



Getting Started With BHIS: SOC Analyst

John Strand



© Black Hills Information Security | @BHInfoSecurity



New Name!!!!



© Black Hills Information Security | @BHInfoSecurity



Getting Started With BHIS: MSP/SOC Analyst

John Strand



© Black Hills Information Security | @BHInfoSecurity

Big Thanks!!



LEVEL UP

The MSP Security Training Challenge

Presented by



Mission: Raise the collective security posture across the channel.

Our challenge for ourselves: Help 500 MSPs get training in 30 Days.

The channel needs more security practitioners.

That's why we've teamed up with vendors across the channel who are passionate about security to make some of the industry's best training more accessible and affordable.



© Black Hills Information Security, LLC

Our Sponsors

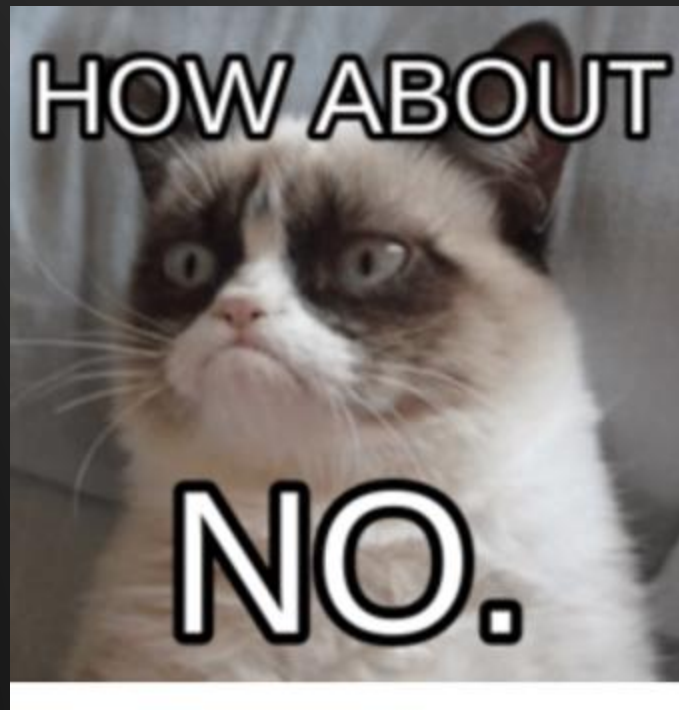
Each one of our sponsors has contributed funds to help secure the course discount and tuition assistance for those needing financial help. In addition, they each will be providing free seats in the course to help us hit our goal of providing the training to as many MSPs as possible.



© Black Hills Information Security

What We Are Covering

- Intro to Windows
- Intro to Linux
- Intro to TCP/IP
- Basics and fundamentals
- Core things to learn to work at the BHIS SOC
- This class is meant to feed into the Intro to Security class



Actually, yes. Today we are Grumpycat

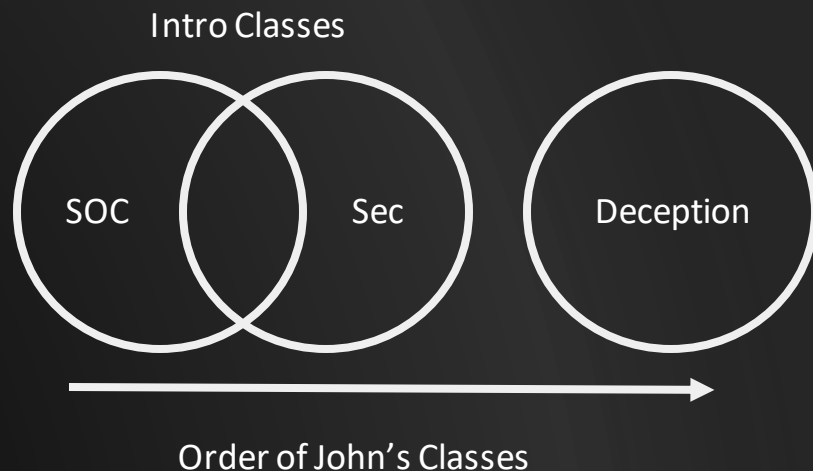


© Black Hills Information Security | @BHInfoSecurity

A Note On Overlap



- For this iteration, there will be some overlap with the Intro to Security class
 - Turns out, there is overlap in the topics.. Who knew?
- In the future, this class will feed into the Intro to Security Class
- The Intro to Security Class will feed to Cyber Deception
- For the near future, any class taught by me will be pay what you can



5 Year Plan

24
SEP
2016

HOW-TO, INFORMATIONAL, INFOSEC 101, WEBCASTS, CAREER CHANGE, GETTING INTO INFOSEC, GETTING STARTED, HOW-TO-GET INTO INFOSEC, STARTING YOUR CAREER

Webcast: John Strand's 5 Year Plan into InfoSec, Part 2

John Strand talks about his own journey into information security and shares his suggestions for those wanting to get started from scratch or who are looking to change career tracks.

Special Guests: Randy Marchany, CISO of Virginia Tech & Director of the VA Tech IT Security Lab, and Ed Capizzi, SANS instructor.



Show Notes / Links: *Just a few of the specific things that were referenced in this show*

FOLLOW US



LOOKING FOR
SOMETHING?

SUBSCRIBE TO THE
BHISBLOG

Don't get left in the dark! Enter your
email address and every time a post





You Are Compromised? What Now?

A bad day in the SOC...



© Black Hills Information Security | @BHInfoSecurity

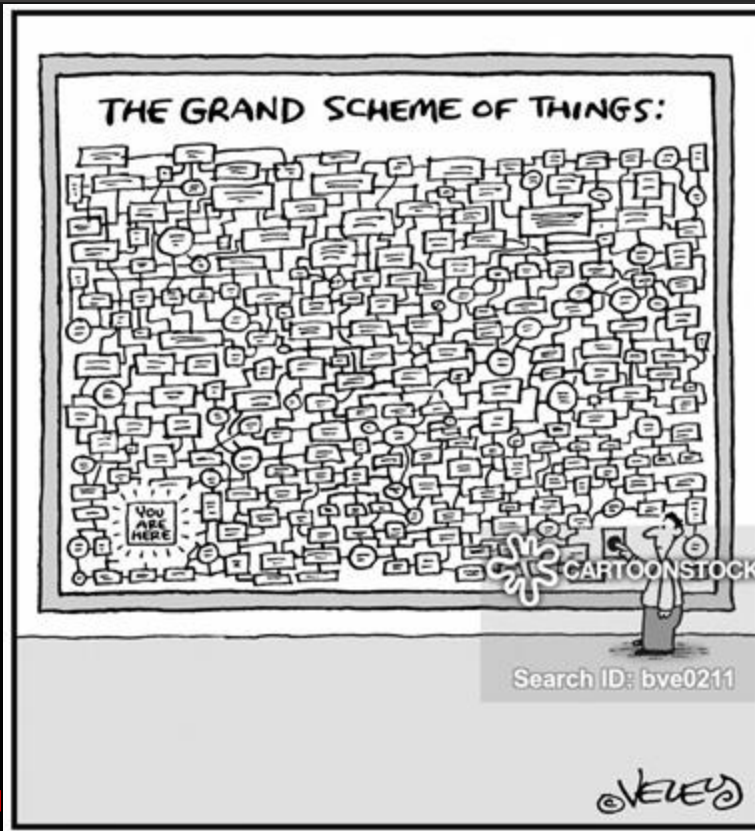


Why?

- First steps are tough..
- Mistakes and paralysis
- Need to keep moving
- Need to have a plan
- I want to cover some basic first steps



The Wrong Way...



© Black Hills Information Security



The Right Way



© Black Hills Information Security | @BHInfoSecurity



IR “Legos”



Don't Panic



- First step... Don't freak out
- I said DON'T FREAK OUT...
- DON'T FREAK OUT!!!!!!
- This only comes with practice
- Think weapons training
- Don't wait for an incident to try tools you have read about
- Memory forensics, Deep Blue CLI, IR Scripts, Logontracer, etc.



**KEEP
CALM
AND ...
NO. PANIC
DEFINITELY PANIC**



© Black Hills Information Security | @BHInfoSecurity

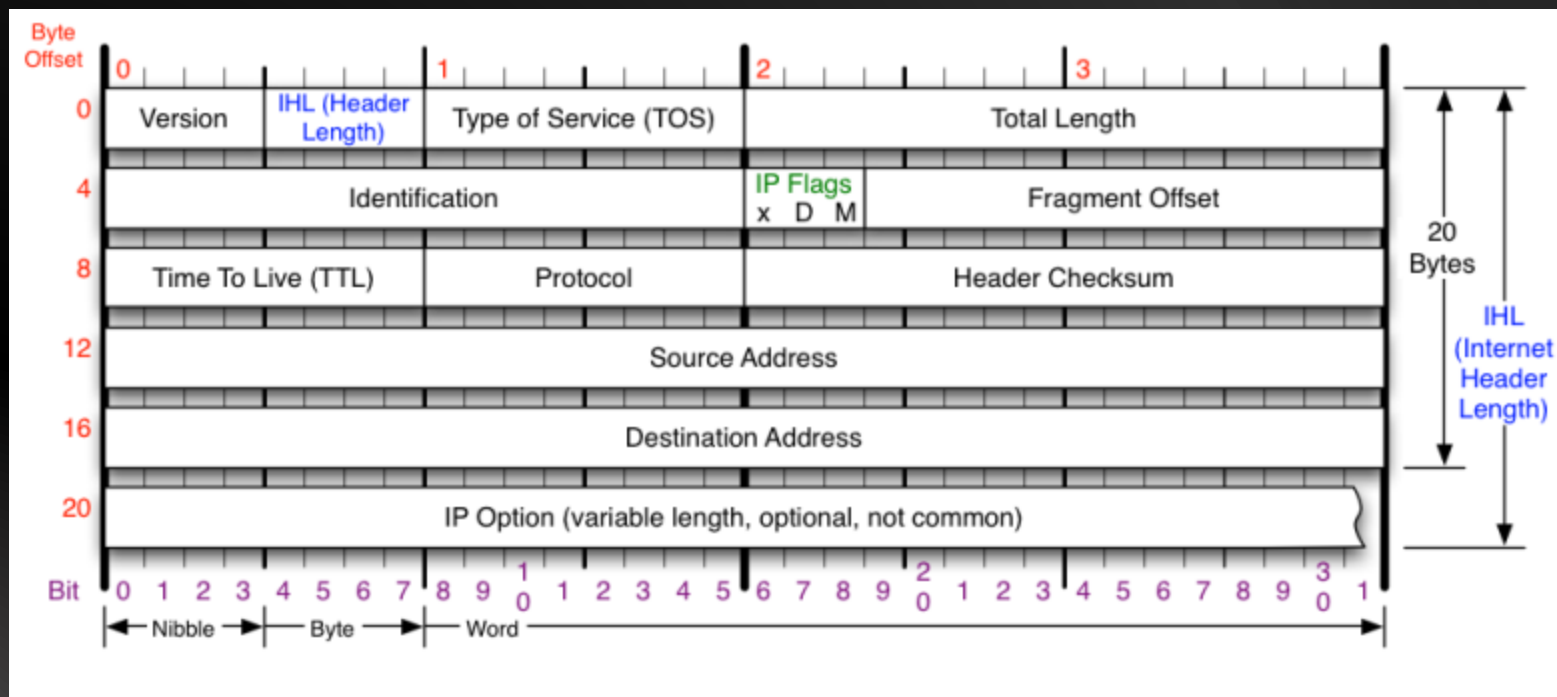


Networking!!!

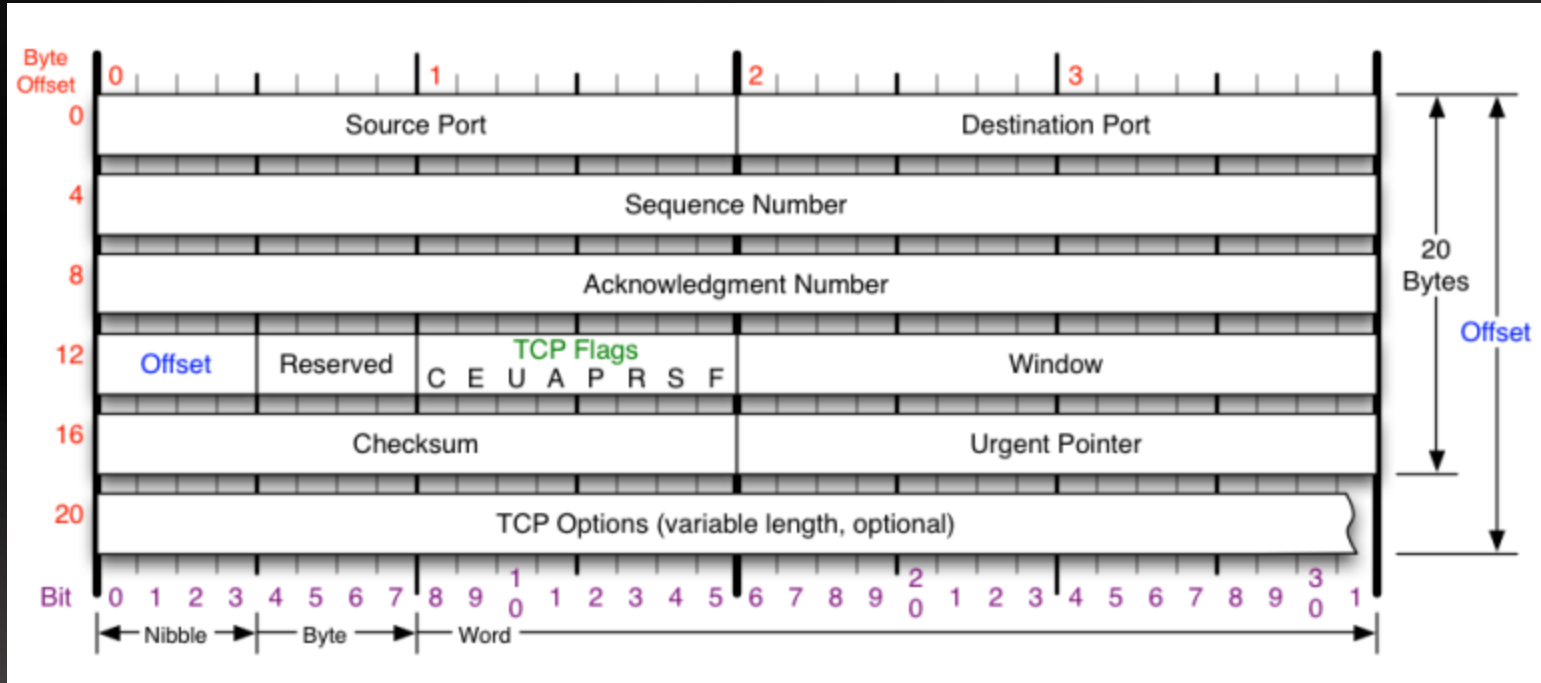


© Black Hills Information Security | @BHInfoSecurity

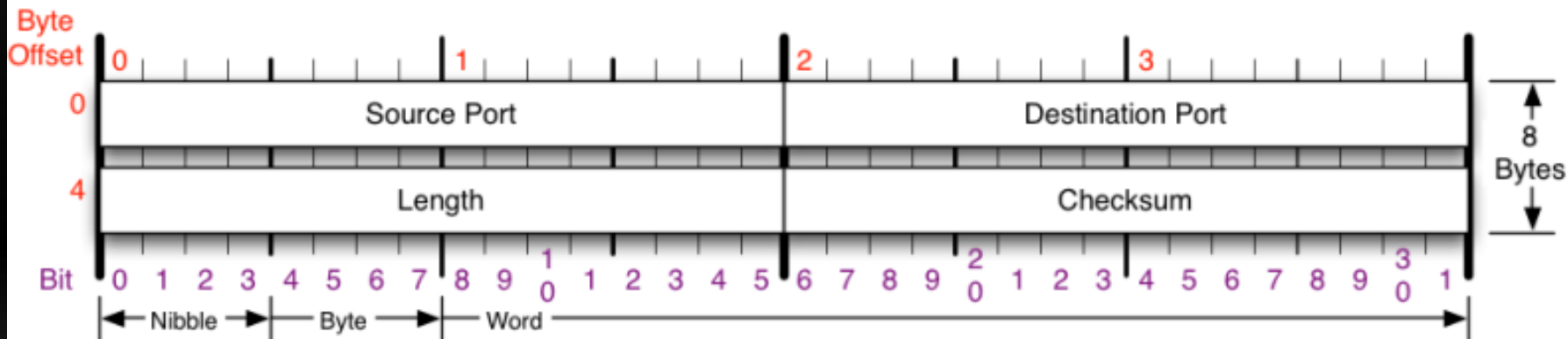
IP Header



TCP Header



UDP Header



Checksum

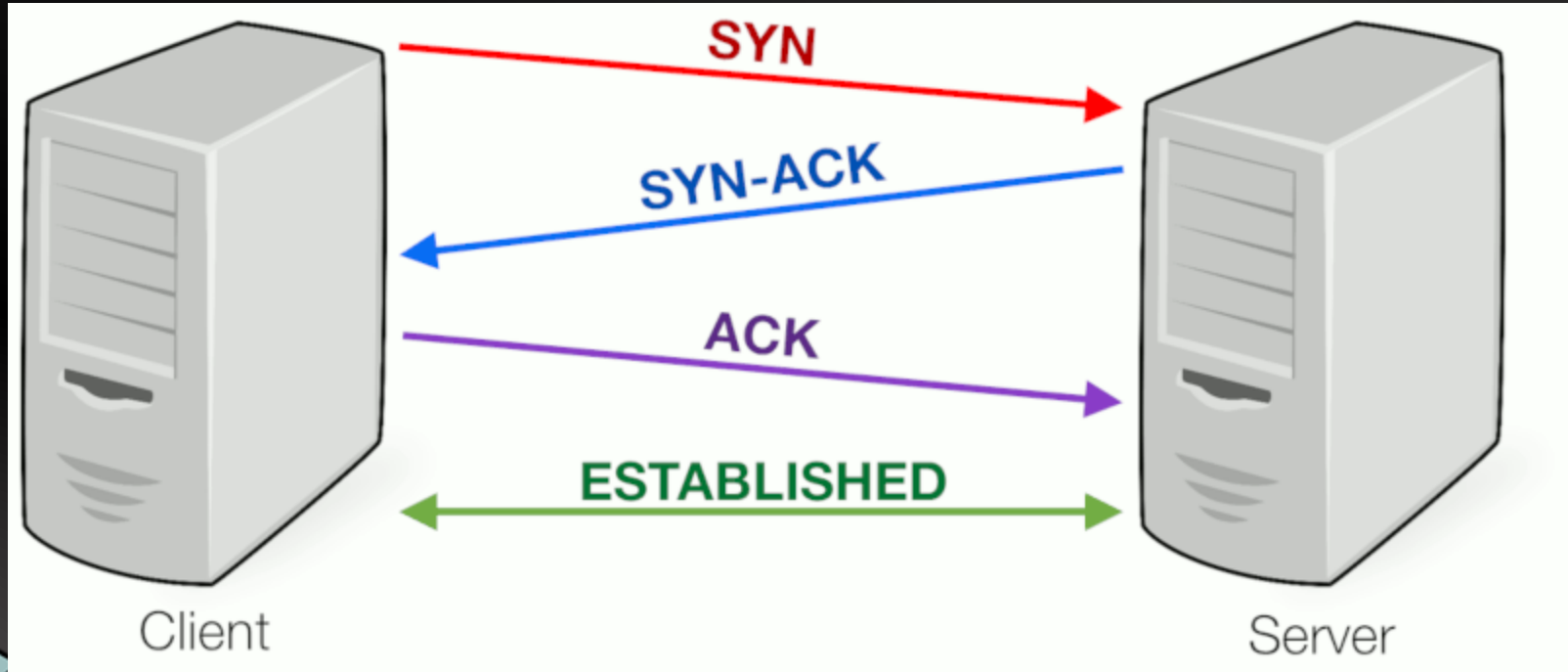
Checksum of entire UDP segment and pseudo header (parts of IP header)

RFC 768

Please refer to RFC 768 for the complete User Datagram Protocol (UDP) Specification.



TCP Three way Handshake



Top Ports



Insecure.Org

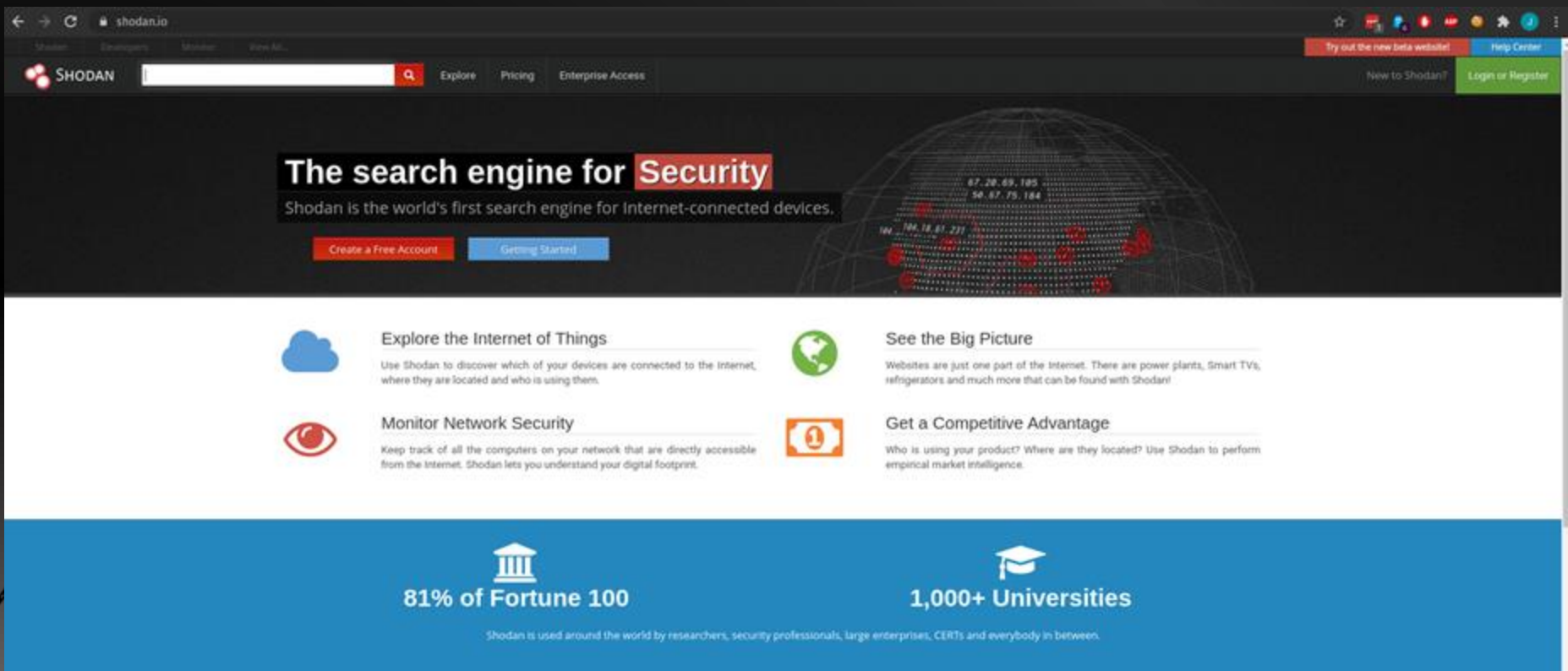
Top 10 TCP ports

- 80 (http)
- 23 (telnet)
- 22 (ssh)
- 443 (https)
- 3389 (ms-term-serv)
- 445 (microsoft-ds)
- 139 (netbios-ssn)
- 21 (ftp)
- 135 (msrpc)
- 25 (smtp)



© Black Hills Information Security

Shodan



The screenshot shows the Shodan website homepage. At the top, there's a navigation bar with the Shodan logo, a search bar, and links for Explore, Pricing, and Enterprise Access. A banner below the navigation bar features the text "The search engine for Security" and "Shodan is the world's first search engine for Internet-connected devices." Below this banner are four main sections: "Explore the Internet of Things", "Monitor Network Security", "See the Big Picture", and "Get a Competitive Advantage". Each section has an icon and a brief description. At the bottom, there's a blue footer section with two statistics: "81% of Fortune 100" and "1,000+ Universities", followed by a statement about Shodan's global usage.

Shodan

The search engine for **Security**

Shodan is the world's first search engine for Internet-connected devices.

Create a Free Account | Getting Started

Explore the Internet of Things

Use Shodan to discover which of your devices are connected to the Internet, where they are located and who is using them.

Monitor Network Security

Keep track of all the computers on your network that are directly accessible from the Internet. Shodan lets you understand your digital footprint.

See the Big Picture

Websites are just one part of the Internet. There are power plants, Smart TVs, refrigerators and much more that can be found with Shodan!

Get a Competitive Advantage

Who is using your product? Where are they located? Use Shodan to perform empirical market intelligence.

81% of Fortune 100

1,000+ Universities

Shodan is used around the world by researchers, security professionals, large enterprises, CERTs and everybody in between.

Shodan Ports



Shodan collects data mostly on **web** servers (HTTP/HTTPS – **ports** 80, 8080, 443, 8443), as well as FTP (**port** 21), SSH (**port** 22), Telnet (**port** 23), SNMP (**port** 161), IMAP (**ports** 143, or (encrypted) 993), SMTP (**port** 25), SIP (**port** 5060), and Real Time Streaming Protocol (RTSP, **port** 554).

[en.wikipedia.org > wiki > Shodan_\(website\)](https://en.wikipedia.org/wiki/Shodan_(website))

[Shodan \(website\) - Wikipedia](https://en.wikipedia.org/wiki/Shodan_(website))



tcpdump -D



-D Lists Interfaces

```
john@john-onion ~/pcaps> tcpdump -D
1.docker0 [Up, Running]
2.veth9807ef0 [Up, Running]
3.vethba446cd [Up, Running]
4.veth07191f2 [Up, Running]
5.veth53bc0a7 [Up, Running]
6.veth6b6fe9e [Up, Running]
7.vethc06fe9e [Up, Running]
8.ens33 [Up, Running]
9.vethe5b4e39 [Up, Running]
10.veth7539a85 [Up, Running]
11.veth028a400 [Up, Running]
12.vethbd60970 [Up, Running]
13.br-0edb29070257 [Up, Running]
14.any (Pseudo-device that captures on all interfaces) [Up, Running]
15.lo [Up, Running, Loopback]
16.bluetooth0 (Bluetooth adapter number 0)
17.nflog (Linux netfilter log (NFLOG) interface)
18.nfqueue (Linux netfilter queue (NFQUEUE) interface)
19.usbmon1 (USB bus number 1)
20.usbmon2 (USB bus number 2)
john@john-onion ~/pcaps>
```



tcpdump -X and -A



```
john@john-onion ~/pcaps> sudo tcpdump -i ens33 -XA
```

```
0x0050: 3435 3637 4567
19:28:09.078439 IP dns.google > john-onion: ICMP echo reply, id 58067, seq 2, length 64
0x0000: 4500 0054 61b4 0000 8001 b9bc 0808 0808 E..Ta.....
0x0010: c0a8 4e80 0000 ae60 e2d3 0002 498a 135e ..N....^....I..^
0x0020: 0000 0000 530e 0000 0000 0000 1011 1213 .....S.....
0x0030: 1415 1617 1819 1a1b 1c1d 1e1f 2021 2223 .....!"#
0x0040: 2425 2627 2829 2a2b 2c2d 2e2f 3031 3233 $%&'()*+,-./0123
0x0050: 3435 3637 4567
19:28:10.005420 IP john-onion > dns.google: ICMP echo request, id 58067, seq 3, length 64
0x0000: 4500 0054 55ac 4000 4001 c5c4 c0a8 4e80 E..TU.@. ....N.
0x0010: 0808 0808 0800 e558 e2d3 0003 4a8a 135e .....X....J..^
0x0020: 0000 0000 1315 0000 0000 0000 1011 1213 .....
0x0030: 1415 1617 1819 1a1b 1c1d 1e1f 2021 2223 .....!"#
0x0040: 2425 2627 2829 2a2b 2c2d 2e2f 3031 3233 $%&'()*+,-./0123
0x0050: 3435 3637 4567
19:28:10.145845 IP dns.google > john-onion: ICMP echo reply, id 58067, seq 3, length 64
0x0000: 4500 0054 61b5 0000 8001 b9bb 0808 0808 E..Ta.....
0x0010: c0a8 4e80 0000 ed58 e2d3 0003 4a8a 135e ..N....X....J..^
0x0020: 0000 0000 1315 0000 0000 0000 1011 1213 .....
0x0030: 1415 1617 1819 1a1b 1c1d 1e1f 2021 2223 .....!"#
0x0040: 2425 2627 2829 2a2b 2c2d 2e2f 3031 3233 $%&'()*+,-./0123
0x0050: 3435 3637 4567
```

-X is for the Hex
-A is for the ASCII



tcpdump: host, port and -r



```
john@john-onion ~/pcaps> tcpdump -r taidoor_traffic_no_interaction.pcap -X -A host 10.0.2.15 and port 80
```

-r = read a previous capture

```
16:09:36.179880 IP 10.0.2.15.49845 > 104.248.234.238.http: Flags [P.], seq 1:516, ack 1, win 65535, length 515: HTTP: GET /process.jsp?mn=IOEHPJEALJEPFPEDJDFMBLNDHBAFJCIECPOMOHMNFKIPNMJIFBGHGLJIJOAMCBDBKBFPEONMJAFAKMNKBGGJOPKHJPJOGGLPGBDNCKIOBDFOLKAODLKLBDNFLKFOHABGIKCDPNNABOGHBDHCGIGBIPBHLHCHIKKOHAIHIFCAOHGNLDNKPBLEAHKAFOLHLPGBFOHIFDKNNCOGNHPDHIHLABKCMMBGCOMBEIBAPHJIHGOCBHBBOGJHFENJNILIPMA HTTP/1.1
```

0x0000:	4500	022b	0926	4000	8006	0000	0a00	020f	E..+.&@.....
0x0010:	68f8	eaee	c2b5	0050	57f5	8e78	27b7	f802	h.....PW..x'...
0x0020:	5018	ffff	6213	0000	4745	5420	2f70	726f	P...b...GET./pro
0x0030:	6365	7373	2e6a	7370	3f6d	6e3d	494f	4548	cess.jsp?mn=IOEH
0x0040:	504a	4541	4c4a	4550	4650	4544	4a44	464d	PJEALJEPFPEDJDFM
0x0050:	424c	4e48	4442	4146	4a43	4945	4350	4f4d	BLNHDBAFJCIECPOM
0x0060:	4f48	4d4e	464b	4950	4e4d	4a49	4642	4748	OHMNFKIPNMJIFBGH
0x0070:	474c	4a49	4a4f	414d	4342	4442	4b42	4650	GLJIJOAMCBDBKBF
0x0080:	454f	4e4d	4a41	464b	4d4e	4b42	4747	4a4f	EONMJAFAKMNKBGGJO
0x0090:	504b	484a	504a	4f47	474c	5047	4244	4e43	PKHJPJOGGLPGBDNC
0x00a0:	4b49	4f42	4446	4f4c	4b41	4f44	4c4b	4c42	KIOBDFOLKAODLKL
0x00b0:	4444	464c	4b46	4f48	4142	4749	4b43	4450	DDFLKFOHABGIKCDP



tcpdump -w



```
john@john-onion ~/pcaps> tcpdump -i ens33
```

-w is to write the data to a file



© Black Hills Information Security | @BHInfoSecurity



LAB: TCPDump



© Black Hills Information Security | @BHInfoSecurity

Wireshark



© Black Hills Information Security | @BHInfoSecurity

Wireshark and Interfaces



Welcome to Wireshark

Open

/home/john/pcaps/taidoor_traffic_no_interaction.pcap (291 KB)

Capture

...using this filter: All interfaces shown

docker0
veth9807ef0
veth8a446cd
veth07191f2
veth53bc0a7
veth6b6fc9e
vethc06fe9e
ens33
veth5b4e39
veth7539a85
veth028a400
veth8d60970
br-0edb29070257
any
loopback: lo
bluetooth0
nftog
nftoguse
usbmon1
usbmon2
Cisco remote capture: ciscodump
Random packet generator: randpkt
SSH remote capture: sshdump
UDP Listener remote capture: udpdump

Choose wisely..

Watching the traffic



Capture

...using this filter:  Enter a capture filter ...

docker0

veth9807ef0

vethba446cd

veth07191f2

veth53bc0a7

veth6b6fe9e

vethc06fe9e

ens33

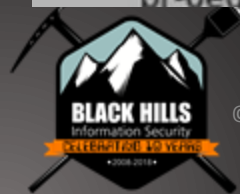
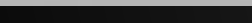
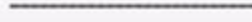
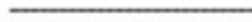
veth5b4e39

veth7539a85

veth028a400

vethbd60970

br-0ed29070257



Wireshark and ping



File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help

Apply a display filter: <Ctrl-F>

No.	Time	Source	Destination	Protocol	Length	Info
4	1.08769642	8.8.8.8	192.168.78.128	ICMP	98	Echo (ping) reply id=0xeb17, seq=2/512, ttl=128 (request in 3)
5	2.004877175	192.168.78.128	8.8.8.8	ICMP	98	Echo (ping) request id=0xeb17, seq=3/768, ttl=64 (reply in 6)
6	2.077256052	8.8.8.8	192.168.78.128	ICMP	98	Echo (ping) reply id=0xeb17, seq=3/768, ttl=128 (request in 5)
7	3.097830581	192.168.78.128	8.8.8.8	ICMP	98	Echo (ping) request id=0xeb17, seq=4/1024, ttl=64 (reply in 8)
8	3.077607996	8.8.8.8	192.168.78.128	ICMP	98	Echo (ping) reply id=0xeb17, seq=4/1024, ttl=128 (request in 7)
9	4.010325051	192.168.78.128	8.8.8.8	ICMP	98	Echo (ping) request id=0xeb17, seq=5/1280, ttl=64 (reply in 10)
10	4.067996146	8.8.8.8	192.168.78.128	ICMP	98	Echo (ping) reply id=0xeb17, seq=5/1280, ttl=128 (request in 9)
11	5.013397556	192.168.78.128	8.8.8.8	ICMP	98	Echo (ping) request id=0xeb17, seq=6/1536, ttl=64 (reply in 12)
12	5.077439419	8.8.8.8	192.168.78.128	ICMP	98	Echo (ping) reply id=0xeb17, seq=6/1536, ttl=128 (request in 11)
13	6.014979999	192.168.78.128	8.8.8.8	ICMP	98	Echo (ping) request id=0xeb17, seq=7/1792, ttl=64 (reply in 14)
14	6.078445118	8.8.8.8	192.168.78.128	ICMP	98	Echo (ping) reply id=0xeb17, seq=7/1792, ttl=128 (request in 13)
15	7.016632749	192.168.78.128	8.8.8.8	ICMP	98	Echo (ping) request id=0xeb17, seq=8/2048, ttl=64 (reply in 16)
16	7.095525879	8.8.8.8	192.168.78.128	ICMP	98	Echo (ping) reply id=0xeb17, seq=8/2048, ttl=128 (request in 15)
17	8.018774859	192.168.78.128	8.8.8.8	ICMP	98	Echo (ping) request id=0xeb17, seq=9/2304, ttl=64 (reply in 18)
18	8.180699887	8.8.8.8	192.168.78.128	ICMP	98	Echo (ping) reply id=0xeb17, seq=9/2304, ttl=128 (request in 17)
19	9.019955825	192.168.78.128	8.8.8.8	ICMP	98	Echo (ping) request id=0xeb17, seq=10/2560, ttl=64 (reply in 20)
20	9.077489556	8.8.8.8	192.168.78.128	ICMP	98	Echo (ping) reply id=0xeb17, seq=10/2560, ttl=128 (request in 19)
21	10.023519183	192.168.78.128	8.8.8.8	ICMP	98	Echo (ping) request id=0xeb17, seq=11/2816, ttl=64 (reply in 22)
22	10.085618832	8.8.8.8	192.168.78.128	ICMP	98	Echo (ping) reply id=0xeb17, seq=11/2816, ttl=128 (request in 21)

▶ Frame 9: 98 bytes on wire (784 bits), 98 bytes captured (784 bits) on interface 0
▶ Ethernet II, Src: VMware_46:52:0b (00:9c:2b:46:52:0b), Dst: VMware_eb:58:26 (00:50:56:eb:58:26)
▶ Internet Protocol Version 4, Src: 192.168.78.128, Dst: 8.8.8.8
▶ Internet Control Message Protocol

0000 69 59 56 eb 58 26 69 6c 29 46 82 06 08 90 45 90 PV X6)Fb E-
0018 69 54 22 f1 49 69 49 01 f8 7f c0 a8 4e 08 98 98 T* 0 0 N
0020 68 08 08 09 ed f1 eb 17 90 95 49 8d 13 5e 90 90 I A-
0030 69 68 f8 82 00 69 69 09 00 10 11 12 13 14 15 2 !*%K
0040 18 17 18 19 1a 1b 1c 1d 1e 1f 20 21 22 23 24 25
0050 26 27 28 29 2a 2b 2c 2d 2e 2f 30 31 32 33 34 35 4'()*+,-./012345
0060 36 37 67



© Black Hills

Packet Breakdown



```
▶ Frame 9: 98 bytes on wire (784 bits), 98 bytes captured (784 bits) on interface 0
▼ Ethernet II, Src: Vmware_46:62:0b (00:0c:29:46:62:0b), Dst: Vmware_eb:58:26 (00:50:56:eb:58:26)
  ▼ Destination: Vmware_eb:58:26 (00:50:56:eb:58:26)
    Address: Vmware_eb:58:26 (00:50:56:eb:58:26)
    .....0. .... = IG bit: Globally unique address (factory default)
    .....0. .... = IG bit: Individual address (unicast)
  ▼ Source: Vmware_46:62:0b (00:0c:29:46:62:0b)
    Address: Vmware_46:62:0b (00:0c:29:46:62:0b)
    .....0. .... = IG bit: Globally unique address (factory default)
    .....0. .... = IG bit: Individual address (unicast)
  Type: IPv4 (0x0800)
▼ Internet Protocol Version 4, Src: 192.168.78.128, Dst: 8.8.8.8
  0100 .... = Version: 4
  .... 0101 = Header Length: 20 bytes (5)
  ▶ Differentiated Services Field: 0x00 (DSCP: CS0, ECN: Not-ECT)
  Total Length: 84
  Identification: 0x22f1 (8945)
  ▼ Flags: 0x4000, Don't fragment
    0... .. = Reserved bit: Not set
    .1... .. = Don't fragment: Set
    0... .. = More fragments: Not set
  0900 00 50 56 eb 58 26 00 0c 29 46 62 0b 00 00 45 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
  0910 00 54 22 f1 40 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
  0920 00 00 00 00 ed f1 eb 17 00 05 49 8d 13 5e 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
  0930 00 00 f8 32 0b 00 00 00 00 00 00 10 11 12 13 14 15 16 17 18 19 1a 1b 1c 1d 1e 1f 20 21 22 23 24 25 26 27 28 29 2a 2b 2c 2d 2e 2f 30 31 32 33 34 35 36 37
  0940 16 17 18 19 1a 1b 1c 1d 1e 1f 20 21 22 23 24 25 26 27 28 29 2a 2b 2c 2d 2e 2f 30 31 32 33 34 35 36 37
  0950 26 27 28 29 2a 2b 2c 2d 2e 2f 30 31 32 33 34 35 36 37
  0960 36 37
```

Wireshark



Follow TCP Stream



Wireshark interface showing packet capture data. The packet list on the left shows several packets, with packet 19 selected. The packet details pane on the right shows the structure of the selected packet, including Ethernet II, Internet Protocol Version 4, and Hypertext Transfer Protocol (HTTP) fields. The packet bytes pane at the bottom shows the raw data in hexadecimal and ASCII.

Wireshark · Follow TCP Stream (tcp.stream eq 0) · taidoor_traffic_no_inte... - □ ×

```
GET /process.jsp?
mn=IOEHPJEALJEPFEDJDFMBLNHDBAFJCIECPOMOHNFKIPNMJIFBGHGLJJOAMCDBDBKBFPEONMJAFKMNKB
GGJOPKHJPJOGGLPGBDNCKIOBDFOLKAODLKLBDLFLKFOHABGKICDPNNAOGHBDHCGIGBIPBHLHCHIKKOHAI
IFCAOHGNLDNKPBLEAHKAFOLHLPGBFOHIFDKNNCOGNHPDHIHLABCKMMBCGOMBEIBAPHJIHGOCBHBBOGJHF
ENJNILPMA HTTP/1.1
Accept: */*
Connection: Keep-Alive
Cache-Control: no-cache
User-Agent: Mozilla/4.0 (compatible; MSIE 8.0; Windows NT 5.1; Trident/4.0; .NET
CLR 2.0.50727; .NET CLR 3.0.4506.2152; .NET CLR 3.5.30729)
Host: 104.248.234.238
```

```
HTTP/1.1 200 OK
Date: Tue, 24 Dec 2019 15:10:02 GMT
Server: Microsoft-IIS/5.0
Content-Type: text/html
Connection: close
Content-Length: 110
```

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 3.2 Final//EN>
<html dir=ltr>
<head>
<style>

</style>
</head>
</html>
```

Red == Request
Blue == Response



Statistics > Endpoints



Wireshark interface showing the 'Statistics > Endpoints' view. The main window displays a table of network endpoints for the file 'taidoor_traffic_no_interaction.pcap'.

Wireshark · Endpoints · taidoor_traffic_no_interaction.pcap

Ethernet · 3	IPv4 · 14	IPv6	TCP · 132	UDP · 26						
Address	Packets	Bytes	Tx Packets	Tx Bytes	Rx Packets	Rx Bytes	Country	City	AS Number	AS Organization
4.2.2.2	12	1,537	6	1,032	6	505	—	—	—	—
8.240.119.254	18	2,897	9	1,549	9	1,348	—	—	—	—
10.0.2.15	1,635	271 k	777	118 k	858	153 k	—	—	—	—
10.0.2.255	5	1,215	0	0	5	1,215	—	—	—	—
10.70.0.1	48	5,801	24	3,833	24	1,968	—	—	—	—
13.68.92.143	54	18 k	26	13 k	28	4,910	—	—	—	—
13.107.5.88	32	9,641	17	7,740	15	1,901	—	—	—	—
13.107.21.209	32	12 k	18	9,671	14	3,137	—	—	—	—
23.0.153.104	30	11 k	16	10 k	14	1,206	—	—	—	—
51.143.106.177	25	5,928	14	4,736	11	1,192	—	—	—	—
52.113.194.131	25	9,816	13	8,064	12	1,752	—	—	—	—
52.179.129.229	114	37 k	59	27 k	55	9,832	—	—	—	—
52.230.222.68	8	882	4	468	4	414	—	—	—	—
104.248.234.238	1,232	154 k	652	65 k	580	88 k	—	—	—	—

Additional statistics shown in the left pane:

- Frame 11: 66 bytes on wire (528 bits)
- Ethernet II, Src: PcsCompu_af:09:1e, Destination: RealtekU_17:35:02 (f)
- Address: RealtekU_12:35:02 (5)
- Source: PcsCompu_af:09:1e (00:00)
- Address: PcsCompu_af:09:1e (00:00)
- Type: IPv4 (0x0800)
- Internet Protocol Version 4, Src: 5.8.8, Version: 4
- Differentiated Services Field: 0
- Identification: 0x6666 (1766)
- Flags: 0x4000, Don't Fragment

Endpoint Types: ☐ Name resolution ☐ Limit to display filter

Buttons: Copy, Close, Help

Statistics > Conversations

Wireshark - Conversations - taidoor_traffic_no_interaction.pcap

Ethernet 2 IPv4 13 IPv6 TCP 122 UDP 29

Address A	Port A	Address B	Port B	Packets	Bytes	Packets A → B	Bytes A → B	Packets B → A	Bytes B → A	Rel Start	Duration	Bits/s A → B
10.0.2.15	49725	104.248.234.238	80	12	1,457	6	851	6	606	0.000000	0.3364	
10.0.2.15	49726	104.248.234.238	80	11	1,403	5	797	6	606	33.219100	0.2974	
10.0.2.15	49727	104.248.234.238	80	11	1,403	5	797	6	606	70.221079	0.3203	
10.0.2.15	49728	104.248.234.238	80	11	1,403	5	797	6	606	110.250334	0.3082	
10.0.2.15	49729	104.248.234.238	80	12	1,457	6	851	6	606	144.140630	0.3567	
10.0.2.15	49730	23.0.153.104	80	15	5,661	7	603	8	5,058	145.101599	95.4669	
10.0.2.15	49731	52.179.129.229	443	38	10 k	19	3,371	19	7,467	164.354640	13.6087	
10.0.2.15	49732	13.68.92.143	443	27	9,007	14	2,456	13	6,551	165.433315	12.5307	
10.0.2.15	49733	104.248.234.238	80	11	1,403	5	797	6	606	178.699405	0.2926	
10.0.2.15	49734	104.248.234.238	80	12	1,457	6	851	6	606	212.911576	0.3167	
10.0.2.15	49735	104.248.234.238	80	11	1,403	5	797	6	606	249.282086	0.3812	
10.0.2.15	49736	104.248.234.238	80	11	1,403	5	797	6	606	282.469902	0.2920	
10.0.2.15	49737	51.143.106.177	443	25	5,928	11	1,192	14	4,736	285.859695	126.2107	
10.0.2.15	49738	13.107.5.88	443	32	9,641	15	1,901	17	7,740	288.900134	128.7050	
10.0.2.15	49739	104.248.234.238	80	11	1,403	5	797	6	606	318.470030	0.2934	
10.0.2.15	49740	104.248.234.238	80	11	1,403	5	797	6	606	349.767351	0.2919	
10.0.2.15	49741	104.248.234.238	80	11	1,403	5	797	6	606	377.829367	0.2848	
10.0.2.15	49742	104.248.234.238	80	11	1,403	5	797	6	606	405.970345	0.3130	
10.0.2.15	49743	104.248.234.238	80	11	1,403	5	797	6	606	438.953512	0.3448	
10.0.2.15	49744	104.248.234.238	80	11	1,403	5	797	6	606	466.563935	0.2848	
10.0.2.15	49745	104.248.234.238	80	11	1,403	5	797	6	606	494.126398	0.2807	
10.0.2.15	49746	104.248.234.238	80	12	1,457	6	851	6	606	527.407759	0.3055	
10.0.2.15	49747	104.248.234.238	80	11	1,403	5	797	6	606	560.874916	0.3173	
10.0.2.15	49748	104.248.234.238	80	11	1,403	5	797	6	606	589.905672	0.3682	
10.0.2.15	49749	104.248.234.238	80	11	1,403	5	797	6	606	623.938345	0.3094	
10.0.2.15	49750	104.248.234.238	80	11	1,403	5	797	6	606	663.218650	0.2809	
10.0.2.15	49751	104.248.234.238	80	11	1,403	5	797	6	606	698.672924	0.2814	
10.0.2.15	49752	104.248.234.238	80	11	1,403	5	797	6	606	725.799040	0.2940	
10.0.2.15	49753	104.248.234.238	80	11	1,403	5	797	6	606	752.202497	0.2878	
10.0.2.15	49754	104.248.234.238	80	12	1,457	6	851	6	606	788.374833	0.3169	

☐ Name resolution ☐ Limit to display filter ☐ Absolute start time

Conversation Types

Copy Follow Stream... Graph... Close Help

Statistics > Protocol Hierarchy



The image shows the Wireshark Protocol Hierarchy Statistics window for a capture file named 'taidoor_traffic_no_interaction.pcap'. The window displays a hierarchical tree of protocols and their corresponding statistics.

Protocol	Percent Packets	Packets	Percent Bytes	Bytes
Frame	100.0	1645	100.0	272,000
Ethernet	100.0	1645	8.5	230,000
Internet Protocol Version 4	99.4	1635	12.0	320,000
User Datagram Protocol	4.0	65	0.2	520
NetBIOS Datagram Service	0.3	5	0.4	100
SMB (Server Message Block Protocol)	0.3	5	0.2	595
SMB MailSlot Protocol	0.3	5	0.0	125
Microsoft Windows Browser Protocol	0.3	5	0.1	165
Domain Name System	3.6	60	1.8	480
Transmission Control Protocol	95.4	1570	76.1	200,000
Secure Sockets Layer	5.7	94	28.7	785
Hypertext Transfer Protocol	13.9	228	35.2	955
Line-based text data	6.6	109	4.4	115
eXtensible Markup Language	0.1	2	3.1	855
Address Resolution Protocol	0.6	10	0.1	280

The left pane shows the packet list with details for the selected packet (Frame 1). The bottom pane shows the packet bytes pane with hex and ASCII data.

Statistics > HTTP > Requests



The image displays the Wireshark network protocol analyzer interface. The main window shows a packet list on the left, a packet details pane in the middle, and a packet bytes pane at the bottom. The 'Statistics' menu is open, and the 'HTTP' section is selected. The 'Requests' sub-menu is also open, showing a list of HTTP requests. The 'Display filter' field at the bottom is empty, and the 'Apply' button is visible.

Wireshark - Requests - taidoor_traffic_no_interaction.pcap

Topic / Item

- HTTP Requests by HTTP Host
 - tile-service.weather.microsoft.com
 - /en-US/livetile/preinstall?region=US&appid=C98EA5B0842DBB9405BBF071E1DA76512D21FE36&FORM=Threshold
 - ctdl.windowsupdate.com
 - /msdownload/update/v3/static/trustedr/en/pinrulesstl.cab?13db202675b3f464
 - /msdownload/update/v3/static/trustedr/en/disallowedcertstl.cab?b56e63a9af45dd80
 - /msdownload/update/v3/static/trustedr/en/authrootstl.cab?aac643d5e8c41bf9
 - 104.248.234.238
 - /process.jsp?mn=IOEHPJEAJEPFPEDJDFMBLNHDBAFJCIECPOMOHMNFKIPNMJIFBGHGLJJOAMCDBDBKBFPEONMJAFKMNKB

Display filter: Enter a display filter ...

Copy Save as... Close



LAB: Wireshark



© Black Hills Information Security | @BHInfoSecurity