Liên kết khác k3751070033@gmail.com Bảng điều khiển Đăng xuất

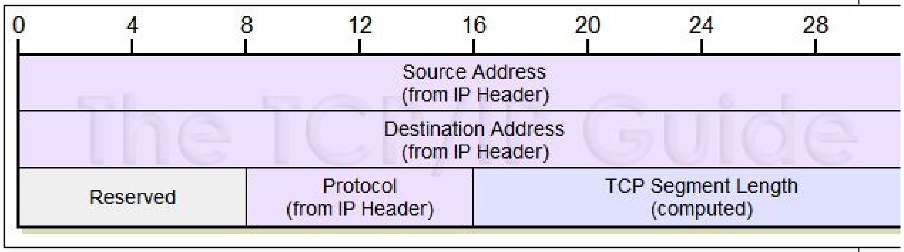
Learning by practicing

Learning is an ongoing activity ... practicing makes it fun

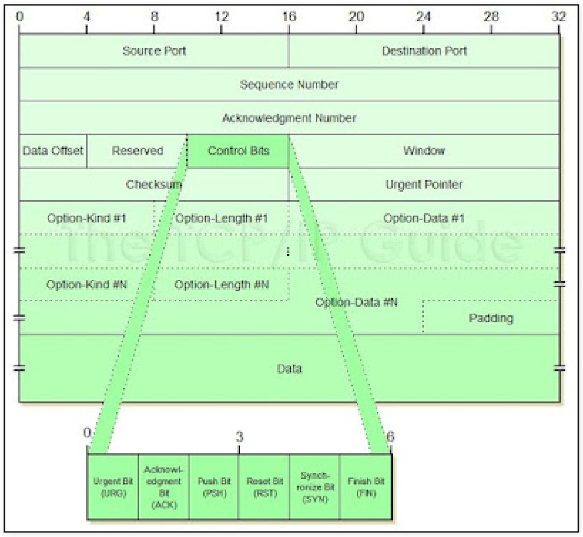
Monday, August 3, 2015

Calculating the TCP Checksum, with a taste of scapy + Wireshark

In this post we will calculate the TCP checksum. To calculate the TCP checksum we first must understand that in addition to its own header, TCP checksum uses a pseudo header. This pseudo header consists of the original source IP, destination IP, reserved (identified as 0000 0000), protocol (x06) and the length from the TCP header.



TCP pseudo header: reprinted with permission from tcpipguide.com



TCP Header: reprinted with permission from tcpipguide.com

Contributors

- Abdul
- Nik Alleyne, MSc | CISSP | GC|IA|IH|REM|PEN

Keep up-to-date with the latest posts
Enter your email address:

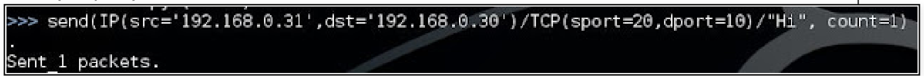
Subscribe

Delivered by FeedBurner

My Book Resume, SANS GCIA Gold Paper, etc

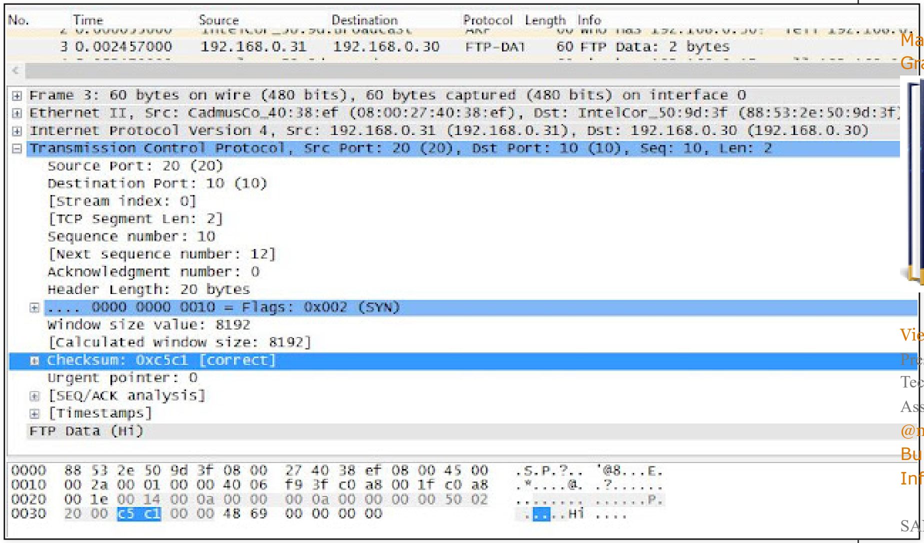
Grab a copy of Hack and Detect from Amazon. Read the sample chapters.

Considering the above, let us craft a TCP Packet in scapy. We have the following
Source IP = 192.168.0.31
Destination = 192.168.0.30
TCP source port = 20
TCP destination port = 10
Data (2 bytes) = “Hi”



```
>>> send(IP(src='192.168.0.31',dst='192.168.0.30')/TCP(sport=20,dport=10)/"Hi", count=1)
.
Sent 1 packets.
```

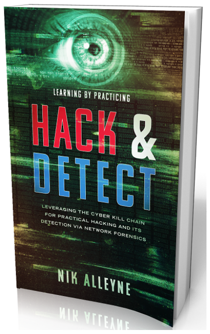
Let’s see what the receiving host got from a wireshark perspective



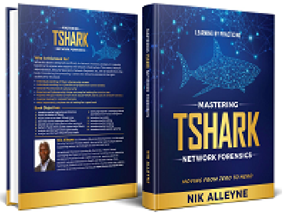
```
No.   Time          Source         Destination    Protocol  Length  Info
    3 0.002457000   192.168.0.31   192.168.0.30   FTP-DAT       60  FTP Data: 2 bytes

⊞ Frame 3: 60 bytes on wire (480 bits), 60 bytes captured (480 bits) on interface 0
⊞ Ethernet II, Src: CadmusCo_40:38:ef (08:00:27:40:38:ef), Dst: IntelCor_50:9d:3f (88:53:2e:50:9d:3f)
⊞ Internet Protocol Version 4, Src: 192.168.0.31 (192.168.0.31), Dst: 192.168.0.30 (192.168.0.30)
⊟ Transmission Control Protocol, Src Port: 20 (20), Dst Port: 10 (10), Seq: 10, Len: 2
    Source Port: 20 (20)
    Destination Port: 10 (10)
    [Stream index: 0]
    [TCP Segment Len: 2]
    Sequence number: 10
    [Next sequence number: 12]
    Acknowledgment number: 0
    Header Length: 20 bytes
  ⊞ .... 0000 0000 0010 = Flags: 0x002 (SYN)
    Window size value: 8192
    [Calculated window size: 8192]
  ⊞ Checksum: 0xc5c1 [correct]
    Urgent pointer: 0
  ⊞ [SEQ/ACK analysis]
  ⊞ [Timestamps]
  FTP Data (Hi)

0000  88 53 2e 50 9d 3f 08 00  27 40 38 ef 08 00 45 00   .S.P.?.. '@8...E.
0010  00 2a 00 01 00 00 40 06  f9 3f c0 a8 00 1f c0 a8   .*....@. .?......
0020  00 1e 00 14 00 0a 00 00  00 0a 00 00 00 00 50 02   ........ ......P.
0030  20 00 c5 c1 00 00 48 69  00 00 00 00               ......Hi ....
```

Note from the above image wireshark has already computed the TCP checksum for us. Now let’s try to see if we can get the same value as wireshark.

So from the information we have, we can go ahead and build out pseudo header. Also when adding, these values needs to be added 16 bits or 2 bytes at a time.


LEARNING BY PRACTICING
HACK & DETECT
LEVERAGING THE CYBER KILL CHAIN FOR PRACTICAL HACKING AND ITS DETECTION VIA NETWORK FORENSICS
NIK ALLEYNE

Mastering TShark Network Forensics. Grab a copy: Get the Sample Chapters



View a PDF version of my resume
Presentation to the Ottawa Chapter of the High Technology Crimes Investigation (HTCIA) Association & ISC2 Toronto Chapter, SANS @night, etc
Building a Forensically Capable Network Infrastructure

SANS Webcast
The Importance of Intrusion Detection in a compromise prone world

Presentation at Canada International Cyber Security Conference:
Threat Analysis and Investigative Techniques in the Modern world

SANS GCIA Gold Paper
Building a Forensically Capable Network Infrastructure

Github Project QRadar ThreatIntel
Qradar Threat Intel on the Cheap

Python/Scapy DNS Spoofing Tool/Tutorial

Masters Capstone Paper
FROM PAPERLESS TO PLASTICLESS, EMV CARD SECURITY AND THE FUTURE OF PAYMENTS IN THE USA

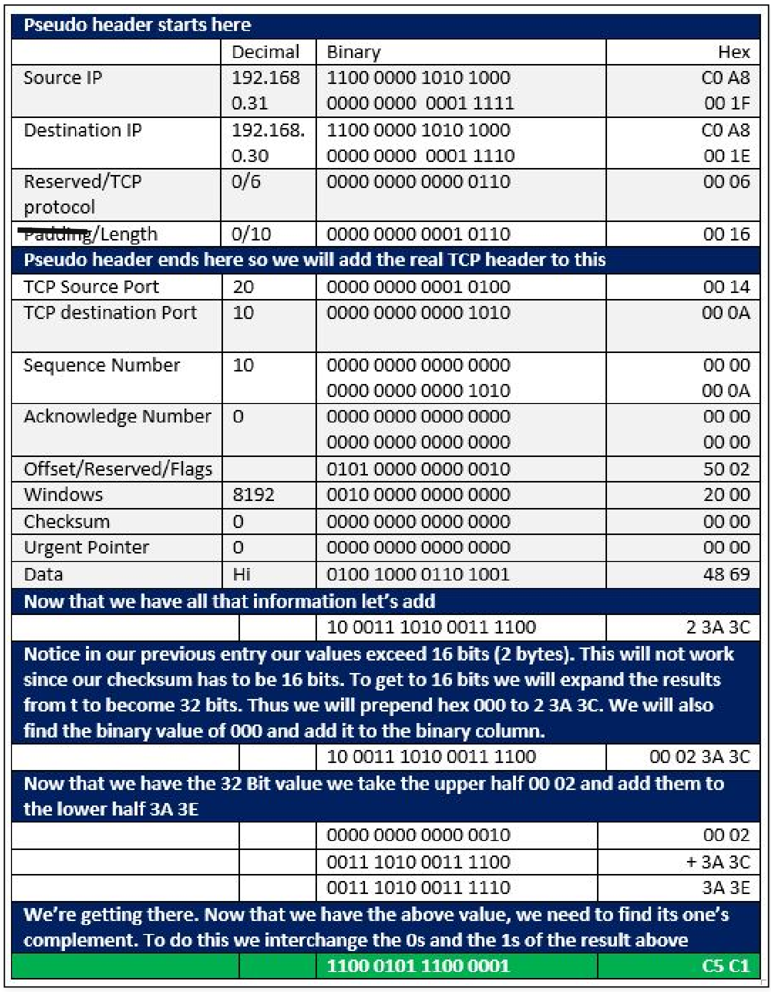
asecure.cloud - Build a Secure Cloud
A free repository of customizable AWS security configurations and best practices

Generate custom CloudFormation and CLI deployment scripts by adding multiple configuration items to stacks go get it **here**

My GITHub

Blog Archive

- ► 2023 (3)
- ► 2022 (16)
- ► 2021 (32)
- ► 2020 (39)
- ► 2019 (27)
- ► 2018 (24)



Pseudo header starts here			
	Decimal	Binary	Hex
Source IP	192.168 0.31	1100 0000 1010 1000 0000 0000 0001 1111	C0 A8 00 1F
Destination IP	192.168. 0.30	1100 0000 1010 1000 0000 0000 0001 1110	C0 A8 00 1E
Reserved/TCP protocol	0/6	0000 0000 0000 0110	00 06
~~Padding~~/Length	0/10	0000 0000 0001 0110	00 16
Pseudo header ends here so we will add the real TCP header to this			
TCP Source Port	20	0000 0000 0001 0100	00 14
TCP destination Port	10	0000 0000 0000 1010	00 0A
Sequence Number	10	0000 0000 0000 0000 0000 0000 0000 1010	00 00 00 0A
Acknowledge Number	0	0000 0000 0000 0000 0000 0000 0000 0000	00 00 00 00
Offset/Reserved/Flags		0101 0000 0000 0010	50 02
Windows	8192	0010 0000 0000 0000	20 00
Checksum	0	0000 0000 0000 0000	00 00
Urgent Pointer	0	0000 0000 0000 0000	00 00
Data	Hi	0100 1000 0110 1001	48 69
Now that we have all that information let's add			
		10 0011 1010 0011 1100	2 3A 3C
Notice in our previous entry our values exceed 16 bits (2 bytes). This will not work since our checksum has to be 16 bits. To get to 16 bits we will expand the results from t to become 32 bits. Thus we will prepend hex 000 to 2 3A 3C. We will also find the binary value of 000 and add it to the binary column.			
		10 0011 1010 0011 1100	00 02 3A 3C
Now that we have the 32 Bit value we take the upper half 00 02 and add them to the lower half 3A 3E			
		0000 0000 0000 0010	00 02
		0011 1010 0011 1100	+ 3A 3C
		0011 1010 0011 1110	3A 3E
We're getting there. Now that we have the above value, we need to find its one's complement. To do this we interchange the 0s and the 1s of the result above			
		1100 0101 1100 0001	C5 C1

22 Byte

That's it our TCP Checksum is 0XC5C1 which matches what wireshark provided us above.
Hope this helps someone who wanted to know how to calculate the TCP Checksum
References:

http://www.tcpipguide.com/free/t_TCPChecksumCalculationandtheTCPPseudoHeader-2.htm
http://www.tcpipguide.com/free/t_TCPMessageSegmentFormat-3.htm
http://www.secdev.org/projects/scapy/
https://www.wireshark.org/

Posted by Nik Alleyne, MSc | CISSP | GC|IA|IH|REM|PEN at 8:41 PM

2 comments:

amirm December 7, 2018 at 1:52 PM

Very nice! However there seems to be a typo there....
If "Padding/Length" is "0/10", the HEX value cannot be 0x0016.
I tried checksum calculation with 0x0016 and it gave 0xC5C1 as result.
Therefore "Padding/Length" should be "22", which is the right value (20 bytes TCP header + 2 bytes data).

Reply

Replies

Nik Alleyne, MSc | CISSP | GC|IA|IH|REM|PEN December 8, 2018 at 9:38 AM

- ► 2017 (64)
- ► 2016 (42)
- ▼ 2015 (86)
  - ► December (5)
  - ► November (11)
  - ► October (4)
  - ► September (3)
  - ▼ August (7)
    - Windows 10 Recycle Bin Analys - RIFIUTI2
    - Calculating the TCP Checksum, with a taste of scap...
    - Calculating the UDP Checksum, with a taste of scap...
    - Calculating the IP Checksum, with a taste of scapy...
    - Stimulus and response revisited
    - Integrating RSA 8.1 (SNMPv3 traps) with Splunk on ...
    - Metadata - Telling the story
  - ► July (6)
  - ► June (9)
  - ► May (7)
  - ► April (8)
  - ► March (9)
  - ► February (8)
  - ► January (9)
- ► 2014 (70)

Learning Sites:
http://www.securitytube.net
http://www.cybrary.it/
ENISA
Seed Labs
Open Security Training
Fuzzy Security
Honeynet Project
Corelan Exploiting Writing Tutorial
Mitre

Additional Readings
SANS Reading Room
https://www.us-cert.gov/
http://taosecurity.blogspot.ca/
http://krebsonsecurity.com/
http://securityweekly.com/
http://www.csoonline.com/blogs
https://securosis.com/blog/
http://threatpost.com/
http://nakedsecurity.sophos.com/
http://blog.zeltser.com/
https://www.schneier.com/
Morning Star Security
Infosec Industry

Intelligence Feeds - IPs/Domains/URLs

[MALICIOUS IPs]
Emerging Threats - Compromised IPs
My IPs
Spamhause drop
Spamhause edrop
Emerging Threats Block IP
DShield
SANS ISC
Zonefiles.io
SSL IP Blokclist
SSL IP Blokclist - Aggressive
Feedotracker - recommended ip blocklist
Feedotracker - IP Blocklist

Amirm,
You are spot on! Thanks for catching this. This was a typo on my part. If you notice, I have the correct binary values for "Padding/Length" as in "0000 0000 0001 0110". This value is indeed decimal "22" and Hex "0x0016", which represents the TCP header of 20 bytes and the data "HI" which is 2 bytes for the total of 22.

For the next reader of this post, I'm unable to change the value above as this is an image. However, the decimal for "Padding/Length" should be "0/22" and not "0/10" as I have it above.

Once again, thanks Amirm, for catching this.

More importantly, thanks for reading!

Reply

Enter Comment

Newer Post Home Older Post

Subscribe to: Post Comments (Atom)

Block IP

[MALICIOUS DOMAINS]
Malware Domains Delisted
OpenPhish
Hostnames
Domains
Domains

[MALICIOUS URLS]
vxvault.net URL_List

[PHISHING URLS]
openphish

Below is a list of threat intelligence websites that you can use. Cymon.io is an excellent one as it searches around 200 different sources. If you’re looking for a more exhaustive list of threat intel sites, check out https://github.com/rshipp/awesome-malware-analysis

IP and Domain Reputation / Malicious Activity Reports
http://cymon.io
https://www.recordedfuture.com/live/
http://urlquery.net/ (URL Scanner)
https://virustotal.com/
https://otx.alienvault.com/
https://exchange.xforce.ibmcloud.com/

IP Information (open ports, details, WHOIS, etc)
https://www.censys.io
https://www.shodan.io/
https://centralops.net/co/
http://viewdns.info/
https://www.threatcrowd.org

Malware Analysis
https://malwr.com/
https://www.hybrid-analysis.com/

Misc
https://isc.sans.edu/services.html (Port information)

Malware / Malicious Site Samples:
https://malwr.com/
http://vxvault.net/ViriList.php
http://cybercrime-tracker.net/
https://ransomwaretracker.abuse.ch/tracker/
http://malc0de.com/database/

OSINT Framework

Nik Alleyne (www.securitynik.com). Simple theme. Powered by Blogger.