Malware Command and Control Channels

- a journey into Darkness -

By Brad Woodberg

- Emerging Threats Product Manager / Proofpoint

Agenda

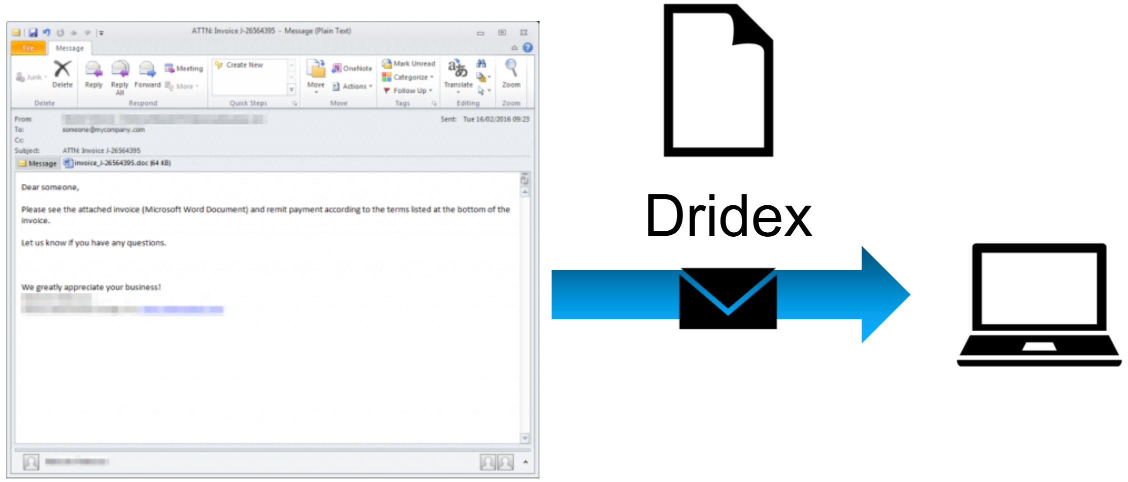
- C2 Intro and Background (7 mins)
- Modern C2 Techniques (6 mins)
- Case Studies (15 mins)
- Predictions for C2 (5 mins)
- Defense (10 mins)
- Wrap Up (2 mins)

Why Command & Control?

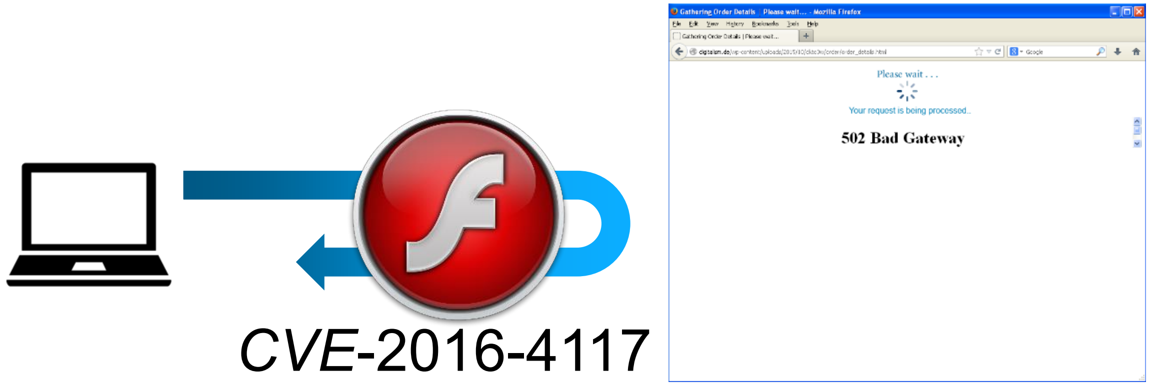
- Vulnerabilities, Exploits, and Malware grab the headlines and analyst focus
- While very interesting, it is also very noisy, many exploits fail, very FP prone.
- If you can effectively detect C2 activity, you have a high fidelity indicator that an asset is actually compromised.
- With C2, the tables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

Target Compromised, C2

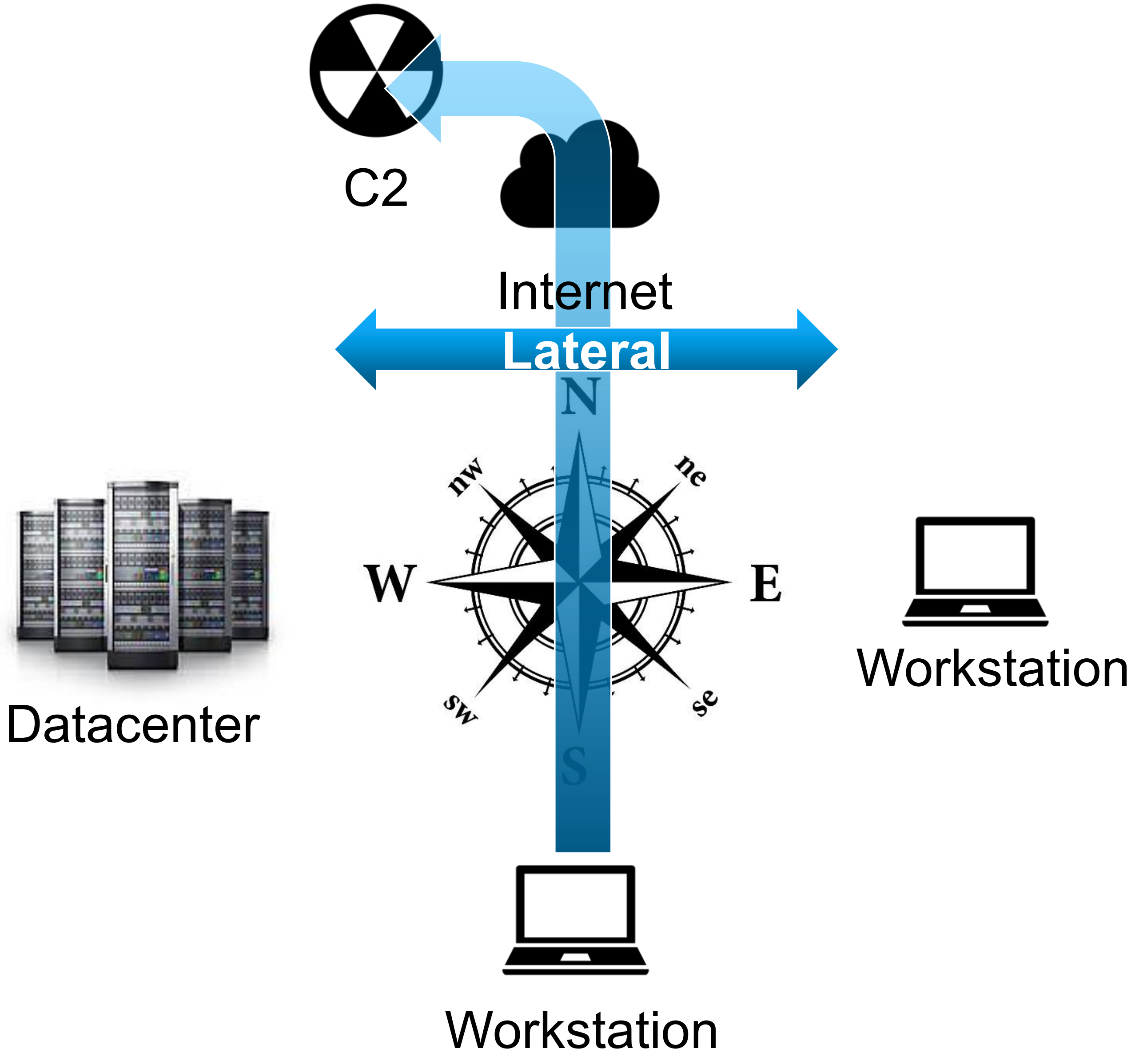
Exploit / Payload Delivered

TDS Evaluates Target Client

Redirect to Angler Infrastructure

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

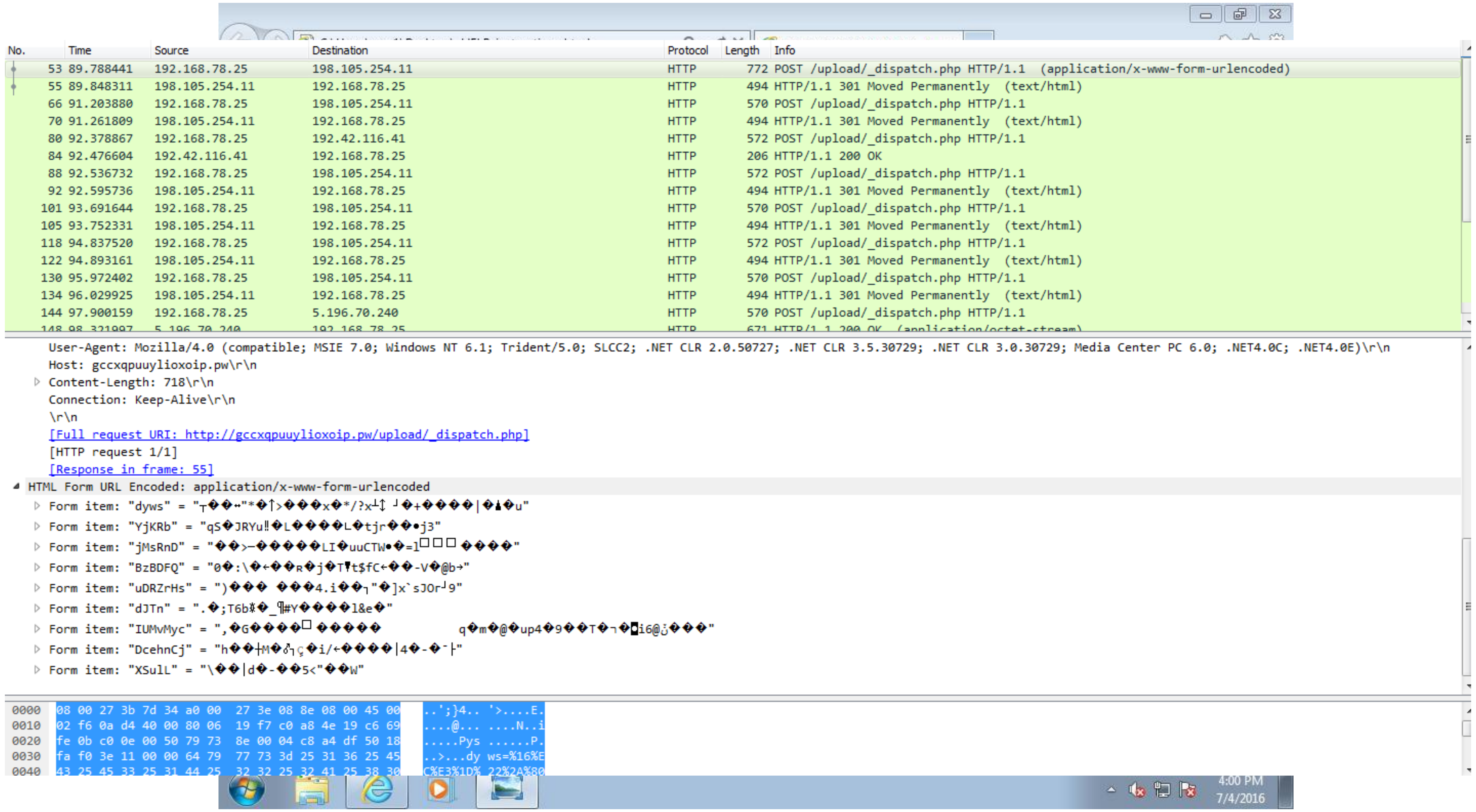


Lateral Infection vs. C2 Continued

- 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 it's 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



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

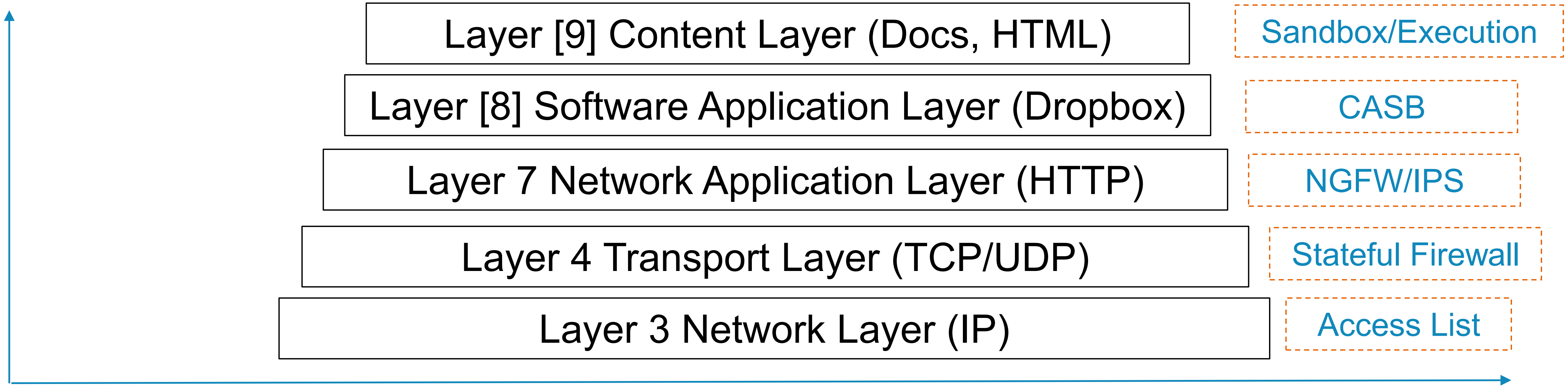
Targeted:

- 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:

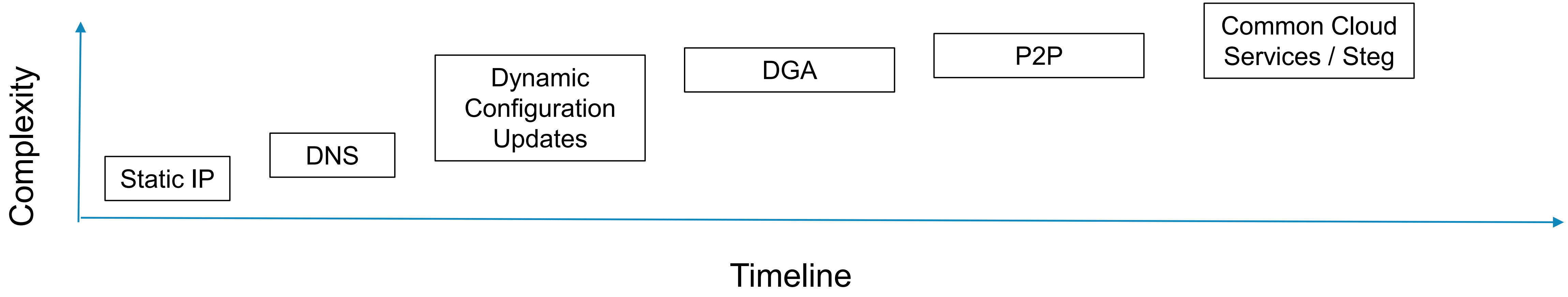
- Most exotic form of malware
- Far more sophisticated than traditional targeted.
- May lack network based C2 channels altogether.
- May leverage insiders as well as covert HW to bridge air gaps.

Cat and Mouse



- Enter the Stateful Firewall which leveraged Layer 7 payload inspection (similar to IPS) to identify applications rather than just ports.
  - 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 is true for potentially unwanted applications like Bittorent / Tor / Skype which also leveraged evasion techniques.
  - Early malware often heavily leveraged IRC channels for a simple C2 infrastructure e.g. PrettyPark (1999)
- To evade NGFW and other deep inspection technologies, malware shifted to leverage steganographic techniques to hide in plain sight. E.g.
  - As organizations tamped down on allowing ports outside of TCP 80/443 to communicate to the internet, so did malware evolve.
- Finally, malware has evolved even further to leverage highly obfuscated and embedded communication channels like jpgs, flash, encoded ASCII.
  - At the time, the malware just took advantage of the fact that stateful firewalls didn'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

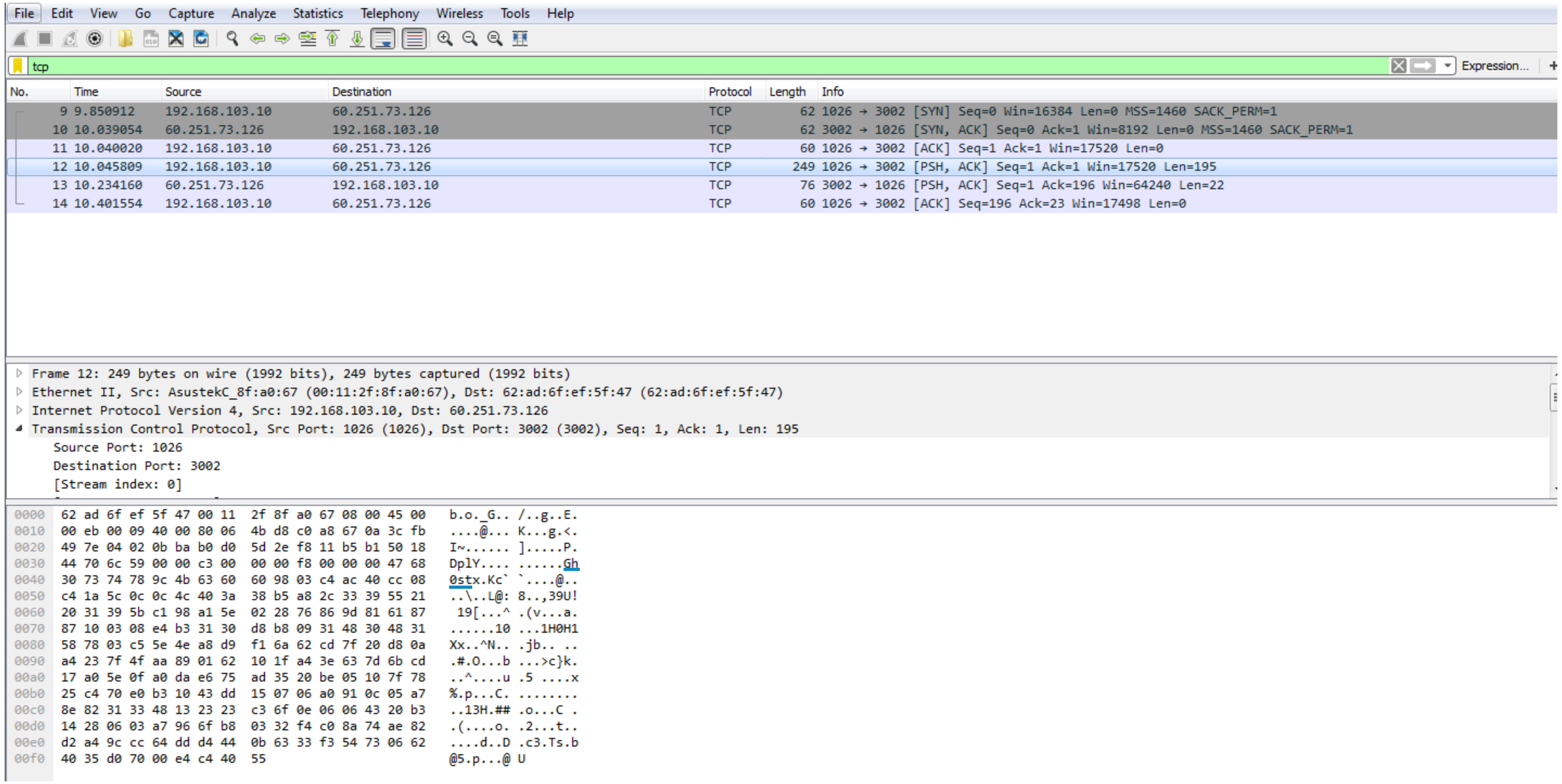
- Attackers think economically, want their malware to last as long as possible thus bringing the most ROI.
- Botnet authors utilize several counter detection techniques to ensure the viability of their malware.
  - Filter who can connect (e.g. IP filtering to eliminate non-targets, researchers and sandboxing 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.

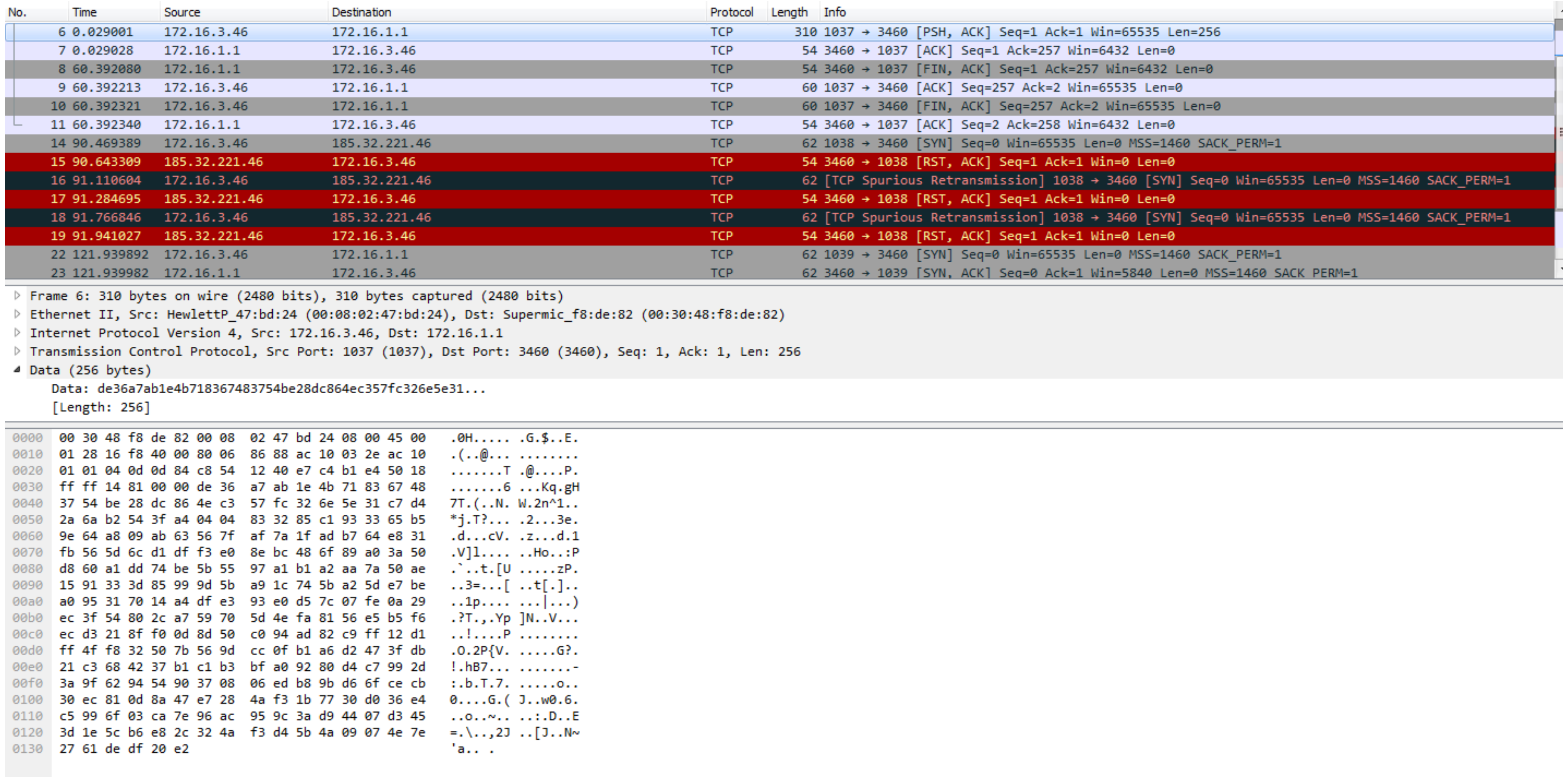
Gh0stRAT

- 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



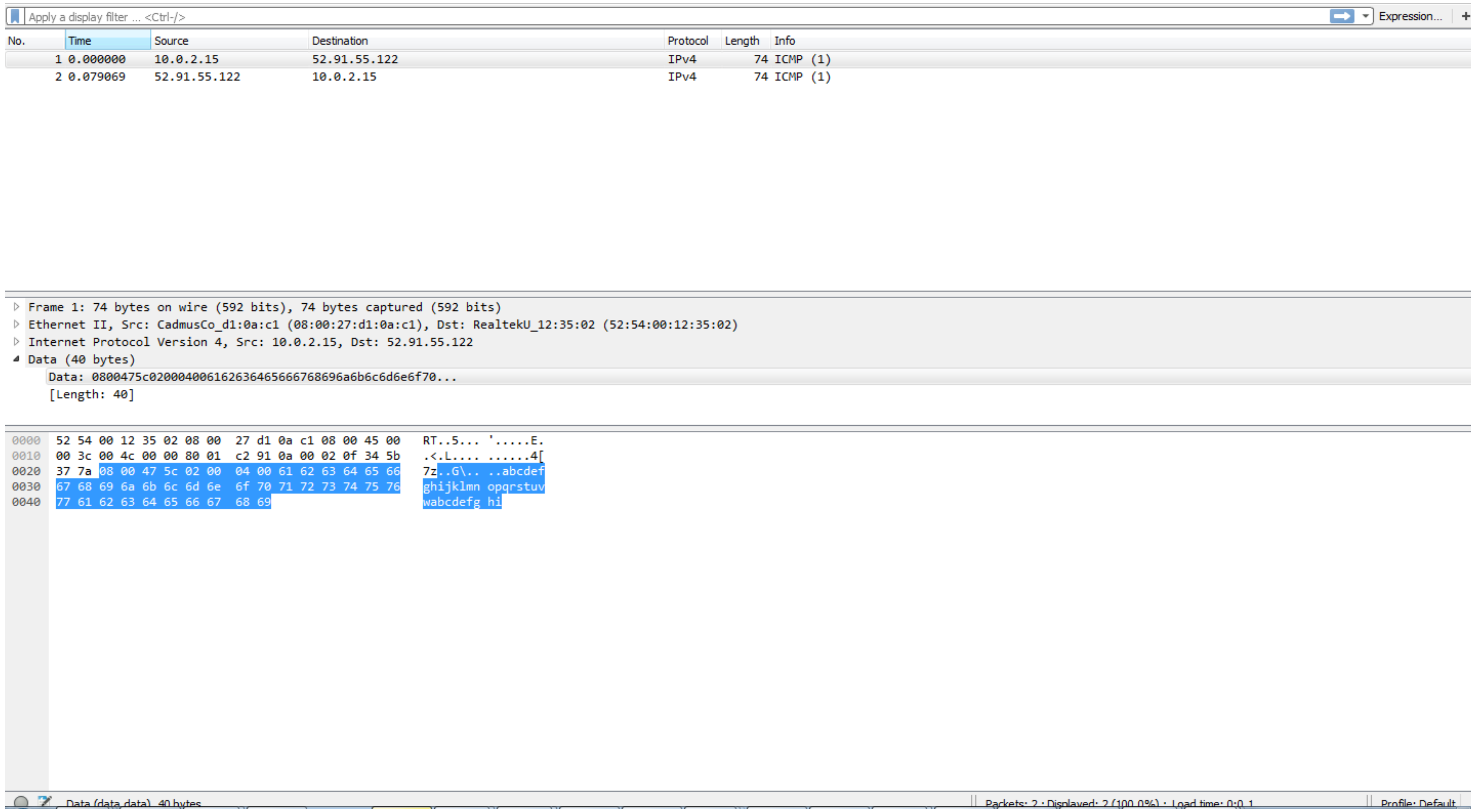
PoisonIvy

- 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.



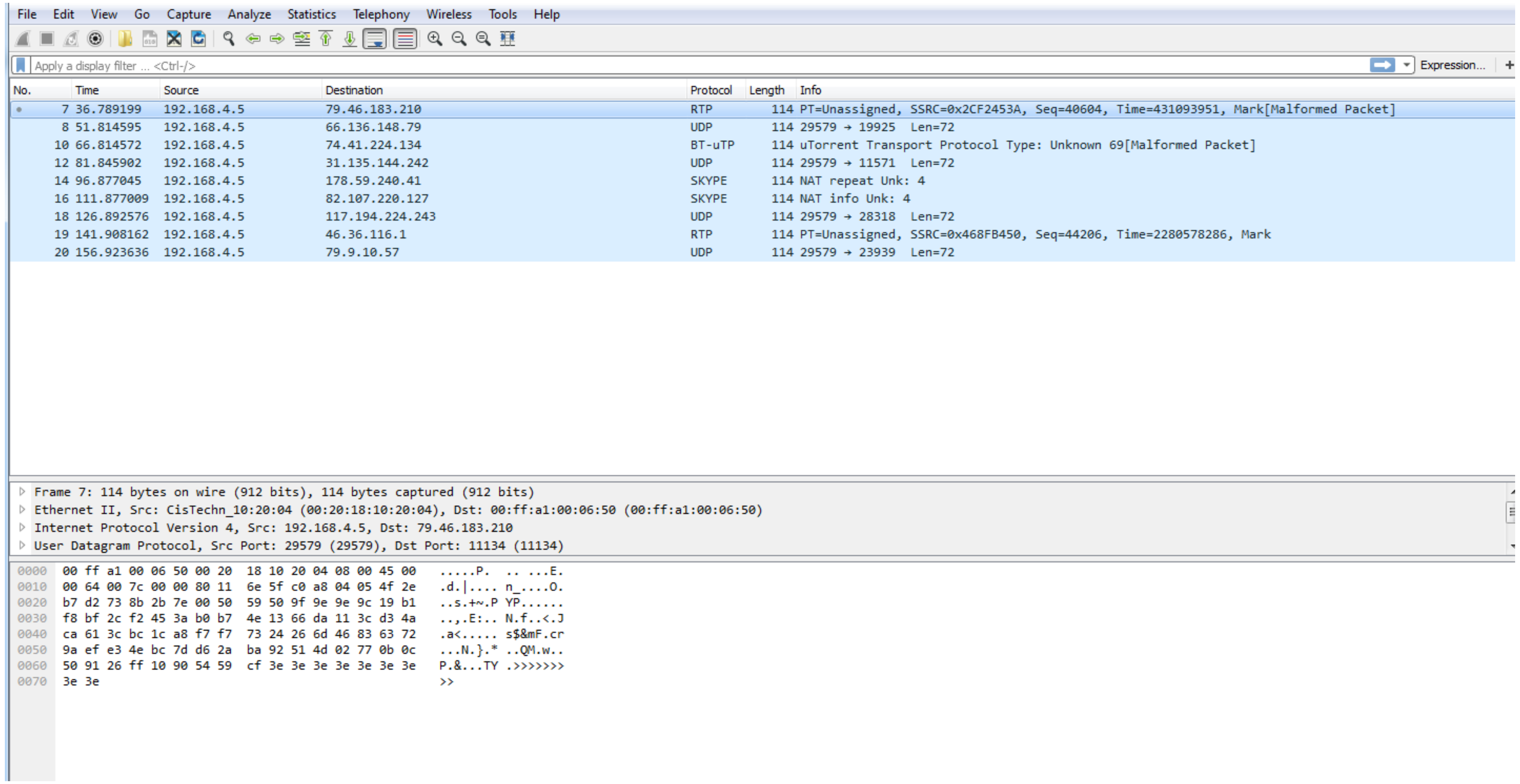
NanoLocker

- 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.



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.



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
Submision Date 2015-08-11 17:38:02
Type PCAP

SHA256 N/A
File Size N/A
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	2021076	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	2021076	ET INFO SUSPICIOUS Dotted Quad Host MZ Response	2	185.14.29.178	80	private	49431
2015-08-11	10000029	FILE ET magic 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

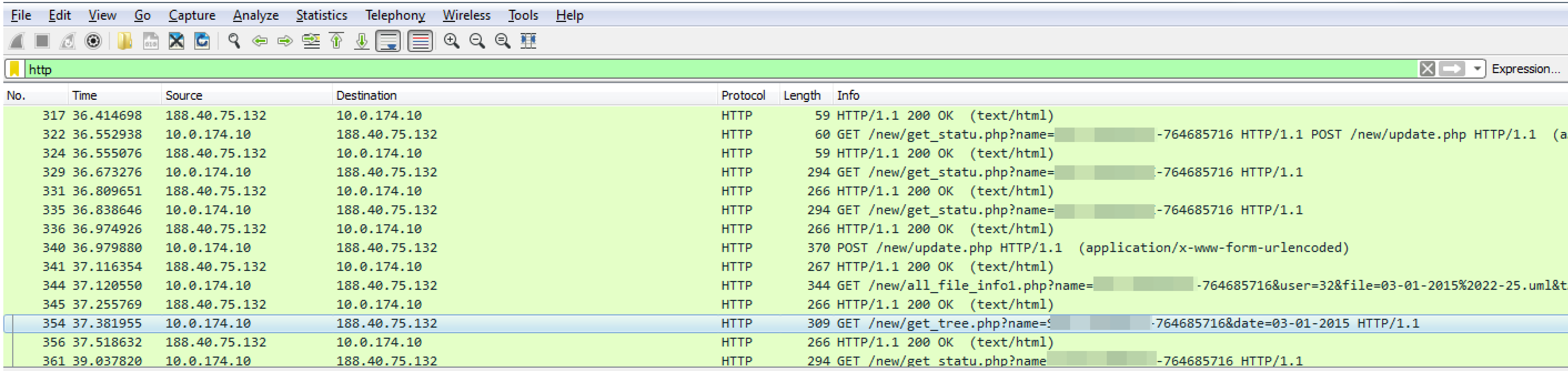
MD5 eef89c15b2625a8614d8c898fb802e04
Submision Date 2015-02-10 17:03:21
Type PE32 executable (GUI) Intel 80386, for MS Windows

SHA256 c026e9528b880d62e686c837494da9d6fc3ed90374f69c5496de63066eb9f575
File Size 46592
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	49366	46.19.37.108	80
2015-10-11	2019925	ET TROJAN Win32/Dalexis.A Possible SSL Cert (cargol.cat)	2	217.149.7.213	443	private	49354
2015-10-11	2019924	ET TROJAN Win32/Dalexis.A Possible SSL Cert (ppc.cba.pl)	2	85.17.73.180	443	private	49353

AridViper

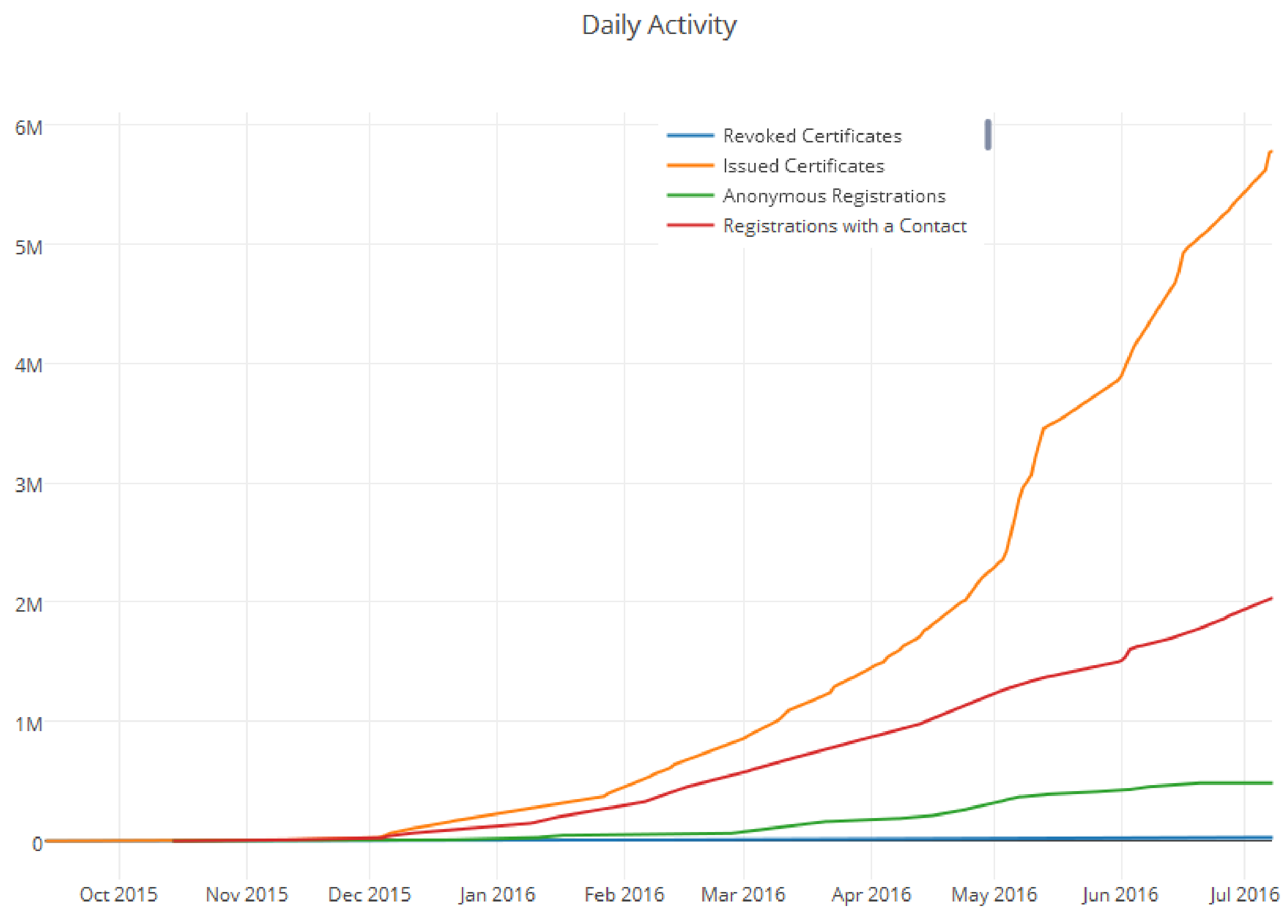


No.	Time	Source	Destination	Protocol	Length	Info
317	36.414698	188.40.75.132	10.0.174.10	HTTP	59	HTTP/1.1 200 OK (text/html)
322	36.552938	10.0.174.10	188.40.75.132	HTTP	60	GET /new/get_statu.php?name=[illegible]-764685716 HTTP/1.1 POST /new/update.php HTTP/1.1 (a
324	36.555076	188.40.75.132	10.0.174.10	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=[illegible]-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=[illegible]-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=[illegible]-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=[illegible]-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[illegible]-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

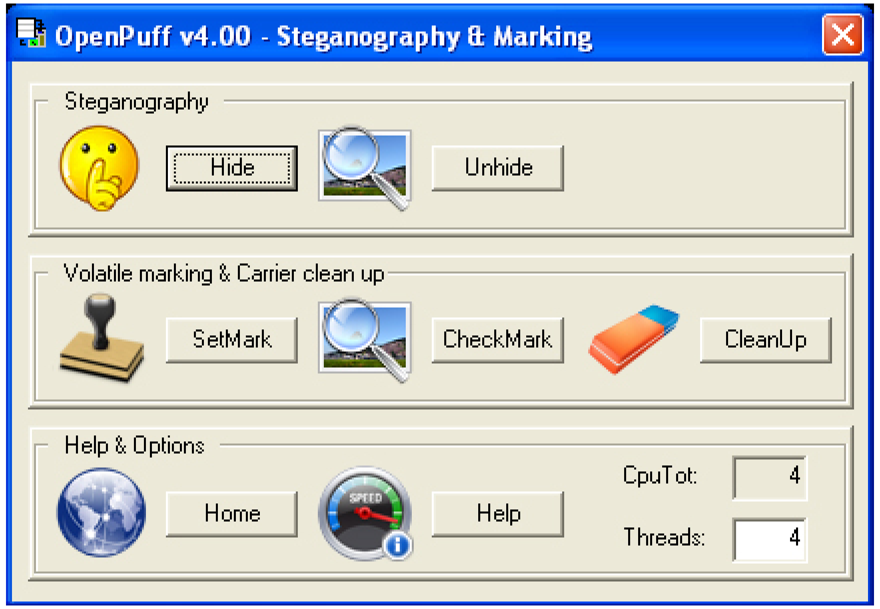
- Encryption:
  - Let’s Encrypt could be huge game changer for malware
  - Previously cost/overhead was high for SSL, Let’s Encrypt eliminates this limitation.
  - Won’t impact state sponsored or targeted attacks much, but will impact Crimeware heavily.



Source: Let’s Encrypt: https://letsencrypt.org/stats/

Advanced Steg ** Recorded Demo **

- Steganography
  - Hiding in plan 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.

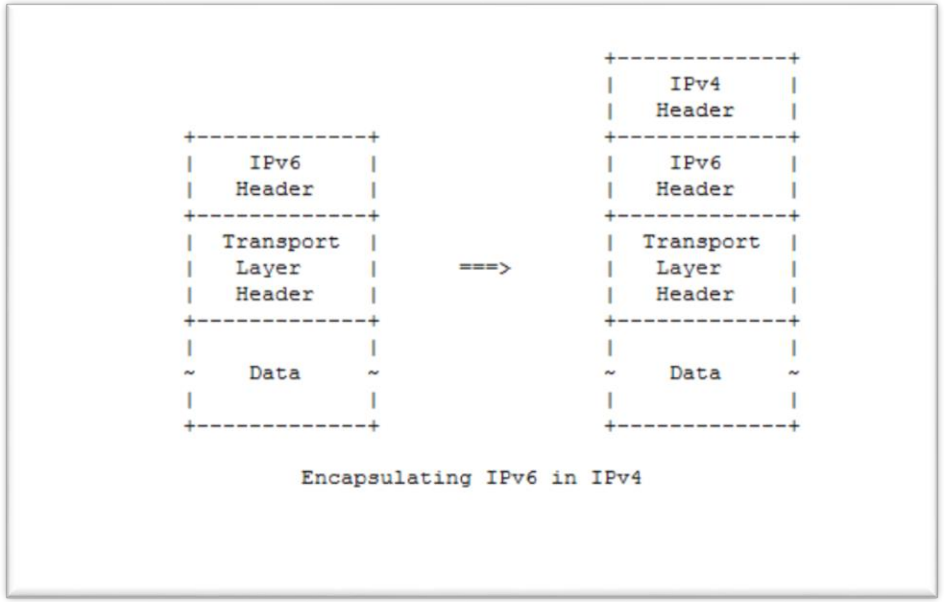


Leveraging Cloud Apps

- 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



Source: RFC 4213: https://tools.ietf.org/html/rfc4213, Nordmark and Gilligan

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.

Embedded Content (Encoding, Compression, Metadata, Dynamic Content)

HTTP: Chunking, GZIP, Base64,

SSL Encryption

TCP Segment Overlaps

IP Fragmentation

IP Protocol 41 (IPv6 in IPv4 Tunnel)

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_metho
d; 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:md5,6f8987e28fed878d08858a943e7c6e7c; classtype:trojan-activity; sid:202
2952; 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.

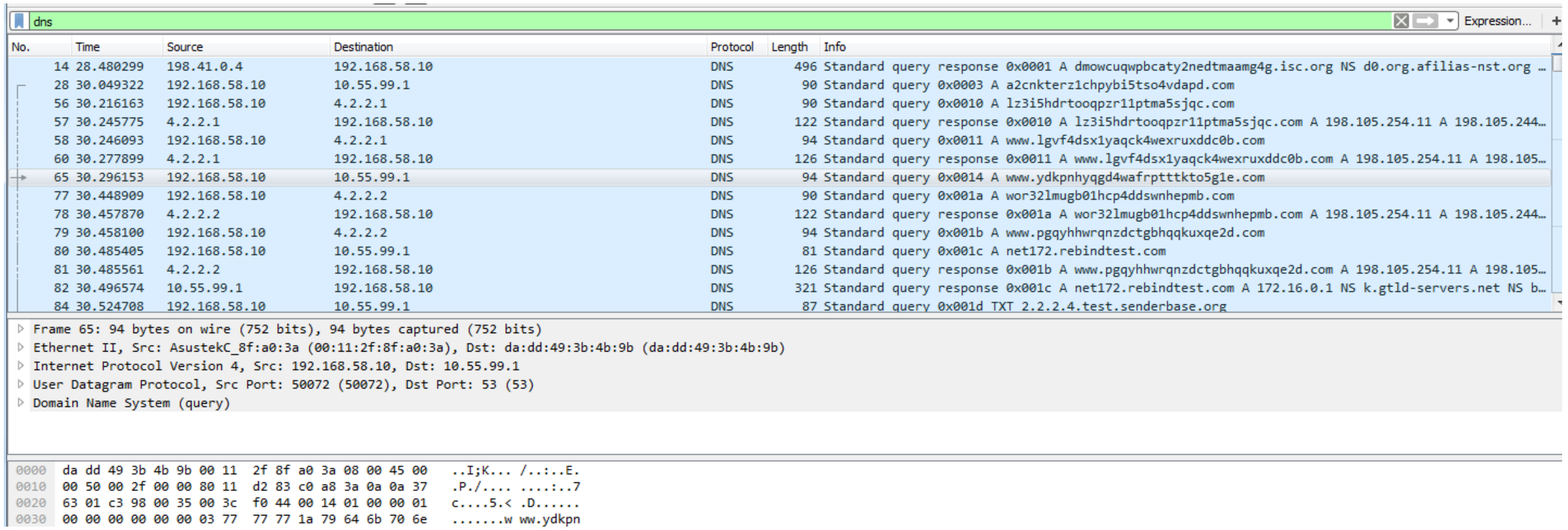
Defense Mechanisms Phase 3

- 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_s
erver; 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|"; di
stance: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; pc
re:"/^.{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.



- 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