Threat intelligence week 2

Jurjen de Jonge
500731921
Hogeschool van Amsterdam

September 23, 2018

Contents

1. *Netcat assignment*

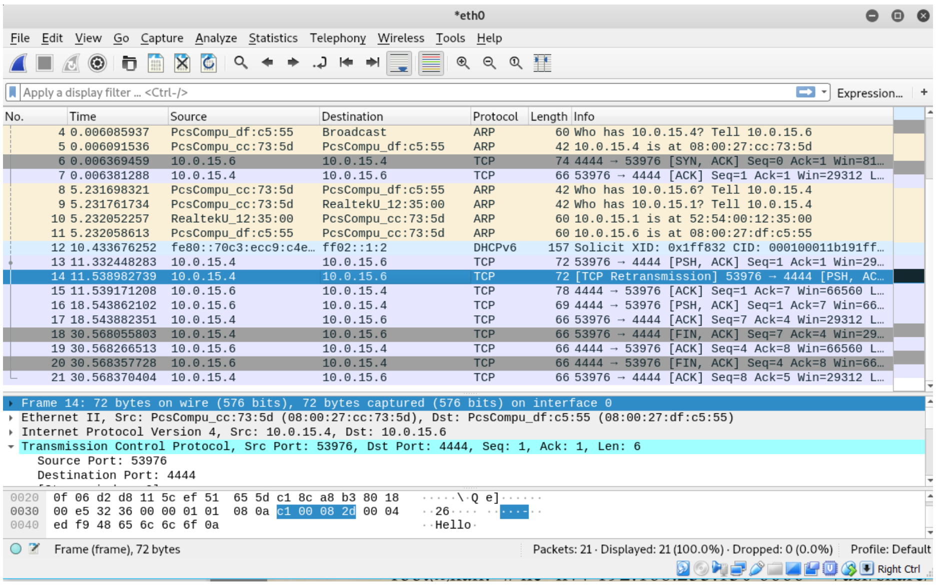
1.1 Banner grabbing SSH

Wireshark shows an TCP handshake, after the handshake has been completed it shows the SSH protocol sending it's banner information towards the Windows computer that executed the netcat command.

1.2 Setting up a chat relay

On windows the netcat will be listening to incoming connections on port 4444. After establishing the connection with netcat on the Kali linux machine it shows that a connection has been made from the ip address of the Kali linux machine (10.0.15.4) Then after typing Hello in the terminal and hitting the enter key the message showed up at the Windows machine. The same goes for when typing ”hi” in the windows machine.

In Wireshark it first shows a handshake between the Kali and Windows machine. Then the Kali machine sends the Hello message which gets acknowledged by the Windows machine. The same scenario for the Hi message from the Windows message towards the kali machine.

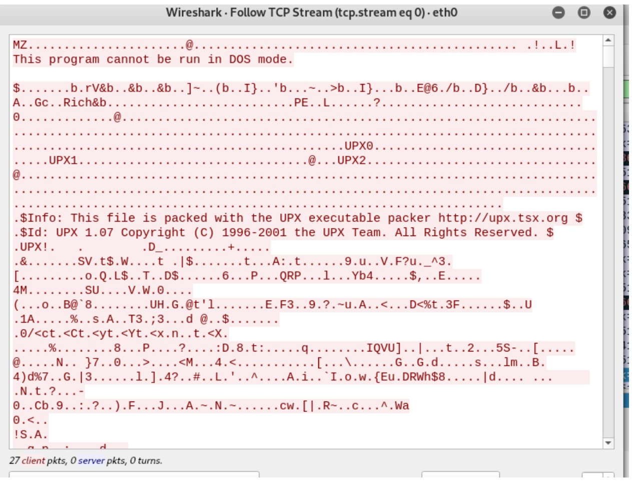


```
No.    Time           Source                 Destination           Protocol Length Info
     4 0.006085937    PcsCompu_df:c5:55      Broadcast             ARP        60 Who has 10.0.15.4? Tell 10.0.15.6
     5 0.006091536    PcsCompu_cc:73:5d      PcsCompu_df:c5:55     ARP        42 10.0.15.4 is at 08:00:27:cc:73:5d
     6 0.006369459    10.0.15.6              10.0.15.4             TCP        74 4444 → 53976 [SYN, ACK] Seq=0 Ack=1 Win=81…
     7 0.006381288    10.0.15.4              10.0.15.6             TCP        66 53976 → 4444 [ACK] Seq=1 Ack=1 Win=29312 L…
     8 5.231698321    PcsCompu_cc:73:5d      PcsCompu_df:c5:55     ARP        42 Who has 10.0.15.6? Tell 10.0.15.4
     9 5.231761734    PcsCompu_cc:73:5d      RealtekU_12:35:00     ARP        42 Who has 10.0.15.1? Tell 10.0.15.4
    10 5.232052257    RealtekU_12:35:00      PcsCompu_cc:73:5d     ARP        60 10.0.15.1 is at 52:54:00:12:35:00
    11 5.232058613    PcsCompu_df:c5:55      PcsCompu_cc:73:5d     ARP        60 10.0.15.6 is at 08:00:27:df:c5:55
    12 10.433676252   fe80::70c3:ecc9:c4e…   ff02::1:2             DHCPv6    157 Solicit XID: 0x1ff832 CID: 000100011b191ff…
    13 11.332448283   10.0.15.4              10.0.15.6             TCP        72 53976 → 4444 [PSH, ACK] Seq=1 Ack=1 Win=29…
    14 11.538982739   10.0.15.4              10.0.15.6             TCP        72 [TCP Retransmission] 53976 → 4444 [PSH, AC…
    15 11.539171208   10.0.15.6              10.0.15.4             TCP        78 4444 → 53976 [ACK] Seq=1 Ack=7 Win=66560 L…
    16 18.543862102   10.0.15.6              10.0.15.4             TCP        69 4444 → 53976 [PSH, ACK] Seq=1 Ack=7 Win=66…
    17 18.543882351   10.0.15.4              10.0.15.6             TCP        66 53976 → 4444 [ACK] Seq=7 Ack=4 Win=29312 L…
    18 30.568055803   10.0.15.4              10.0.15.6             TCP        66 53976 → 4444 [FIN, ACK] Seq=7 Ack=4 Win=29…
    19 30.568266513   10.0.15.6              10.0.15.4             TCP        66 4444 → 53976 [ACK] Seq=4 Ack=8 Win=66560 L…
    20 30.568357728   10.0.15.6              10.0.15.4             TCP        66 4444 → 53976 [FIN, ACK] Seq=4 Ack=8 Win=66…
    21 30.568370404   10.0.15.4              10.0.15.6             TCP        66 53976 → 4444 [ACK] Seq=8 Ack=5 Win=29312 L…

▸ Frame 14: 72 bytes on wire (576 bits), 72 bytes captured (576 bits) on interface 0
▸ Ethernet II, Src: PcsCompu_cc:73:5d (08:00:27:cc:73:5d), Dst: PcsCompu_df:c5:55 (08:00:27:df:c5:55)
▸ Internet Protocol Version 4, Src: 10.0.15.4, Dst: 10.0.15.6
▾ Transmission Control Protocol, Src Port: 53976, Dst Port: 4444, Seq: 1, Ack: 1, Len: 6
    Source Port: 53976
    Destination Port: 4444

0020  0f 06 d2 d8 11 5c ef 51  65 5d c1 8c a8 b3 80 18   ·····\·Q e]······
0030  00 e5 32 36 00 00 01 01  08 0a c1 00 08 2d 00 04   ··26···· ·····-··
0040  ed f9 48 65 6c 6c 6f 0a                            ··Hello·

Frame (frame), 72 bytes          Packets: 21 · Displayed: 21 (100.0%) · Dropped: 0 (0.0%)   Profile: Default
```

*eth0
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
Apply a display filter ... <Ctrl-/>
Expression...
+
Right Ctrl

1.3 File transfer using netcat

After launching netcat on the Kali machine a large amount of TCP packets appear in Wireshark. When following these packets in Wireshark it becomes clear that there has been a transfer of a Windows executable by the MZ keyword in the beginning. In windows it shows that a connection has established from the Kali IP address



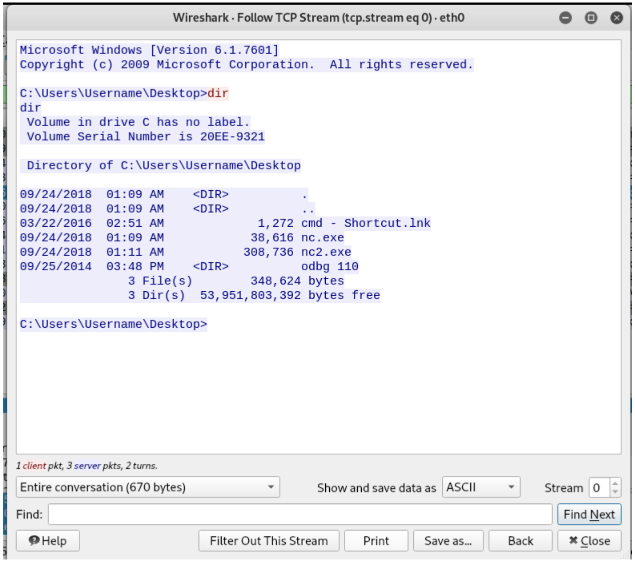
```
MZ......................@............................................ .!..L.!
This program cannot be run in DOS mode.

$.......b.rV&b..&b..&b..]~..(b..I}..'b...~..>b..I}...b..E@6./b..D}../b..&b...b..
A..Gc..Rich&b.........................PE..L......?..........................
0.............@.................................................................
................................................................................
............................................UPX0................................
.....UPX1..............................@...UPX2.................................
@...............................................................................
................................................................................
..................................................................
.$Info: This file is packed with the UPX executable packer http://upx.tsx.org $
.$Id: UPX 1.07 Copyright (C) 1996-2001 the UPX Team. All Rights Reserved. $
.UPX!.   .        .D_.........+.....
.&.......SV.t$.W....t .|$.......t...A:.t......9.u..V.F?u._^3.
[.........o.Q.L$..T..D$......6...P...QRP...l...Yb4.....$,..E.....
4M........SU....V.W.0....
(...o..B@`8........UH.G.@t'l.......E.F3..9.?.~u.A..<...D<%t.3F......$..U
.1A.....%..s.A..T3.;3...d @..$.......
.0/<ct.<Ct.<yt.<Yt.<x.n..t.<X.
.....%........8...P....?....:D.8.t:.....q........IQVU]..|...t..2...5S-..[.....
@.....N.. }7..0...>....<M...4.<...........[...\......G..G.d.....s...lm..B.
4)d%7..G.|3......l.].4?..#..L.'..^....A.i..`I.o.w.{Eu.DRWh$8.....|d.... ...
.N.t.?...-
0..Cb.9..:.?..).F...J...A.~.N.~......cw.[|.R~..c...^.Wa
0.<..
!S.A.
```

27 client pkts, 0 server pkts, 0 turns.

1.4 Bind shell using netcat

In the Kali terminal a Windows CMD shell appears. It’s possible to execute commands on the Windows machine. Wireshark shows the command and the text that is being transferred over the NC connection.

The pro about the bind shell is that the service is being hosted on the machine it self, no need to ping a c&c or have a server that can be taken down. The problem with it is that the machine needs to have a exposed / non firewalled port that can be binded.



```
Microsoft Windows [Version 6.1.7601]
Copyright (c) 2009 Microsoft Corporation.  All rights reserved.

C:\Users\Username\Desktop>dir
dir
 Volume in drive C has no label.
 Volume Serial Number is 20EE-9321

 Directory of C:\Users\Username\Desktop

09/24/2018  01:09 AM    <DIR>          .
09/24/2018  01:09 AM    <DIR>          ..
03/22/2016  02:51 AM             1,272 cmd - Shortcut.lnk
09/24/2018  01:09 AM            38,616 nc.exe
09/24/2018  01:11 AM           308,736 nc2.exe
09/25/2014  03:48 PM    <DIR>          odbg 110
               3 File(s)        348,624 bytes
               3 Dir(s)  53,951,803,392 bytes free

C:\Users\Username\Desktop>
```

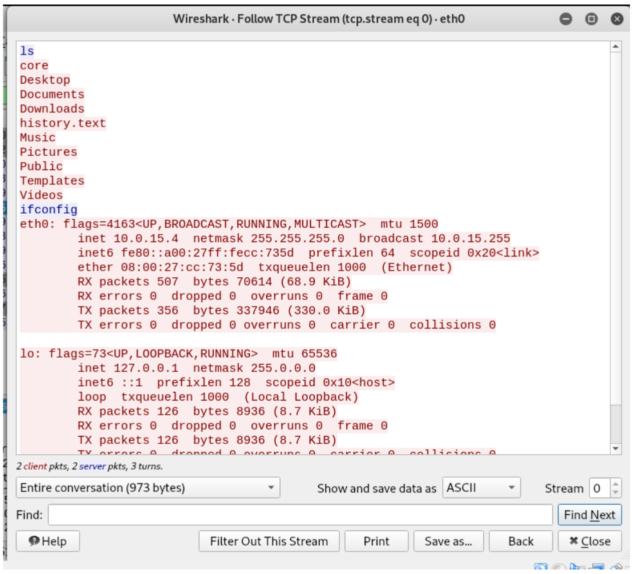
1 client pkt, 3 server pkts, 2 turns.

Entire conversation (670 bytes) — Show and save data as: ASCII — Stream: 0

1.5 Reverse shell using netcat

The command given in this part of the presentation seems off. I had to add -e /bin/bash to the command to get the reverse shell working.

After the reverse connection was made to the listing Windows machine I could type ls to see a list of files and directories. In wireshark it shows that the Linux machine is connecting to the Windows machine to over the TCP protocol.



```
ls
core
Desktop
Documents
Downloads
history.text
Music
Pictures
Public
Templates
Videos
ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 10.0.15.4  netmask 255.255.255.0  broadcast 10.0.15.255
        inet6 fe80::a00:27ff:fecc:735d  prefixlen 64  scopeid 0x20<link>
        ether 08:00:27:cc:73:5d  txqueuelen 1000  (Ethernet)
        RX packets 507  bytes 70614 (68.9 KiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 356  bytes 337946 (330.0 KiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING>  mtu 65536
        inet 127.0.0.1  netmask 255.0.0.0
        inet6 ::1  prefixlen 128  scopeid 0x10<host>
        loop  txqueuelen 1000  (Local Loopback)
        RX packets 126  bytes 8936 (8.7 KiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 126  bytes 8936 (8.7 KiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
```

2 client pkts, 2 server pkts, 3 turns.

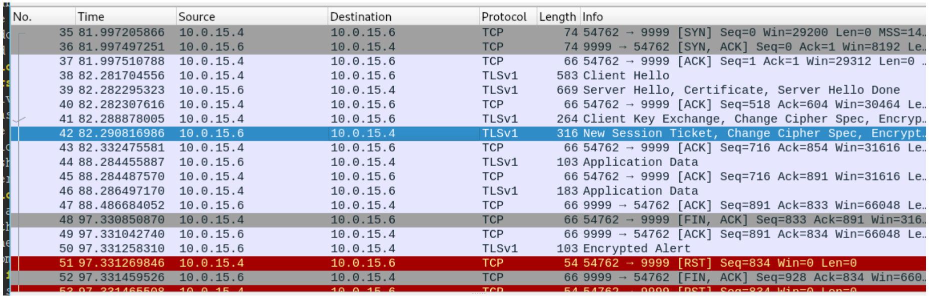
Entire conversation (973 bytes)   Show and save data as: ASCII   Stream: 0

Find:

The pro about a reverse shell is that the machine could be behind a firewall and the attacker is still able to access that computer. A big con is that there should always be a server listening to see if any connections come in.

1.6 ncat for ssl encrypted shell

Using ncat it's possible to encrypt the shell using ssl. After setting up the connection between Kali linux towards windows it and typing "ls" in the CMD prompt it show the list of files and folders.
But now looking at Wireshark it shows TLS traffic instead of plain text transfer of the commands and responses.



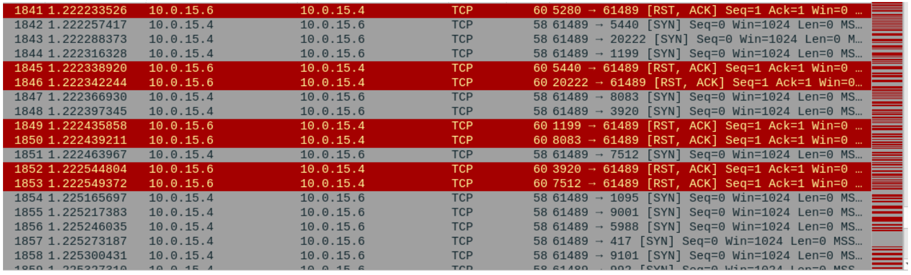
No.	Time	Source	Destination	Protocol	Length	Info
35	81.997205866	10.0.15.4	10.0.15.6	TCP	74	54762 → 9999 [SYN] Seq=0 Win=29200 Len=0 MSS=14…
36	81.997497251	10.0.15.6	10.0.15.4	TCP	74	9999 → 54762 [SYN, ACK] Seq=0 Ack=1 Win=8192 Le…
37	81.997510788	10.0.15.4	10.0.15.6	TCP	66	54762 → 9999 [ACK] Seq=1 Ack=1 Win=29312 Len=0 …
38	82.281704556	10.0.15.4	10.0.15.6	TLSv1	583	Client Hello
39	82.282295323	10.0.15.6	10.0.15.4	TLSv1	669	Server Hello, Certificate, Server Hello Done
40	82.282307616	10.0.15.4	10.0.15.6	TCP	66	54762 → 9999 [ACK] Seq=518 Ack=604 Win=30464 Le…
41	82.288878005	10.0.15.4	10.0.15.6	TLSv1	264	Client Key Exchange, Change Cipher Spec, Encryp…
42	82.290816986	10.0.15.6	10.0.15.4	TLSv1	316	New Session Ticket, Change Cipher Spec, Encrypt…
43	82.332475581	10.0.15.4	10.0.15.6	TCP	66	54762 → 9999 [ACK] Seq=716 Ack=854 Win=31616 Le…
44	88.284455887	10.0.15.6	10.0.15.4	TLSv1	103	Application Data
45	88.284487570	10.0.15.4	10.0.15.6	TCP	66	54762 → 9999 [ACK] Seq=716 Ack=891 Win=31616 Le…
46	88.286497170	10.0.15.4	10.0.15.6	TLSv1	183	Application Data
47	88.486684052	10.0.15.6	10.0.15.4	TCP	66	9999 → 54762 [ACK] Seq=891 Ack=833 Win=66048 Le…
48	97.330850870	10.0.15.4	10.0.15.6	TCP	66	54762 → 9999 [FIN, ACK] Seq=833 Ack=891 Win=316…
49	97.331042740	10.0.15.6	10.0.15.4	TCP	66	9999 → 54762 [ACK] Seq=891 Ack=834 Win=66048 Le…
50	97.331258310	10.0.15.6	10.0.15.4	TLSv1	103	Encrypted Alert
51	97.331269846	10.0.15.4	10.0.15.6	TCP	54	54762 → 9999 [RST] Seq=834 Win=0 Len=0
52	97.331459526	10.0.15.6	10.0.15.4	TCP	66	9999 → 54762 [FIN, ACK] Seq=928 Ack=834 Win=660…
53	97.331465508	10.0.15.4	10.0.15.6	TCP	54	54762 → 9999 [RST] Seq=834 Win=0 Len=0

The pro's of using ncat is that it supports all the features of the older netcat and extended that with SSL support. A con is that it doesn't come by default on all *unix machines. So the binary has to be downloaded or locally compiled.

2. *Nmap assignment*

2.1 Nmap scan of Windows computer

It looks like a total of almost 100.000 bytes have been sent over the network. This would surly be noticed if all the 65556 ports will be scanned. The speed would still be quick if it would be a local network. Over the internet it could take a lot of time to complete the full scan. A lot of failed handshakes are visible in Wireshark.



```
1841 1.222233526   10.0.15.6          10.0.15.4          TCP        60 5280 → 61489 [RST, ACK] Seq=1 Ack=1 Win=0 …
1842 1.222257417   10.0.15.4          10.0.15.6          TCP        58 61489 → 5440 [SYN] Seq=0 Win=1024 Len=0 MS…
1843 1.222288373   10.0.15.4          10.0.15.6          TCP        58 61489 → 20222 [SYN] Seq=0 Win=1024 Len=0 M…
1844 1.222316328   10.0.15.4          10.0.15.6          TCP        58 61489 → 1199 [SYN] Seq=0 Win=1024 Len=0 MS…
1845 1.222338920   10.0.15.6          10.0.15.4          TCP        60 5440 → 61489 [RST, ACK] Seq=1 Ack=1 Win=0 …
1846 1.222342244   10.0.15.6          10.0.15.4          TCP        60 20222 → 61489 [RST, ACK] Seq=1 Ack=1 Win=0…
1847 1.222366930   10.0.15.4          10.0.15.6          TCP        58 61489 → 8083 [SYN] Seq=0 Win=1024 Len=0 MS…
1848 1.222397345   10.0.15.4          10.0.15.6          TCP        58 61489 → 3920 [SYN] Seq=0 Win=1024 Len=0 MS…
1849 1.222435858   10.0.15.6          10.0.15.4          TCP        60 1199 → 61489 [RST, ACK] Seq=1 Ack=1 Win=0 …
1850 1.222439211   10.0.15.6          10.0.15.4          TCP        60 8083 → 61489 [RST, ACK] Seq=1 Ack=1 Win=0 …
1851 1.222463967   10.0.15.4          10.0.15.6          TCP        58 61489 → 7512 [SYN] Seq=0 Win=1024 Len=0 MS…
1852 1.222544804   10.0.15.6          10.0.15.4          TCP        60 3920 → 61489 [RST, ACK] Seq=1 Ack=1 Win=0 …
1853 1.222549372   10.0.15.6          10.0.15.4          TCP        60 7512 → 61489 [RST, ACK] Seq=1 Ack=1 Win=0 …
1854 1.225165697   10.0.15.4          10.0.15.6          TCP        58 61489 → 1095 [SYN] Seq=0 Win=1024 Len=0 MS…
1855 1.225217383   10.0.15.4          10.0.15.6          TCP        58 61489 → 9001 [SYN] Seq=0 Win=1024 Len=0 MS…
1856 1.225246035   10.0.15.4          10.0.15.6          TCP        58 61489 → 5988 [SYN] Seq=0 Win=1024 Len=0 MS…
1857 1.225273187   10.0.15.4          10.0.15.6          TCP        58 61489 → 417 [SYN] Seq=0 Win=1024 Len=0 MSS…
1858 1.225300431   10.0.15.4          10.0.15.6          TCP        58 61489 → 9101 [SYN] Seq=0 Win=1024 Len=0 MS…
```

2.2 Network sweeping

The first scan uses an arp broadcast to get a list of active IP addresses in the network.

The second scan shows that nmap tries to establish a tcp connection on port 80 which gets refused because the port is closed.

Scans for the top 20 ports, can be seen in Wireshark. Again accessing the ports using tcp.

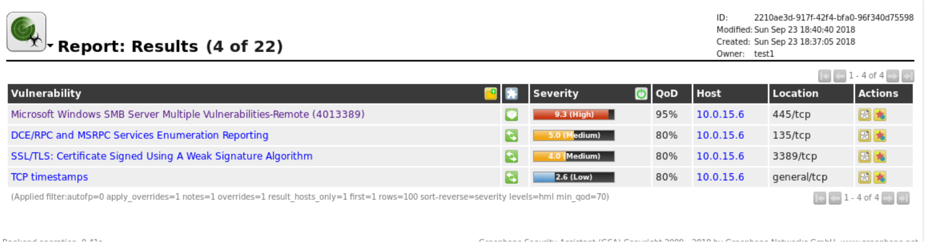
Version scanning generates a lot of traffic in Wireshark, you can see that different protocols are being tested on different ports. If it connects it tries to obtain information about the service.

OS finger print scan performs again a large port scan and then behaves a little funny to test what kind of operating system the scanned IP address could be using. I've read that this is possible by looking at the different time out between operating systems.

The result of performing all scans was a lot of traffic compared to all the other scans. Iptables told me that the scan was 221KB.This is huge compared to the 100.000 bytes in the beginning.

2.3 OpenVAS scan

The installation process of OpenVAS is easy on Kali linux. One high risk vulnerability has been detected by OpenVAS. This was in the SMB server of the Windows 7 installation. Multiple issues have been reported under several CVE's. In the worst case scenario this could lead to remote code execution (RCE)



Report: Results (4 of 22)

ID: 2210ae3d-917f-42f4-bfa0-96f340d75598
Modified: Sun Sep 23 18:40:40 2018
Created: Sun Sep 23 18:37:05 2018
Owner: test1

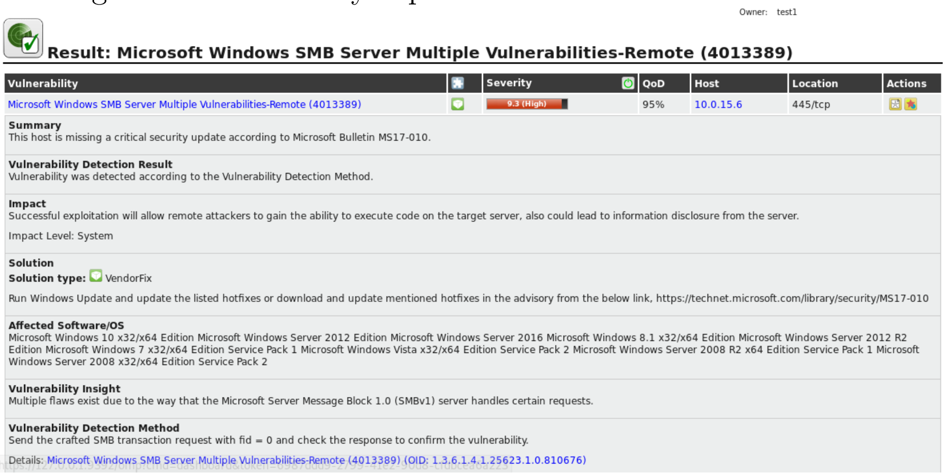
1 - 4 of 4

Vulnerability		Severity	QoD	Host	Location	Actions
Microsoft Windows SMB Server Multiple Vulnerabilities-Remote (4013389)		9.3 (High)	95%	10.0.15.6	445/tcp	
DCE/RPC and MSRPC Services Enumeration Reporting		5.0 (Medium)	80%	10.0.15.6	135/tcp	
SSL/TLS: Certificate Signed Using A Weak Signature Algorithm		4.0 (Medium)	80%	10.0.15.6	3389/tcp	
TCP timestamps		2.6 (Low)	80%	10.0.15.6	general/tcp	

(Applied filter:autofp=0 apply_overrides=1 notes=1 overrides=1 result_hosts_only=1 first=1 rows=100 sort-reverse=severity levels=hml min_qod=70)

1 - 4 of 4

The High risk vulnerability explained



Owner: test1

Result: Microsoft Windows SMB Server Multiple Vulnerabilities-Remote (4013389)

Vulnerability		Severity	QoD	Host	Location	Actions
Microsoft Windows SMB Server Multiple Vulnerabilities-Remote (4013389)		9.3 (High)	95%	10.0.15.6	445/tcp	

Summary
This host is missing a critical security update according to Microsoft Bulletin MS17-010.

Vulnerability Detection Result
Vulnerability was detected according to the Vulnerability Detection Method.

Impact
Successful exploitation will allow remote attackers to gain the ability to execute code on the target server, also could lead to information disclosure from the server.

Impact Level: System

Solution
Solution type: VendorFix

Run Windows Update and update the listed hotfixes or download and update mentioned hotfixes in the advisory from the below link, https://technet.microsoft.com/library/security/MS17-010

Affected Software/OS
Microsoft Windows 10 x32/x64 Edition Microsoft Windows Server 2012 Edition Microsoft Windows Server 2016 Microsoft Windows 8.1 x32/x64 Edition Microsoft Windows Server 2012 R2 Edition Microsoft Windows 7 x32/x64 Edition Service Pack 1 Microsoft Windows Vista x32/x64 Edition Service Pack 2 Microsoft Windows Server 2008 R2 x64 Edition Service Pack 1 Microsoft Windows Server 2008 x32/x64 Edition Service Pack 2

Vulnerability Insight
Multiple flaws exist due to the way that the Microsoft Server Message Block 1.0 (SMBv1) server handles certain requests.

Vulnerability Detection Method
Send the crafted SMB transaction request with fid = 0 and check the response to confirm the vulnerability.

Details: Microsoft Windows SMB Server Multiple Vulnerabilities-Remote (4013389) (OID: 1.3.6.1.4.1.25623.1.0.810676)