SSL Blacklist Malicious SSL certificate detected (Dridex)"; flow:established,from_server; content:"|03 02 01 02 02 09 00|"; fast_pattern; content:"|30 09 06 03 55 04 06 13 02|"; distance:0; pcre:"/^[A-Z]{2}/R"; content:"|55 04 08|"; distance:0; content:"|55 04 07|"; distance:0; pcre:"/^{2}[A-Z][a-z]+(?:\x27[a-z]+|(?:\x20[A-Z][a-z]+){1,2})?[01]/Rs"; content:"|55 04 0a|"; distance:0; pcre:"/^{2}[A-Z][a-z]{3,}\s[A-Z][a-z]{3,}\s(?:[A-Z](?:[A-Za-z]{0,4}?[A-Z]|(?:\.[A-Za-z]){1,3})|[A-Z]?[a-z]+\.[01]/Rs"; content:"|55 04 03|"; distance:0; byte_test:1,>,7,1,relative; pcre:"/^{2}(?:[a-z]{1,4}(?:\d{3})?\.)?[a-z]{5,}\.(?!(?:com|net|org)[01])[a-z]{2,}[01]/Rs"; content:"|2a 86 48 86 f7 0d 01 09 01|"; reference:url,sslbl.abuse.ch; classtype:trojan-activity; sid:2022627; rev:8;)
```

Defense Mechanisms Phase 3

> Heuristics

- Pattern matching is not a perfect catch all for identifying suspicious activity due to highly evasive techniques.
- One high fidelity indicator of compromise can be to examine DNS data to try to identify domain generation algorithms used by modern malware.
- Some IDS can also identify this activity, but placement is very important because it needs to be between the client and the DNS server, otherwise all attacks will look like they are coming from the DNS server.

No.	Time	Source	Destination	Protocol	Length	Info
14	28.480299	198.41.0.4	192.168.58.10	DNS	496	Standard query response 0x0001 A dmowcuqwpbcaty2nedtmaamg4g.isc.org NS d0.org.afiliast-nst.org ...
28	30.049322	192.168.58.10	10.55.99.1	DNS	90	Standard query 0x0003 A a2cnkterz1chpybi5tso4vdapd.com
56	30.216163	192.168.58.10	4.2.2.1	DNS	90	Standard query 0x0010 A lz3i5hdrtoopzr11ptma5sjqc.com
57	30.245775	4.2.2.1	192.168.58.10	DNS	122	Standard query response 0x0010 A lz3i5hdrtoopzr11ptma5sjqc.com A 198.105.254.11 A 198.105.244...
58	30.246093	192.168.58.10	4.2.2.1	DNS	94	Standard query 0x0011 A www.lgvf4dsx1yaqck4wexruxddc0b.com
60	30.277899	4.2.2.1	192.168.58.10	DNS	126	Standard query response 0x0011 A www.lgvf4dsx1yaqck4wexruxddc0b.com A 198.105.254.11 A 198.105...
65	30.296153	192.168.58.10	10.55.99.1	DNS	94	Standard query 0x0014 A www.ydkpnyqgd4wafprtttkto5gle.com
77	30.448909	192.168.58.10	4.2.2.2	DNS	90	Standard query 0x001a A wor32lmugb01hcp4ddswnhepmb.com
78	30.457870	4.2.2.2	192.168.58.10	DNS	122	Standard query response 0x001a A wor32lmugb01hcp4ddswnhepmb.com A 198.105.254.11 A 198.105.244...
79	30.458100	192.168.58.10	4.2.2.2	DNS	94	Standard query 0x001b A www.pgqyhwrqzdcctgbhqqkuxqe2d.com
80	30.485405	192.168.58.10	10.55.99.1	DNS	81	Standard query 0x001c A net172.rebindtest.com
81	30.485561	4.2.2.2	192.168.58.10	DNS	126	Standard query response 0x001b A www.pgqyhwrqzdcctgbhqqkuxqe2d.com A 198.105.254.11 A 198.105...
82	30.496574	10.55.99.1	192.168.58.10	DNS	321	Standard query response 0x001c A net172.rebindtest.com A 172.16.0.1 NS k.gtld-servers.net NS b...
84	30.524708	192.168.58.10	10.55.99.1	DNS	87	Standard query 0x001d TXT 2.2.2.4.test.senderbase.org

Frame 65: 94 bytes on wire (752 bits), 94 bytes captured (752 bits)
Ethernet II, Src: AsustekC_8f:a0:3a (00:11:2f:8f:a0:3a), Dst: da:dd:49:3b:4b:9b (da:dd:49:3b:4b:9b)
Internet Protocol Version 4, Src: 192.168.58.10, Dst: 10.55.99.1
User Datagram Protocol, Src Port: 50072 (50072), Dst Port: 53 (53)
Domain Name System (query)

```
0000 da dd 49 3b 4b 9b 00 11 2f 8f a0 3a 08 00 45 00 ..I;K... /...E.  
0010 00 50 00 2f 00 00 80 11 d2 83 c0 a8 3a 0a 0a 37 .P./... .....7  
0020 63 01 c3 98 00 35 00 3c f0 44 00 14 01 00 00 01 c....5.< .D.....  
0030 00 00 00 00 00 00 77 77 77 1a 79 64 6b 70 6e .....w ww.ydkpn
```

> Network Profiling

- Leveraging a network profiling IDS like BRO can also help to not only identify malicious activity but also to provide a strong audit trail in the event that a breach occurs.



Defense Mechanisms Closing the Loop 4

> REVIEW YOUR SECURITY LOGS!

- As we've seen with many high profile breaches, it is often the case that malicious activity is detected, but it isn't acted upon.
- Most off the shelf malware and attacks provide many IOC's to key on which can be detected by freely available software and systems.
- There are commercial and open source solutions available that can help to solve the problem of the signal to noise, auxiliary endpoint verification, and end to end IR containment.

Summary

> Summary

- In modern computer security, it's not a matter of if, but when, and what they will take, and how much it will cost you to deal with it. The attack surface is simply too massive, to put all of your hopes in the fact that you might be able to keep malware out. In taking the fight to the attackers, we need to be smart, and to holistically detect breaches. Not only on the initial phases, but perhaps where the attackers are most exposed and we have the most defensive capabilities to detect them by detecting the C2 channels.
- As we continue to up our game, we should expect that the malicious actors will do the same, and come up with even more creative ways to leverage the same technology which can be used for incredible good for their own malicious purposes. But at the very least, we can keep them on their game, and further tip the economics of hacking by making their job that much harder. We'll do it by exploiting them for a change; at their weakest point, the command and control channel.

Thank You's!

Q&A

