



---

**ECE - 567**

---

**PROJECT REPORT**

**Part 2**



**APRIL 9, 2020**  
**SUBMITTED TO:**  
**Professor Issa Traore**

**SUBMITTED BY:**  
**SANNATH REDDY VEMULA - V00949217**  
**SOMAYEH ROSHANDEL - V00942553**  
**ANJALI - V00037453**

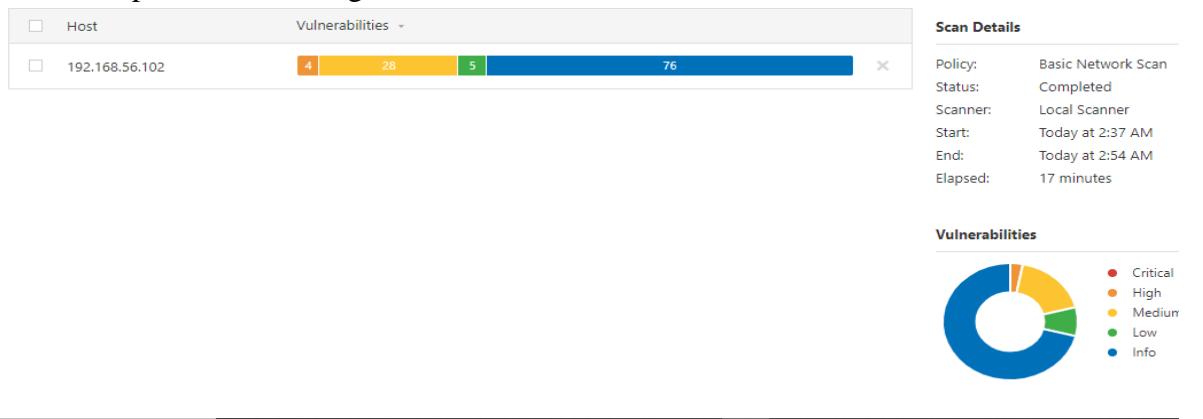
## Defense Strategies

In the second part of the project, we will use the attack intelligence obtained in part 1 to implement adequate defense strategy to prevent or detect similar attacks in the future. As part of the protection mechanisms, we will setup snort IDS on the machine NeptuneR and IP Tables on the machine Neptune N.

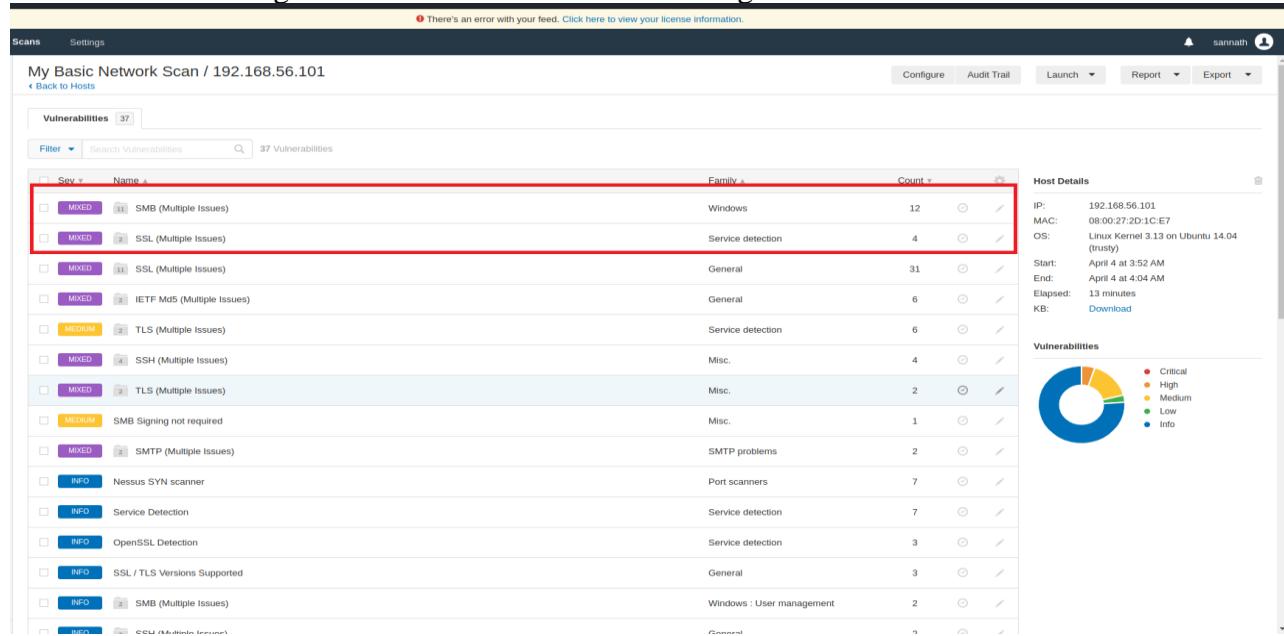
## Phase 1: Intrusion Detection

**Identify and select one medium or high-risk vulnerability (other than password cracking or unsupported OS version) against NeptuneN, for which you can identify an exploit code and execute successfully the exploit using Metasploit.**

- a. Turned on Neptune N machine.
- b. Searched for the IP addresses of N machine (192.168.56.102) using Kali.
- c. The Zenmap scan revealed for Neptune N seven opened ports with the services they were using.
- d. Nessus results after scanning the IP address (192.168.56.102) of machine Neptune N. NeptuneN with 4 high risk vulnerabilities and 26 medium vulnerabilities.



- e. One of the high-risk vulnerabilities is about configuration in the SMB service.



f. It is possible to access SMB shares without acquiring any credentials:

Scans Settings

My Basic Network Scan / 192.168.56.101 / SMB (Multiple Issues)

Configure Audit Trail Launch Report Export

Vulnerabilities 37

Search Vulnerabilities 11 Vulnerabilities

Sev	Name	Family	Count	Actions
HIGH	Microsoft Windows SMB Shares Unprivileged Access	Windows	1	
INFO	Microsoft Windows SMB Service Detection	Windows	2	
INFO	Microsoft Windows SMB Log In Possible	Windows	1	
INFO	Microsoft Windows SMB LsaQueryInformationPolicy Function SID Enumeration	Windows	1	
INFO	Microsoft Windows SMB NativeLanManager Remote System Information Disclosure	Windows	1	
INFO	Microsoft Windows SMB Share Hosting Office Files	Windows	1	
INFO	Microsoft Windows SMB Share Permissions Enumeration	Windows	1	
INFO	Microsoft Windows SMB Shares Enumeration	Windows	1	
INFO	Microsoft Windows SMB Versions Supported (remote check)	Windows	1	
INFO	Microsoft Windows SMB2 and SMB3 Dialects Supported (remote check)	Windows	1	
INFO	Windows NetBIOS / SMB Remote Host Information Disclosure	Windows	1	

Scan Details

Policy: Basic Network Scan  
Status: Completed  
Scanner: Local Scanner  
Start: April 4 at 3:52 AM  
End: April 4 at 4:04 AM  
Elapsed: 13 minutes

Vulnerabilities

Vulnerabilities 36

HIGH Microsoft Windows SMB Shares Unprivileged Access

Description

The remote has one or more Windows shares that can be accessed through the network with the given credentials.

Depending on the share rights, it may allow an attacker to read/write confidential data.

Solution

To restrict access under Windows, open Explorer, do a right click on each share, go to the 'sharing' tab, and click on 'permissions'.

Plugin Details

Severity: High  
ID: 42411  
Version: 1.8  
Type: remote  
Family: Windows  
Published: November 6, 2009  
Modified: July 27, 2018

Output

```
The following shares can be accessed using a NULL session :
- Files - (readable)
+ Content of this share :
..
create_postfix_mysql_tables.sql
vsftpd-2.3.4
openssl-1.0.1f
plfs-bootscripts-20140919
samba-3.5.0.tar.gz
```

Risk Information

Risk Factor: High  
CVSS Base Score: 7.5  
CVSS Temporal Score: 5.5  
CVSS Vector: CVSS2#AV:N/AC:L/Au:N/C:P/I:P/A:P  
CVSS Temporal Vector: CVSS2#E:U/RL:OF/RC:C

(Apart from the above vulnerability, even OpenSSL has a highly sever vulnerability.)

Scans Settings

My Basic Network Scan / 192.168.56.101 / SSL (Multiple Issues)

Configure Audit Trail Launch Report Export

Vulnerabilities 37

Search Vulnerabilities 2 Vulnerabilities

Sev	Name	Family	Count	Actions
HIGH	SSL Version 2 and 3 Protocol Detection	Service detection	3	
LOW	SSL Anonymous Cipher Suites Supported	Service detection	1	

Scan Details

Policy: Basic Network Scan  
Status: Completed  
Scanner: Local Scanner  
Start: April 4 at 3:52 AM  
End: April 4 at 4:04 AM  
Elapsed: 13 minutes

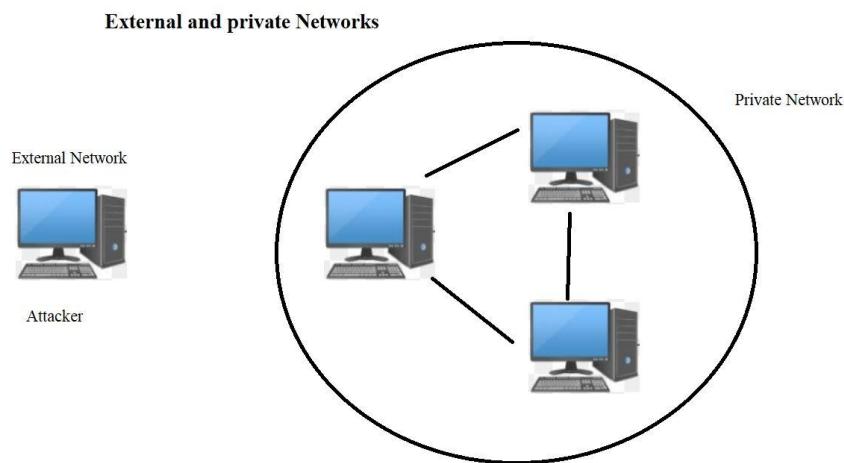
Vulnerabilities

**1. Explain briefly the generic attack scenario associated with the selected vulnerability (2 paragraphs maximum); a graphical sketch (in addition to the explanations) is required. Do not copy and paste paragraphs and figures from the Internet. You can search for such information online; but make sure that the explanation is reworded using your own terms, and the graph is your own (1.5%).**

**SMB:** Using the Server Message Block protocol, an application (or the user of an application within the network) can access files or other resources from the other machines in the same network. This allows applications to read, create, and update files on the remote server. It can also communicate with any server program that is set up to receive an SMB client request. SMB functions as a request-response or client-server protocol. Once the connection is established, the client computer or program can then open, read/write, and access files like the file system on a local computer.

**smb\_enumuser vulnerability** – we can see the general scenario of the attack in the diagram below.

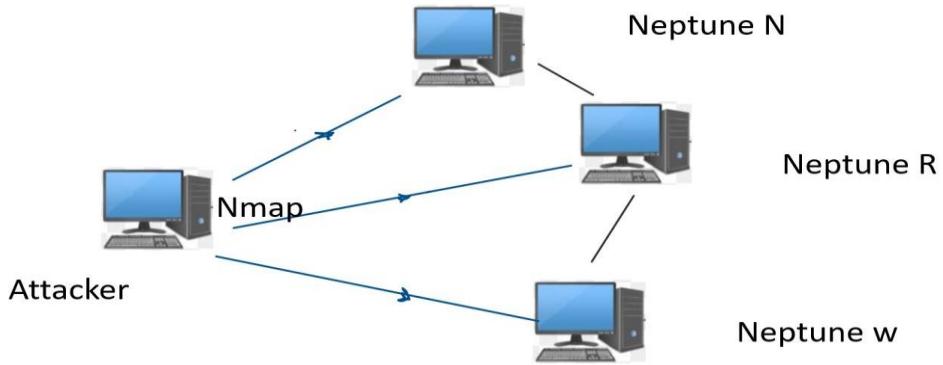
The attacker intrudes into the organization's (private) network and scans each of the machines in the network. The attacker looks for open services and selects the machine which has SMB as open service.



Attacker tries to execute the null sessions (User=NULL, password=NULL) using one of the tools like metasploit, enum4linux or smbclient. On successful execution, the tool should be capable of fetching data like machines list, group and users list, password protection policy and share lists (including comments in share-list). Sends request by sending user as null and password as null to bypass the authentication conducted by SMB protocol. With unprivileged access through null session the attacker can fetch the above-mentioned information.

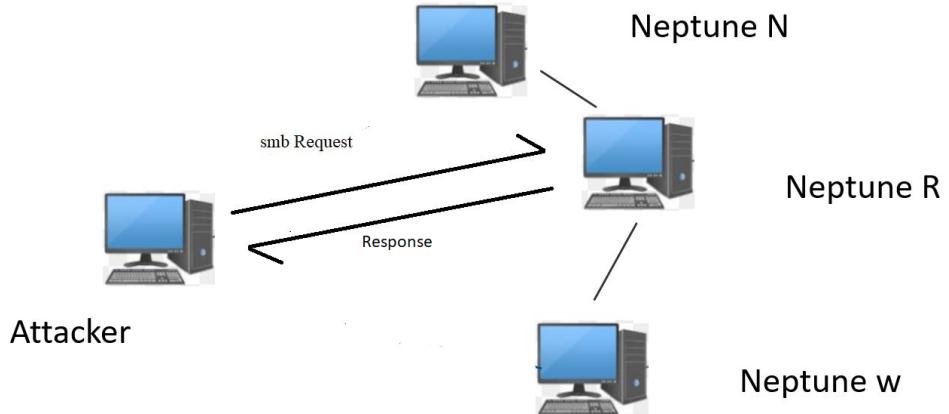
When the attack is initiated, the attack machine starts the request for smb ports. The target then responds to the request through one of the smb ports along with the authentication string. The authentication string is then processed by the attack machine to complete the negotiation which is sent in the next packet to the target. The target now receives the authentication results as response which decides the if to accept the connection and allow access or deny it. With the vulnerability of null session, the attacker bypasses this authentication with empty strings for username and password.

**Scanning open services of every machines in the network:**



(Reads the open services and select the machine which have ports 139 or 445 open\_

**Creating Null Session with the Machine**



**2. Execute the attack against machine NeptuneN by exploiting the selected vulnerability using Metasploit. Capture sample attack packets using wireshark (you can start wireshark, just before typing the “run” or “exploit” command in Metasploit; and stop it after the attack has completed).**

We identified and selected high-risk vulnerability in SMB protocol for Neptune N By reviewing the Nessus results for which we identified an exploit in Metasploit. Below are the exploits of smb which have been executed successfully. Exploits from metasploit related to SMB on Ports 139, 445.

```

msf5 > search scanner/smb
Matching Modules
=====
#  Name                               Disclosure Date  Rank   Check  Description
-  ----
0  auxiliary/scanner/smb/impacket/dcomexec      2018-03-19    normal No    DCOM Exec
1  auxiliary/scanner/smb/impacket/secretsdump     2018-03-19    normal No    DCOM Exec
2  auxiliary/scanner/smb/impacket/wmiexec        2018-03-19    normal No    WMI Exec
3  auxiliary/scanner/smb/pipe_auditor            normal No    SMB Session Pipe Auditor
4  auxiliary/scanner/smb/pipe_dcerpc_auditor      normal No    SMB Session Pipe DCERPC Auditor
5  auxiliary/scanner/smb/psexec_loggedin_users    normal No    Microsoft Windows Authenticated Logged In Users Enumeration
6  auxiliary/scanner/smb/smb1                     normal No    SMBv1 Protocol Detection
7  auxiliary/scanner/smb/smb2                     normal No    SMB 2.0 Protocol Detection
8  auxiliary/scanner/smb/smb_enum_gpp           normal No    SMB Group Policy Preference Saved Passwords Enumeration
9  auxiliary/scanner/smb/smb_enumshares          normal No    SMB Share Enumeration
10 auxiliary/scanner/smb/smb_enumusers           normal No    SMB User Enumeration (SAM EnumUsers)
11 auxiliary/scanner/smb/smb_enumusers_domain    normal No    SMB Domain User Enumeration
12 auxiliary/scanner/smb/smb_login              normal No    SMB Login Check Scanner
13 auxiliary/scanner/smb/smb_lookupsid         normal No    SMB SID User Enumeration (LookupSid)
14 auxiliary/scanner/smb/smb_ms17_010            normal No    MS17-010 SMB RCE Detection
15 auxiliary/scanner/smb/smb_uninit_cred        normal Yes   Samba _netr_ServerPasswordSet Uninitialized Credential State
16 auxiliary/scanner/smb/smb_version           normal No    SMB Version Detection

```

msf5 >

- a. **auxiliary/scanner/smb/smb\_1** - SMB 2.0 Protocol Detection Detect systems that support the SMB 1.0 protocol

```

File  Actions  Edit  View  Help
msf5 >
msf5 >
msf5 >
msf5 > use auxiliary/scanner/smb/smb1
msf5 auxiliary(scanner/smb/smb1) > show options

Module options (auxiliary/scanner/smb/smb1):
Name  Current Setting  Required  Description
----- 
RHOSTS      yes        The target host(s), range CIDR identifier, or hosts file with syntax 'file:<path>'
RPORT       445        yes        The target port (TCP)
THREADS     1          yes        The number of concurrent threads (max one per host)

msf5 auxiliary(scanner/smb/smb1) >
msf5 auxiliary(scanner/smb/smb1) >
msf5 auxiliary(scanner/smb/smb1) > set rhosts 192.168.56.103
rhosts => 192.168.56.103
msf5 auxiliary(scanner/smb/smb1) >
msf5 auxiliary(scanner/smb/smb1) > run
[*] 192.168.56.103:445  - 192.168.56.103 supports SMBv1 dialect.
[*] 192.168.56.103:445  - Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
msf5 auxiliary(scanner/smb/smb1) >

```

- b. **auxiliary/scanner/smb/smb\_2** - SMB 2.0 Protocol Detection Detect systems that support the SMB 2.0 protocol

```

msf5 > use auxiliary/scanner/smb/smb2
msf5 auxiliary(scanner/smb/smb2) >
msf5 auxiliary(scanner/smb/smb2) > show options

Module options (auxiliary/scanner/smb/smb2):
Name  Current Setting  Required  Description
----- 
RHOSTS      yes        The target host(s), range CIDR identifier, or hosts file with syntax 'file:<path>'
RPORT       445        yes        The target port (TCP)
THREADS     1          yes        The number of concurrent threads (max one per host)

msf5 auxiliary(scanner/smb/smb2) > set rhosts 192.168.56.103
rhosts => 192.168.56.103
msf5 auxiliary(scanner/smb/smb2) >
msf5 auxiliary(scanner/smb/smb2) > run
[+] 192.168.56.103:445  - 192.168.56.103 supports SMB 2 [dialect 255.2] and has been online for 3675191 hours
[*] 192.168.56.103:445  - Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
msf5 auxiliary(scanner/smb/smb2) >

```

- c. **auxiliary/scanner/smb/smb\_enumshares** - determines what shares are provided by the SMB service and which ones are readable/writable. It also collects additional information such as share types, directories, files, timestamps, etc.

```

msf5 >
msf5 > use auxiliary/scanner/smb/smb_enumshares
msf5 auxiliary(scanner/smb/smb_enumshares) > set rhosts 192.168.56.103
rhosts => 192.168.56.103
msf5 auxiliary(scanner/smb/smb_enumshares) >
msf5 auxiliary(scanner/smb/smb_enumshares) > show options

Module options (auxiliary/scanner/smb/smb_enumshares):
Name      Current Setting  Required  Description
LogSpider  3              no        0 = disabled, 1 = CSV, 2 = table (txt), 3 = one liner (txt) (Accepted: 0, 1, 2, 3)
MaxDepth   999             yes       Max number of subdirectories to spider
RHOSTS    192.168.56.103  yes       The target host(s), range CIDR identifier, or hosts file with syntax 'file:<path>'
SMBDomain .
SMBPass   .
SMBUser   .
ShowFiles  false            yes      Show detailed information when spidering
SpiderProfiles true            no       Spider only user profiles when share = C$
SpiderShares false           no       Spider shares recursively
THREADS   1              yes      The number of concurrent threads (max one per host)

msf5 auxiliary(scanner/smb/smb_enumshares) > set showfiles true
showfiles => true
msf5 auxiliary(scanner/smb/smb_enumshares) >
msf5 auxiliary(scanner/smb/smb_enumshares) > run
[+] 192.168.56.103:139 - printf - (DISK) Printer Drivers
[+] 192.168.56.103:139 - Files - (DISK)
[+] 192.168.56.103:139 - IPC$ - (IPC) IPC Service (neptuneN server (Samba
[+] 192.168.56.103:139 - Ubuntu))
[+] 192.168.56.103:139 - Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
msf5 auxiliary(scanner/smb/smb_enumshares) > ■

```

#### d. auxiliary/scanner/smb/smb\_enumusers

```

msf5 >
msf5 > use auxiliary/scanner/smb/smb_enumusers
msf5 auxiliary(scanner/smb/smb_enumusers) >
msf5 auxiliary(scanner/smb/smb_enumusers) > set rhosts 192.168.56.103
rhosts => 192.168.56.103
msf5 auxiliary(scanner/smb/smb_enumusers) >
msf5 auxiliary(scanner/smb/smb_enumusers) > show options

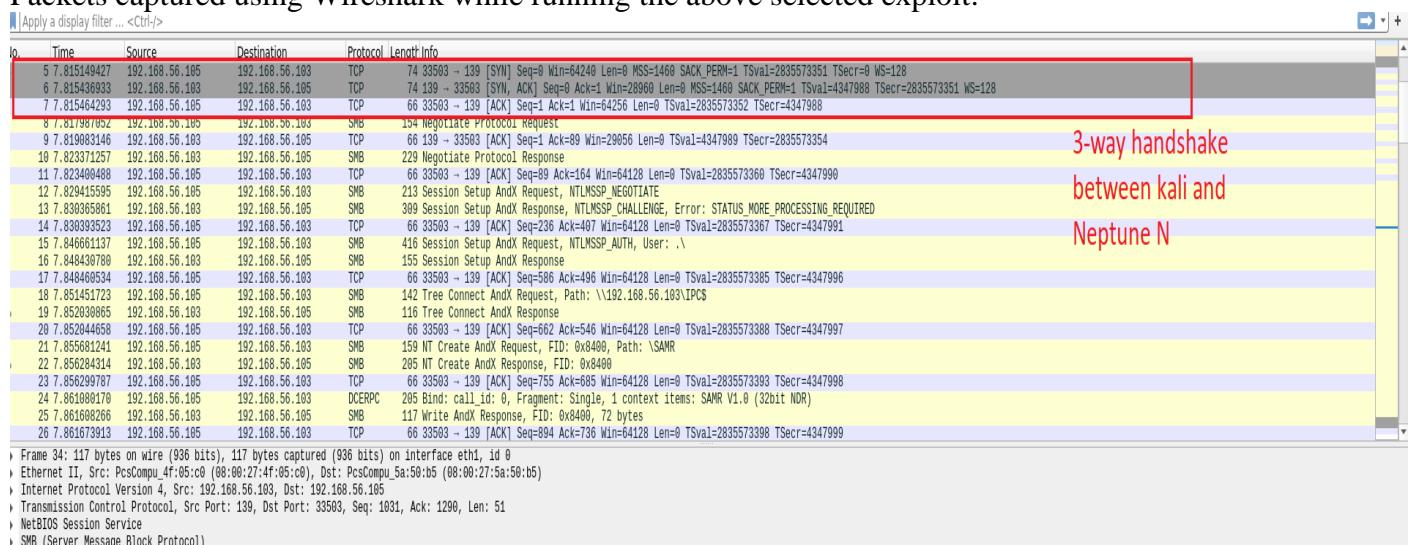
Module options (auxiliary/scanner/smb/smb_enumusers):
Name      Current Setting  Required  Description
RHOSTS    192.168.56.103  yes       The target host(s), range CIDR identifier, or hosts file with syntax 'file:<path>'
SMBDomain .
SMBPass   .
SMBUser   .
THREADS   1              yes      The number of concurrent threads (max one per host)

msf5 auxiliary(scanner/smb/smb_enumusers) > run
[+] 192.168.56.103:139 - NEPTUNEN [ nobody, jpsc ] ( LockoutTries=0 PasswordMin=5 )
[*] 192.168.56.103:139 - Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
msf5 auxiliary(scanner/smb/smb_enumusers) > ■

```

- The above exploits reveal certain information from the target machine. Smb\_enumusers fetches valuable information like some of the password policies and users in target machine (Neptune N).

Packets captured using Wireshark while running the above selected exploit:



```

0000 08 00 27 5a 5b 05 00 00 27 4f 05 c0 00 00 45 00  ..'Z...'0...E...
0010 00 67 de d9 48 00 40 06 69 96 c0 a8 38 67 c0 b5  g-@ i-8g-
0020 38 69 00 6b 02 df f8 16 8a 13 91 46 57 09 18 81.....3.FW-
0030 01 04 07 fb 00 00 01 01 08 0a 00 42 58 03 a9 03 .....BXc-
0040 6a 05 00 00 00 00 00 00 4d 42 2f 00 00 00 00 00 j.../S MB/...
0050 03 48 00 00 00 00 00 00 00 00 00 00 77 1a H.....W...
0060 86 77 08 11 32 9c 06 ff 00 00 00 34 00 00 00 00 w-2...4...
0070 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 .....

```

**3. Analyze the sample attack packets and define new Snort rules (as many as you think are necessary) to detect the attack, and add the rules to the snort rule set. Justify the rationale for the rules. Make sure your Snort rules do not over-fit the attack scenarios.**

a. Rules for attack from Metasploit – snort rules defined to detect the above executed attack.

The screenshot shows a terminal window titled "File: local.rules" with the text:

```
metasploit
alert tcp $EXTERNAL_NET any <> $HOME_NET [139,445]
content:"I00 00 00 00 dc 00 00 001"; depth:8 ; offset:154; msg:"metasploit"; s$
```

## My Basic Network Scan / Plugin #42411

[Back to Vulnerability Group](#)

Vulnerabilities 37

HIGH Microsoft Windows SMB Shares Unprivileged Access

### Description

The remote has one or more Windows shares that can be accessed through the network with the given credentials.

Depending on the share rights, it may allow an attacker to read/write confidential data.

### Solution

To restrict access under Windows, open Explorer, do a right click on each share, go to the 'sharing' tab, and click on 'permissions'.

### Output

```
The following shares can be accessed using a NULL session
- Files - (readable)
  + Content of this share :
.
create_postfix_mysql_tables.sql
vsftpd-2.3.4
openssl-1.0.1f
blfs-bootscripts-20140919
samba-3.5.0.tar.gz
proftpd-basic_1.3.5-rc3-2.1ubuntu2.1_amd64.deb
proftpd-1.3.5
proftpd-1.3.5.tar.gz
samba-3.5.0
blfs-bootscripts-20140919.tar.bz2
```

Port ▲	Hosts
445 / tcp / cifs	192.168.56.101

Above shown vulnerability says that a null session (user = NULL, password = NULL) can get unprivileged access.

### Explaining keywords in the rule:

Alert- to create an alert record in output CSV file TCP protocol

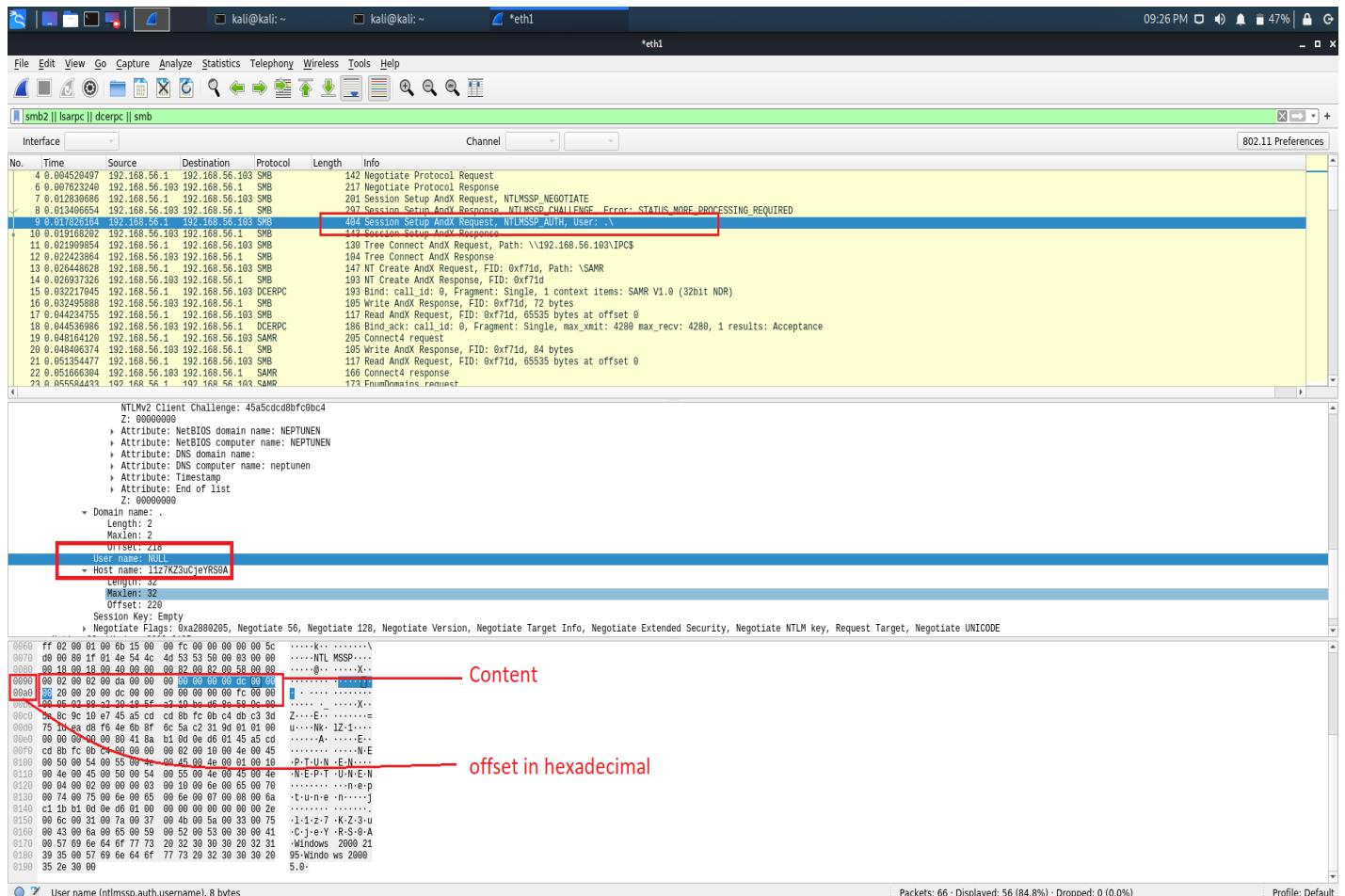
\$EXTERNAL\_ANY – the source ip is any of the ip addresses listed under this variable in snort.conf

any – source port number

→, ←, < > - defines direction/flow of packet

The default port number is 139 and as the port number can be changed to 445 to execute the attack, both the port numbers are mentioned to track attacks from metasploit.

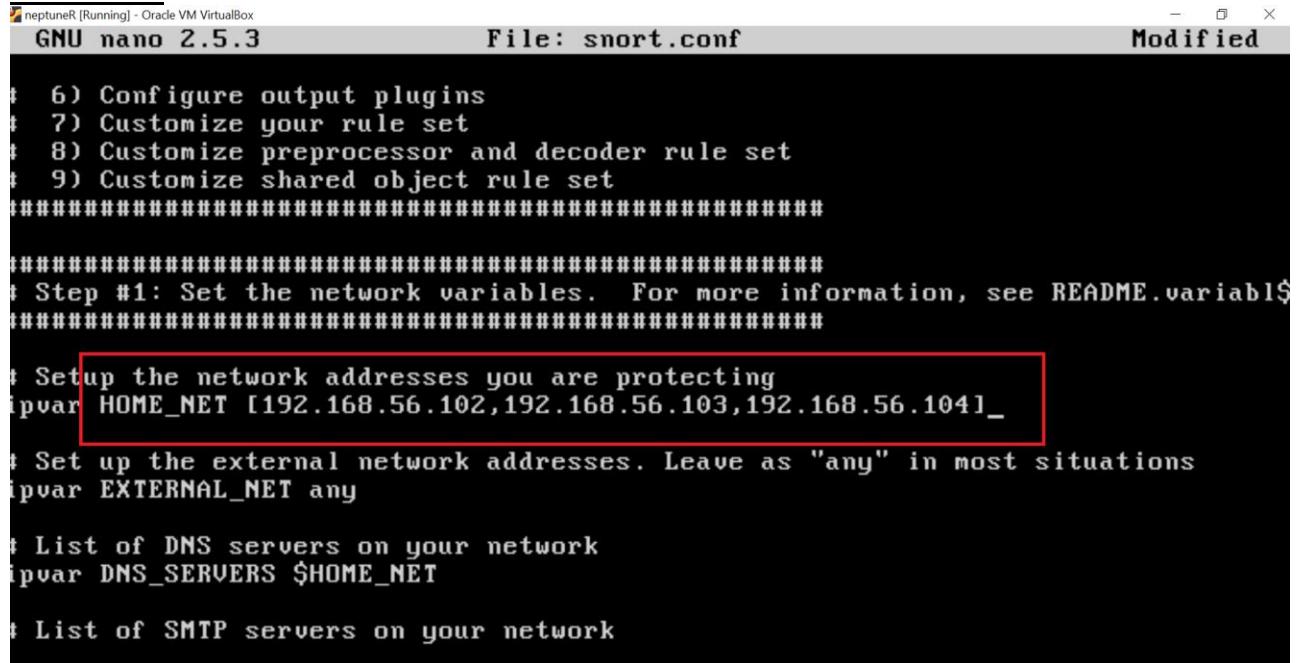
Offset has been used to make sure that the snort captures the packet only with the USER: NULL but nothing else. (As there is a possibility of having a NULL value assigned to other attributes in the packet).



**4. Configure Snort (on the Neptune machine) and run it in intrusion detection mode. Execute the relevant exploit for the vulnerability using your attack machine (i.e. Kali).**

Including the machines belonging to the neptune's network through HOME\_NET variable in configuration file.

**Snort.conf:**



```
neptunR [Running] - Oracle VM VirtualBox
GNU nano 2.5.3                               File: snort.conf                                Modified

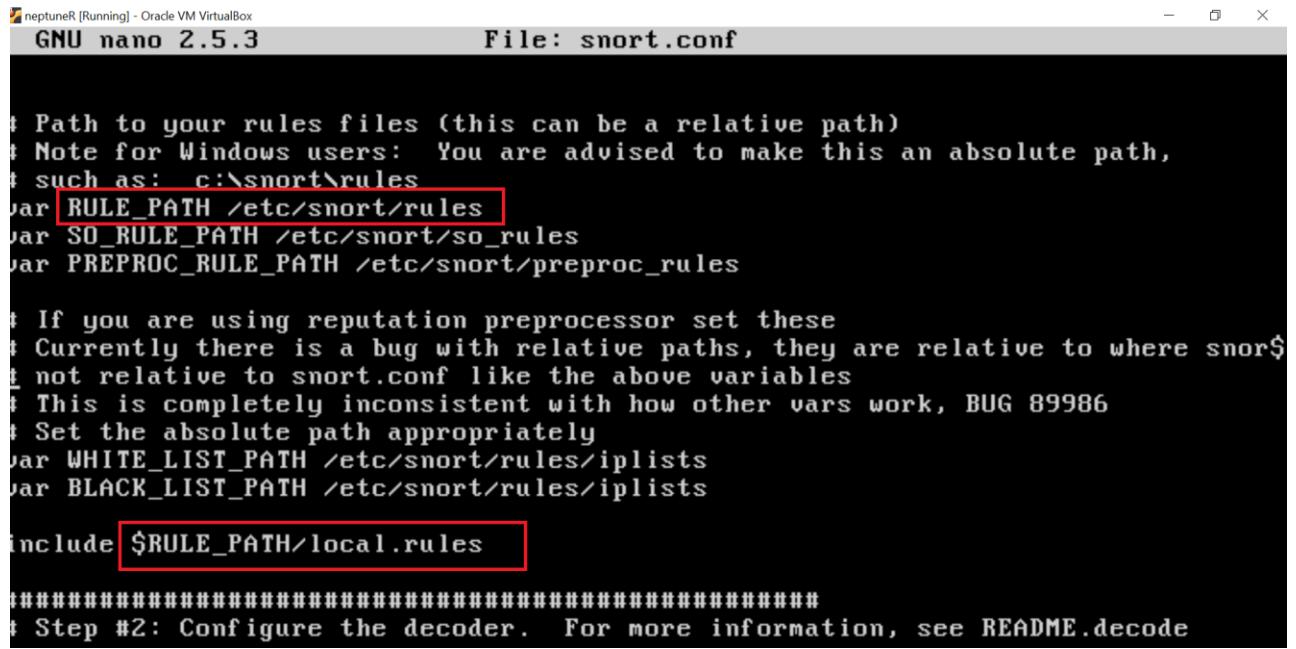
# 6) Configure output plugins
# 7) Customize your rule set
# 8) Customize preprocessor and decoder rule set
# 9) Customize shared object rule set
#####
#
# Step #1: Set the network variables. For more information, see README.variable
#####

# Setup the network addresses you are protecting
ipvar HOME_NET [192.168.56.102,192.168.56.103,192.168.56.104]

# Set up the external network addresses. Leave as "any" in most situations
ipvar EXTERNAL_NET any

# List of DNS servers on your network
ipvar DNS_SERVERS $HOME_NET

# List of SMTP servers on your network
```



```
neptunR [Running] - Oracle VM VirtualBox
GNU nano 2.5.3                               File: snort.conf                                Modified

# Path to your rules files (this can be a relative path)
# Note for Windows users: You are advised to make this an absolute path,
# such as: c:\snort\rules
var RULE_PATH /etc/snort/rules
var SO_RULE_PATH /etc/snort/snort_rules
var PREPROC_RULE_PATH /etc/snort/preproc_rules

# If you are using reputation preprocessor set these
# Currently there is a bug with relative paths, they are relative to where snort.conf is located
# not relative to snort.conf like the above variables
# This is completely inconsistent with how other vars work, BUG 89986
# Set the absolute path appropriately
var WHITE_LIST_PATH /etc/snort/rules/iplists
var BLACK_LIST_PATH /etc/snort/rules/iplists

include $RULE_PATH/local.rules

#####
#
# Step #2: Configure the decoder. For more information, see README.decoder
```

(HOME\_NET and path for local rules and output CSV file.)

```
#metasploit
alert tcp $EXTERNAL_NET any <> $HOME_NET [139,445] (content:"I00 00 00 00 00 dc 00$"

#smbclient
alert tcp $EXTERNAL_NET any <> $HOME_NET 445 (content:"I00 00 00 00 00 01 00 00$"

#enum4linux
alert tcp $EXTERNAL_NET any <> $HOME_NET 445 (content:"I00 00 00 00 00 01 00 00$
```

(Rules to detect all the variations of the attack)

Rule for Metasploit looks for the packet which has username: NULL. The same is conveyed in the rule as hexa-decimal format using content rule option. Rules for smbclient/enum4linux have the same content as the packets of these tools are structured the similar way and have the same content value.

```
enp0s10      Link encap:Ethernet HWaddr 08:00:27:06:9c:3c
              inet addr:192.168.56.102 Bcast:192.168.56.255 Mask:255.255.255.0
              inet6 addr: fe80::a00:27ff:fe06:9c3c/64 Scope:Link
              UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
              RX packets:414 errors:0 dropped:0 overruns:0 frame:0
              TX packets:506 errors:0 dropped:0 overruns:0 carrier:0
              collisions:0 txqueuelen:1000
              RX bytes:60746 (60.7 KB) TX bytes:48288 (48.2 KB)

interface to monitor      Link encap:Local Loopback
                           inet addr:127.0.0.1 Mask:255.0.0.0
                           inet6 addr: ::1/128 Scope:Host
                           UP LOOPBACK RUNNING MTU:65536 Metric:1
                           RX packets:3010 errors:0 dropped:0 overruns:0 frame:0
                           TX packets:3010 errors:0 dropped:0 overruns:0 carrier:0
                           collisions:0 txqueuelen:1
                           RX bytes:886954 (886.9 KB) TX bytes:886954 (886.9 KB)

lo                  Link encap:Local Loopback
                           inet addr:127.0.0.1 Mask:255.0.0.0
                           inet6 addr: ::1/128 Scope:Host
                           UP LOOPBACK RUNNING MTU:65536 Metric:1
                           RX packets:3010 errors:0 dropped:0 overruns:0 frame:0
                           TX packets:3010 errors:0 dropped:0 overruns:0 carrier:0
                           collisions:0 txqueuelen:1
                           RX bytes:886954 (886.9 KB) TX bytes:886954 (886.9 KB)
```

boneta@neptuneR:/etc/snort\$

Command to start snort:

```
pboneta@neptuneR:~/snort_log$ pwd  
/home/pboneta/snort_log  
pboneta@neptuneR:~/snort_log$ ls  
pboneta@neptuneR:~/snort_log$ sudo snort -c /etc/snort/snort.conf -l /home/pbonet  
a/snort_log/ -i enp0s10_
```

Snort started and waiting for packets to come into the network

```
By Martin Roesch & The Snort Team: http://www.snort.org/contact#team  
Copyright (C) 2014-2015 Cisco and/or its affiliates. All rights reserved.  
Copyright (C) 1998-2013 Sourcefire, Inc., et al.  
Using libpcap version 1.7.4  
Using PCRE version: 8.38 2015-11-23  
Using ZLIB version: 1.2.8  
  
Rules Engine: SF_SNORT_DETECTION_ENGINE Version 2.6 <Build 1>  
Preprocessor Object: SF_IMAP Version 1.0 <Build 1>  
Preprocessor Object: SF_GTP Version 1.1 <Build 1>  
Preprocessor Object: SF_SSH Version 1.1 <Build 3>  
Preprocessor Object: SF_DNS Version 1.1 <Build 4>  
Preprocessor Object: SF_SDF Version 1.1 <Build 1>  
Preprocessor Object: SF_REPUTATION Version 1.1 <Build 1>  
Preprocessor Object: SF_SMTP Version 1.1 <Build 9>  
Preprocessor Object: SF_MODBUS Version 1.1 <Build 1>  
Preprocessor Object: SF_SSLPP Version 1.1 <Build 4>  
Preprocessor Object: SF_DNP3 Version 1.1 <Build 1>  
Preprocessor Object: SF_SIP Version 1.1 <Build 1>  
Preprocessor Object: SF_DCERPC2 Version 1.0 <Build 3>  
Preprocessor Object: SF_POP Version 1.0 <Build 1>  
Preprocessor Object: SF_FTPTELNET Version 1.2 <Build 13>  
  
Commencing packet processing (pid=6042)
```

Metasploit - Running exploit

```
kali@kali:~  
File Actions Edit View Help  
msf5 auxiliary(scanner/smb/smb_enumusers) > show options  
  
Module options (auxiliary/scanner/smb/smb_enumusers):  


| Name      | Current Setting | Required | Description                                                                        |
|-----------|-----------------|----------|------------------------------------------------------------------------------------|
| RHOSTS    | 192.168.56.103  | yes      | The target host(s), range CIDR identifier, or hosts file with syntax 'file:<path>' |
| SMBDomain | .               | no       | The Windows domain to use for authentication                                       |
| SMBPass   |                 | no       | The password for the specified username                                            |
| SMBUser   |                 | no       | The username to authenticate as                                                    |
| THREADS   | 1               | yes      | The number of concurrent threads (max one per host)                                |

  
msf5 auxiliary(scanner/smb/smb_enumusers) > run  
[+] 192.168.56.103:139 - NEPTUNEN [ nobody, jpesci ] ( LockoutTries=0 PasswordMin=5 )  
[*] 192.168.56.103:139 - Scanned 1 of 1 hosts (100% complete)  
[*] Auxiliary module execution completed  
msf5 auxiliary(scanner/smb/smb_enumusers) >
```

## Stopping snort

```
smb2 tree disconnect: 0
smb2 close : 0

DCE/RPC
Connection oriented
Packet stats
PDUs: 22
Bind: 1
Bind Ack: 1
Request: 10
Response: 10
Request fragments: 0
Response fragments: 0
Client PDU segmented reassembled: 0
Server PDU segmented reassembled: 0
=====
=====
SIP Preprocessor Statistics
Total sessions: 0
=====
=====
Reputation Preprocessor Statistics
Total Memory Allocated: 0
=====
=====
Snort exiting
pboneta@neptuneR:~/etc/snort/rules$
```

(Stats of the packets encountered by snort)

5. Analyze the Snort alerts log generated after the attack, and discuss the results in terms of false positives and false negatives (in principle the snort configuration must successfully alerts on all suspicious packets, while not raising alerts on legitimate traffic) . Note: It matters that the exploit be relevant (you cannot pick one at random), and you must complete all the proper steps in Metasploit (initializing, launching, and completion). Provide screenshots documenting the different steps.

```
pboneta@neptuneR:~/snort_log$ pwd
/home/pboneta/snort_log
pboneta@neptuneR:~/snort_log$ ls
Output directory before
attack
pboneta@neptuneR:~/snort_log$
```

```

kali@kali:~$ sftp pboneta@192.168.56.102
pboneta@192.168.56.102's password:
Connected to 192.168.56.102.
sftp>
sftp> ls
snort_log snort_src
sftp> cd snort_log
sftp> ls
snort.log.1586397011 snort_d_alerts.csv
sftp> get snort_d_alerts.csv
Fetching /home/pboneta/snort_log/snort_d_alerts.csv to snort_d_alerts.csv
/home/pboneta/snort_log/snort_d_alerts.csv
100% 191 35.1KB/s 00:00
sftp>

```

```

File Edit Search View Document Help
104/09-10:50:22,079127 ,,,,"metasploit",TCP,192.168.1.,52334,192.168.56.103,,139,0A:00:27:00:00:29,08:00:27:4F:05:C0,0x194,**AP***,0xEAD560E,0x34756C6F,,0x2012,128,0,,1
metasploit attack from kali to
neptune R on port 139 (SMB)

```

Packet that matched in local rules

For all the three approaches of SMB attack, the packets are being captured by Snort. There are no false negatives for the snort rules as no malicious packet have gone undetected.

We can check for false positives, when a legit user within a Neptune network from one of the machines is trying to access the other machines in network and if such a packet is detected as malicious packet then it can be considered as a false negative.

This can happen if the content rule option is checking for the existing users and a new user is not updated in the content rule option of snort rule.

## SMB attack using smbclient:

```

root@kali:/home/kali# smbclient //192.168.56.103/Files --user=''
Enter WORKGROUP\`s password:
Try "help" to get a list of possible commands.
smb: \> ls
.
..
create_postfix_mysql_tables.sql      N     806   Wed Oct 25 23:43:51 2017
vsftpd-2.3.4                      D     0   Fri Sep 15 21:30:05 2017
openssl-1.0.1f                     D     0   Thu Sep 14 23:00:21 2017
blfs-bootscripts-20140919          D     0   Mon Sep 22 04:15:04 2014
samba-3.5.0.tar.gz                 N 30755153   Mon Mar  1 07:20:37 2010
proftpd-basic_1.3.5~rc3-2.1ubuntu2.1_amd64.deb    N 1967184   Wed Dec  7 22:18:30 2016
proftpd-1.3.5                      D     0   Thu May 15 12:44:12 2014
proftpd-1.3.5.tar.gz               N 7594509   Thu Sep 21 20:55:51 2017
samba-3.5.0                         D     0   Mon Mar  1 06:16:10 2010
blfs-bootscripts-20140919.tar.bz2   N 22931   Mon Sep 22 19:55:16 2014
smb: \>
8907816 blocks of size 1024. 5695292 blocks available
smb: \>

```

```
File Edit Search View Document Help  
*home/kali/Desktop_d_alerts.csv - Mousepad  
packets of metasploit  
1 04/08-18:50:22.079327 ,1,1,0,"metasploit",TCP,192.168.56.1,52334,192.168.56.103,139,0A:00:27:00:00:29,0B:00:27:4F:05:C0,0x194,***AP***,0xEAD560E,0x34756C6F,,0x2012,128,0,1  
2 3 04/08-19:13:42.568380 ,1,2,0,"smbclient",TCP,192.168.56.1,52479,192.168.56.103,445,0A:00:27:00:00:29,0B:00:27:4F:05:C0,0xDC,***AP***,0xCFD1DC2D,0xE1E83A4A,,0x2013,128,0,1  
4 04/08-19:13:42.568380 ,1,2,0,"enum4linux",TCP,192.168.56.1,52479,192.168.56.103,445,0A:00:27:00:00:29,0B:00:27:4F:05:C0,0xDC,***AP***,0xCFD1DC2D,0xE1E83A4A,,0x2013,128,0,  
5  
6 04/08-19:13:42.569240 ,1,2,0,"smbclient",TCP,192.168.56.103,445,192.168.56.1,52479,0B:00:27:4F:05:C0,0A:00:27:00:00:29,0x13B,***AP***,0xE1E83A4A,0xCFD1DCD3,,0xF5,,0,195  
7 04/08-19:13:42.569240 ,1,3,0,"enum4linux",TCP,192.168.56.103,445,192.168.56.1,52479,0B:00:27:4F:05:C0,0A:00:27:00:00:29,0x13B,***AP***,0xE1E83A4A,0xCFD1DCD3,,0xF5,,0,195  
8  
9 04/08-19:13:42.569896 ,1,2,0,"smbclient",TCP,192.168.56.1,52479,192.168.56.103,445,192.168.56.1,52479,0B:00:27:4F:05:C0,0x1BA,***AP***,0xCFD1DC03,0xE1E83BF,,0x2012,128,0,1  
10 04/08-19:13:42.569896 ,1,3,0,"enum4linux",TCP,192.168.56.1,52479,192.168.56.103,445,0A:00:27:00:00:29,0B:00:27:4F:05:C0,0x1BA,***AP***,0xCFD1DC03,0xE1E83BF,,0x2012,128,0,  
11  
packets of smbclient  
  
as smbclient and enum4linux have the same packet information adn structure, rules mentioned  
for both , smbclient and enum4linux, will be matched when smbclient or enum4linux is used to  
attack
```

# SMB attack using enum4linux:

```
kali㉿kali:~$ enum4linux -a 192.168.56.103
```

packets of enum4linx attack

```

File Edit Search View Document Help
1 04/08-18:50:22.079127 ,,,,"metasploit",TCP,192.168.56.1,52334,192.168.56.103,139,0A:00:27:0F:05:C0,0x194,***AP***,0xEDA560E,0x34756C6F,,0x2012,128,0,
2
3 04/08-19:23:42.568380 ,1,2,0,"smbclient",TCP,192.168.56.1,52479,192.168.56.103,445,0A:00:27:0F:05:C0,0xDC,***AP***,0xCFD1DC2D,,0xE1E83A4A,,0x2013,128,0,
4 04/08-19:23:42.568380 ,1,3,0,"enum4linux",TCP,192.168.56.1,52479,192.168.56.103,445,0A:00:27:0F:05:C0,0xDC,***AP***,0xCFD1DC2D,,0xE1E83A4A,,0x2013,128,0,
5 04/08-19:23:42.569240 ,1,2,0,"smbclient",TCP,192.168.56.103,445,192.168.56.1,52479,0B:00:27:0F:05:C0,0A:00:27:0F:05:C0,,0x13B,***AP***,0xE1E83A4A,,0xCFD1DC3,,0xF5,64,0,105
6 04/08-19:23:42.569240 ,1,3,0,"enum4linux",TCP,192.168.56.103,445,192.168.56.1,52479,0B:00:27:0F:05:C0,0A:00:27:0F:05:C0,,0x13B,***AP***,0xE1E83A4A,,0xCFD1DC3,,0xF5,64,0,105
7 04/08-19:23:42.569896 ,1,2,0,"smbclient",TCP,192.168.56.1,52479,192.168.56.103,445,0A:00:27:0F:05:C0,0x1BA,***AP***,0xCFD1DC3,,0xE1E83B4F,,0x2012,128,0,
8 04/08-19:23:42.569896 ,1,3,0,"enum4linux",TCP,192.168.56.1,52479,192.168.56.103,445,0A:00:27:0F:05:C0,0x1BA,***AP***,0xCFD1DC3,,0xE1E83B4F,,0x2012,128,0,
9
10 04/08-19:29:49.706822 ,1,3,0,"enum4linux",TCP,192.168.56.1,52499,192.168.56.103,445,0A:00:27:0F:05:C0,0xDC,***AP***,0xCF7F7183,,0x2013,128,0,
11 04/08-19:29:49.706822 ,1,2,0,"smbclient",TCP,192.168.56.1,52499,192.168.56.103,445,0A:00:27:0F:05:C0,0x13B,***AP***,0xCF7F7183,,0x2013,128,0,
12 04/08-19:29:49.707517 ,1,3,0,"enum4linux",TCP,192.168.56.103,445,192.168.56.1,52499,0B:00:27:0F:05:C0,0A:00:27:0F:05:C0,,0x13B,***AP***,0xDE99F54C,,0xCF7F7229,,0xF5,64,0,105
13 04/08-19:29:49.707517 ,1,2,0,"smbclient",TCP,192.168.56.103,445,192.168.56.1,52499,0B:00:27:0F:05:C0,0A:00:27:0F:05:C0,,0x13B,***AP***,0xDE99F54C,,0xCF7F7229,,0xF5,64,0,105
14 04/08-19:29:49.707889 ,1,3,0,"enum4linux",TCP,192.168.56.1,52499,192.168.56.103,445,0A:00:27:0F:05:C0,0x120,***AP***,0xCF7F7229,,0x2012,128,0,
15 04/08-19:29:49.707889 ,1,2,0,"smbclient",TCP,192.168.56.1,52499,192.168.56.103,445,0A:00:27:0F:05:C0,0x120,***AP***,0xDE99F651,,0x2012,128,0,
16 04/08-19:29:49.796067 ,1,3,0,"enum4linux",TCP,192.168.56.1,52500,192.168.56.103,445,0A:00:27:0F:05:C0,0x120,***AP***,0x92297F99,,0xD0375D54,,0x2013,128,0,
17 04/08-19:29:49.796067 ,1,2,0,"smbclient",TCP,192.168.56.1,52500,192.168.56.103,445,0A:00:27:0F:05:C0,0x120,***AP***,0x92297F99,,0x7D375D54,,0x2013,128,0,
18 04/08-19:29:49.796715 ,1,3,0,"enum4linux",TCP,192.168.56.103,445,192.168.56.1,52500,0B:00:27:0F:05:C0,0A:00:27:0F:05:C0,,0x13B,***AP***,0x7D375D54,,0x9229803F,,0xF5,64,0,32
19 04/08-19:29:49.796715 ,1,2,0,"smbclient",TCP,192.168.56.103,445,192.168.56.1,52500,0B:00:27:0F:05:C0,0A:00:27:0F:05:C0,,0x13B,***AP***,0x7D375D54,,0x9229803F,,0xF5,64,0,32
20 04/08-19:29:49.797130 ,1,3,0,"enum4linux",TCP,192.168.56.1,52500,192.168.56.103,445,0A:00:27:0F:05:C0,0x102,***AP***,0x9229803F,,0xD0375E59,,0x2012,128,0,
21 04/08-19:29:49.797130 ,1,2,0,"smbclient",TCP,192.168.56.1,52500,192.168.56.103,445,0A:00:27:0F:05:C0,0x102,***AP***,0x9229803F,,0xD0375E59,,0x2012,128,0,
22 04/08-19:29:49.876255 ,1,3,0,"enum4linux",TCP,192.168.56.1,52501,192.168.56.103,445,0A:00:27:0F:05:C0,0x120,***AP***,0x5FB482DF,,0x9E67218,,0x2013,128,0,
23 04/08-19:29:49.876255 ,1,2,0,"smbclient",TCP,192.168.56.1,52501,192.168.56.103,445,0A:00:27:0F:05:C0,0x120,***AP***,0x5FB482DF,,0x9E67218,,0x2013,128,0,
24 04/08-19:29:49.877064 ,1,3,0,"enum4linux",TCP,192.168.56.103,445,192.168.56.1,52501,0B:00:27:0F:05:C0,0A:00:27:0F:05:C0,,0x13B,***AP***,0x79E67218,,0x5FB48385,,0xF5,64,0,44
25 04/08-19:29:49.877064 ,1,2,0,"smbclient",TCP,192.168.56.103,445,192.168.56.1,52501,0B:00:27:0F:05:C0,0A:00:27:0F:05:C0,,0x13B,***AP***,0x79E67218,,0x5FB48385,,0xF5,64,0,44
26 04/08-19:29:49.877786 ,1,3,0,"enum4linux",TCP,192.168.56.1,52501,192.168.56.103,445,0A:00:27:0F:05:C0,0x120,***AP***,0x5FB48385,,0x79E67310,,0x2012,128,0,
27 04/08-19:29:49.877786 ,1,2,0,"smbclient",TCP,192.168.56.1,52501,192.168.56.103,445,0A:00:27:0F:05:C0,0x120,***AP***,0x5FB48385,,0x79E67310,,0x2012,128,0,
28 04/08-19:29:49.941230 ,1,3,0,"enum4linux",TCP,192.168.56.1,52501,192.168.56.103,445,0A:00:27:0F:05:C0,0x120,***AP***,0x20FA11E2,,0x1B614580,,0x2013,128,0,
29 04/08-19:29:49.941230 ,1,2,0,"smbclient",TCP,192.168.56.1,52501,192.168.56.103,445,0A:00:27:0F:05:C0,0x120,***AP***,0x20FA11E2,,0x1B614580,,0x2013,128,0,
30 04/08-19:29:49.942051 ,1,3,0,"enum4linux",TCP,192.168.56.103,445,192.168.56.1,52502,0B:00:27:0F:05:C0,0A:00:27:0F:05:C0,,0x13B,***AP***,0x1B614580,,0x20FA1288,,0xF5,64,0,39
31 04/08-19:29:49.942051 ,1,2,0,"smbclient",TCP,192.168.56.103,445,192.168.56.1,52502,0B:00:27:0F:05:C0,0A:00:27:0F:05:C0,,0x13B,***AP***,0x1B614580,,0x20FA1288,,0xF5,64,0,39
32 04/08-19:29:49.942891 ,1,3,0,"enum4linux",TCP,192.168.56.1,52502,192.168.56.103,445,0A:00:27:0F:05:C0,0x102,***AP***,0x20FA1288,,0x1B614692,,0x2012,128,0,
33 04/08-19:29:49.942891 ,1,2,0,"smbclient",TCP,192.168.56.1,52502,192.168.56.103,445,0A:00:27:0F:05:C0,0x102,***AP***,0x20FA1288,,0x1B614692,,0x2012,128,0,
34 04/08-19:29:50.947490 ,1,3,0,"enum4linux",TCP,192.168.56.1,52503,192.168.56.103,445,0A:00:27:0F:05:C0,0x102,***AP***,0x378FE53,,0xDF429398,,0x2013,128,0,
35 04/08-19:29:50.947490 ,1,2,0,"smbclient",TCP,192.168.56.1,52503,192.168.56.103,445,0A:00:27:0F:05:C0,0x102,***AP***,0x378FE53,,0xDF429398,,0x2013,128,0

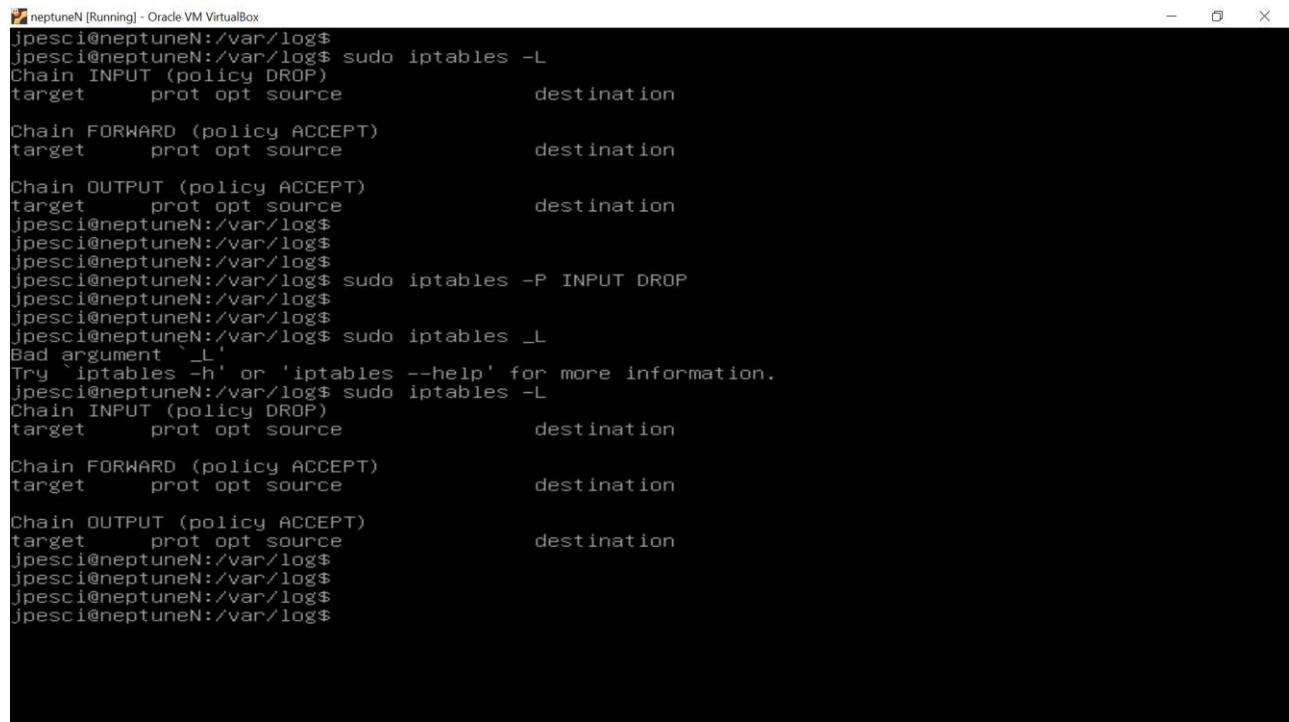
```

Since the flag -a in the command raises multiple requests more than 2000 packets are captured in this approach.

## Phase 2: Intrusion Prevention

To protect against the above attacks, we would like to reinforce the IDS protection using IP Tables firewall. The protection scope (in this phase) will be the Neptune N machine, i.e., the IP Tables rules will be deployed on Neptune N. Since this part of the project focuses on protection, it is assumed that you'll have direct access to the internal network. This means you can update the firewall rules on the machine directly. The default root credentials for Neptune N will be given after the deadline for Part 1. Note that snort will run only in non-inline mode. That is, it does detection only, and does not actively prevent anything. We use IP Tables for that.

- 1. Define the IP Tables rules and provide rationale for each of the rules. You should minimize false negatives and false positives so that a legitimate client is allowed access, but a client that attempts the selected attacks is blocked.**



```
neptuneN [Running] - Oracle VM VirtualBox
jpesci@neptuneN:/var/log$ sudo iptables -L
Chain INPUT (policy DROP)
target     prot opt source          destination
Chain FORWARD (policy ACCEPT)
target     prot opt source          destination
Chain OUTPUT (policy ACCEPT)
target     prot opt source          destination
jpesci@neptuneN:/var/log$ sudo iptables -P INPUT DROP
jpesci@neptuneN:/var/log$ 
jpesci@neptuneN:/var/log$ 
jpesci@neptuneN:/var/log$ sudo iptables -L
Bad argument '-L'
Try 'iptables -h' or 'iptables --help' for more information.
jpesci@neptuneN:/var/log$ sudo iptables -L
Chain INPUT (policy DROP)
target     prot opt source          destination
Chain FORWARD (policy ACCEPT)
target     prot opt source          destination
Chain OUTPUT (policy ACCEPT)
target     prot opt source          destination
jpesci@neptuneN:/var/log$ 
jpesci@neptuneN:/var/log$ 
jpesci@neptuneN:/var/log$ 
jpesci@neptuneN:/var/log$
```

Default iptables : sudo iptables -L

Above screenshot shows changing of policy of INPUT chain from ACCEPT to DROP.

Below is the rule added for the Metasploit attack :

```
neptuneN [Running] - Oracle VM VirtualBox
jpesci@neptuneN:~$ 
jpesci@neptuneN:~$ 
jpesci@neptuneN:~$ 
jpesci@neptuneN:~$ sudo iptables -A INPUT -p tcp -m multiport --dports 139,445 -m string --algo bm --hex-string '|00000000dc000000|' --from 166 --to 173 -j DROP
jpesci@neptuneN:~$
```

```
neptuneN [Running] - Oracle VM VirtualBox
jpesci@neptuneN:~$ 
jpesci@neptuneN:~$ 
jpesci@neptuneN:~$ 
jpesci@neptuneN:~$ sudo iptables -A INPUT -p tcp -m multiport --dports 139,445 -m string --algo bm --hex-string '|00000000dc000000|' --from 166 --to 173 -j DROP
jpesci@neptuneN:~$ 
jpesci@neptuneN:~$ sudo iptables -L
Chain INPUT (policy ACCEPT)
target     prot opt source               destination
DROP      tcp   --  anywhere             anywhere             multiport dports netbios-ssn,microsoft
-ds STRING match "|00000000dc000000|" ALGO name bm FROM 166 TO 173

Chain FORWARD (policy ACCEPT)
target     prot opt source               destination

Chain OUTPUT (policy ACCEPT)
target     prot opt source               destination
jpesci@neptuneN:~$
```

As the request packets from the attacker has to be blocked, we add this rule to INPUT chain and change default policy from ACCEPT to DROP.

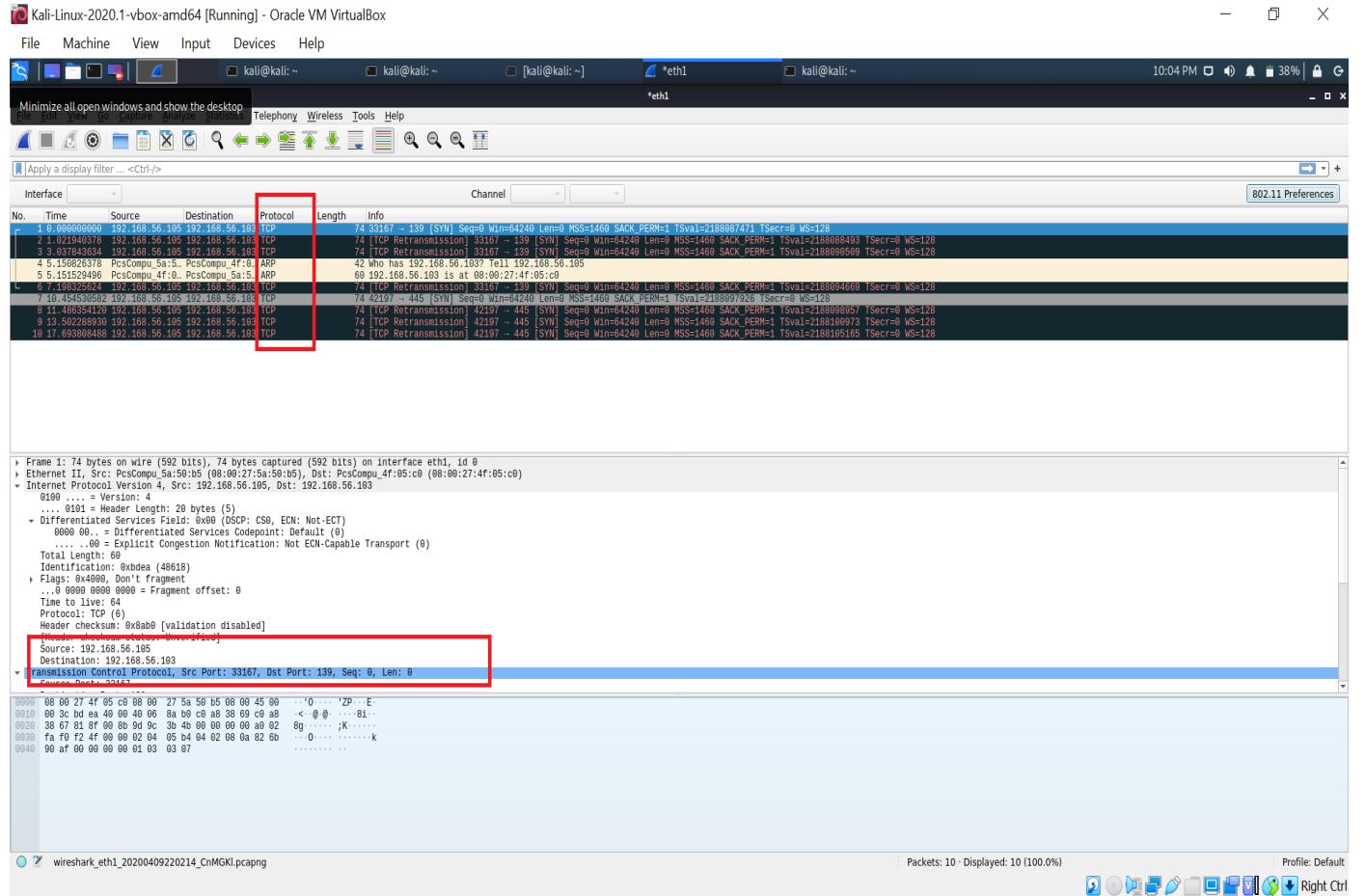
In the above rule, if TCP packets coming into Neptune 's network with following information:

- Port number 135 and 445
- Match the hex-string of (username) NULL
- If the string which is matched starts from 166 byte to 173 bytes

Then such packets are considered as malicious packets and they are blocked(dropped).

The above rule is for the attack from Metasploit but the same attack from other tools (, enum4linux) can be blocked in the same way by changing the --hex- string to be matched and by mentioning the start position(from offset) of the matched string (Username: NULL)

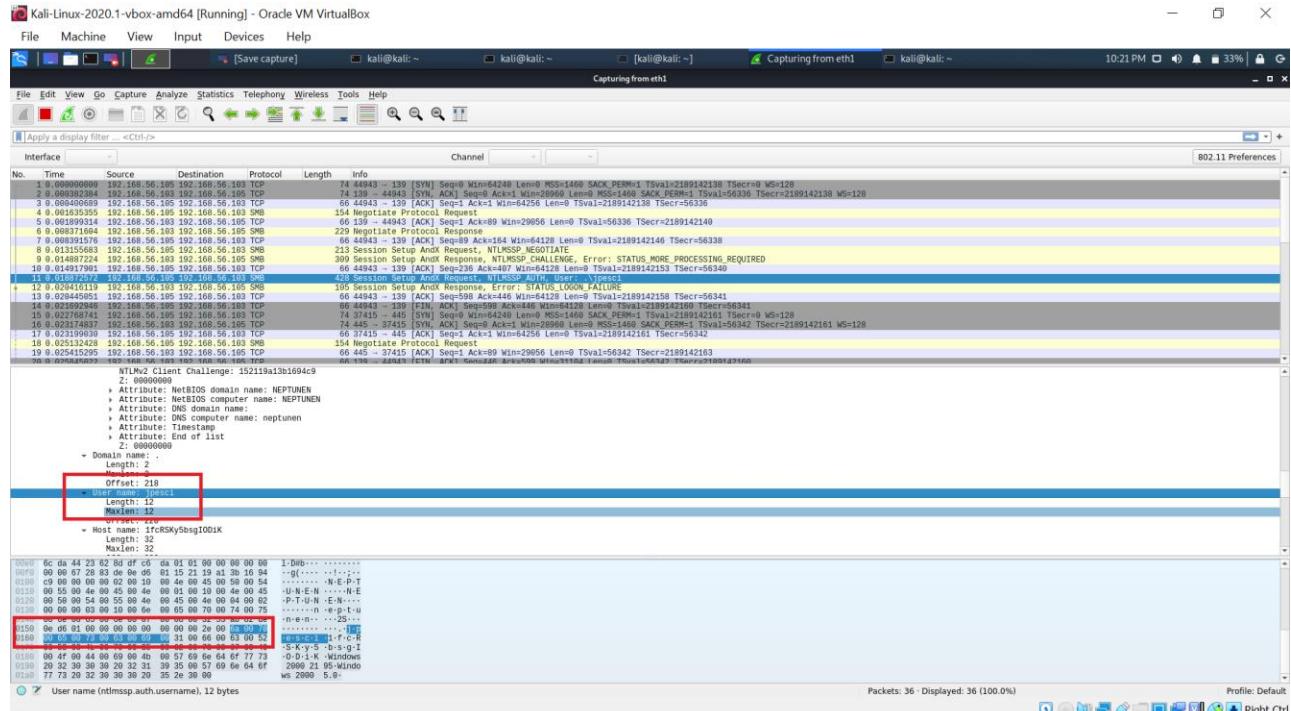
This figure shows Wireshark for null session:



And for legit client since the username field will be having a value other than null, the user would not be blocked.

## Wireshark packets for jpesci:

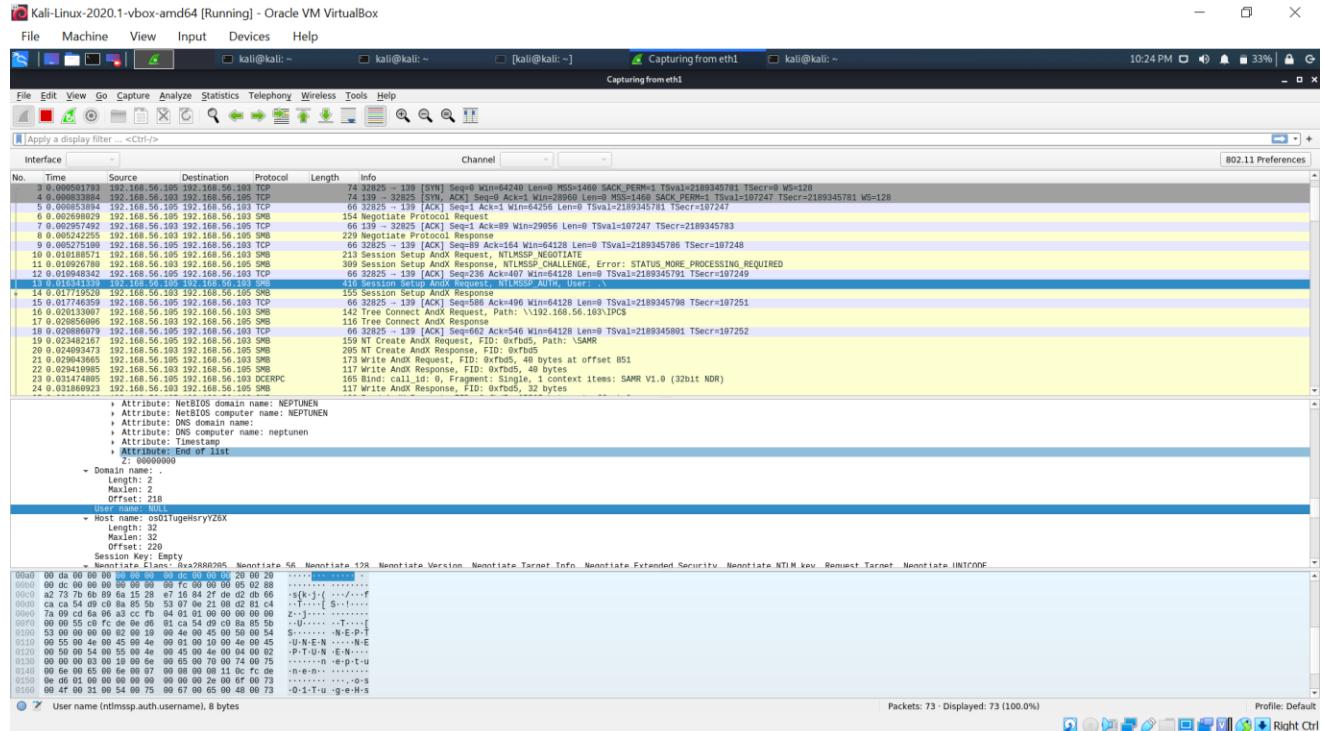
By setting the SMBuser and SMBpass, we tried to simulate the genuine packet and below is the screenshot of those packets.



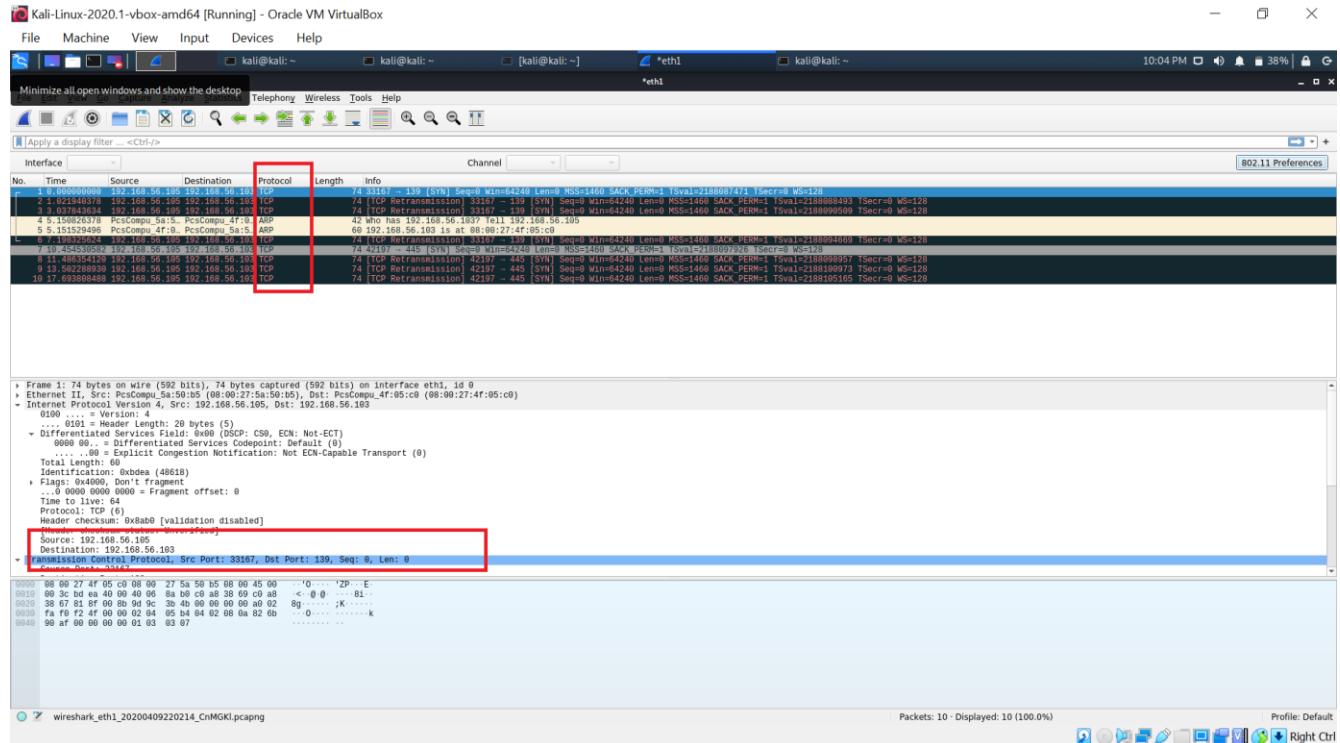
**2. Test the firewall rules by executing the attacks (in metasploit, when relevant exploit exists); provide screenshots documenting the results.**

## Running the exploit:

Packet of the attack are captured using Wireshark:  
 Figure bellow show the packets of the attack before adding the rule to IPtable (packet without Rule)



Packets captured after adding rule to IPtable



In the above screenshots, there are no SMB packets after adding the rule. Which says that attack was not successful.

Metasploit output:

The screenshot shows a terminal window titled 'Kali-Linux-2020.1-vbox-amd64 [Running] - Oracle VM VirtualBox'. The terminal session is as follows:

```
msf5 auxiliary(scanner/smb/smb_enumusers) >
msf5 auxiliary(scanner/smb/smb_enumusers) >
msf5 auxiliary(scanner/smb/smb_enumusers) >
msf5 auxiliary(scanner/smb/smb_enumusers) > show options

Module options (auxiliary/scanner/smb/smb_enumusers):

Name      Current Setting  Required  Description
----      -----          -----    -----
RHOSTS    192.168.56.103  yes        The target host(s), range CIDR identifier, or hosts file with syntax 'file:<path>'
SMBDomain .            no         The Windows domain to use for authentication
SMBPass   .            no         The password for the specified username
SMBUser   .            no         The username to authenticate as
THREADS   1            yes        The number of concurrent threads (max one per host)

[*] 192.168.56.103:139  - Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
msf5 auxiliary(scanner/smb/smb_enumusers) >
```

The output indicates that the scan was completed successfully, scanning 1 host (100% complete). A red box highlights the completion message: '[\*] Auxiliary module execution completed'.

### 3.By reviewing the scan results (obtained in project – Part 1), suggest and describe any additional defense strategy to protect the target systems.

By reviewing the Zenmap scans from project 1 and at the beginning of project 2, we could see that most of the critical services on the network were exposed and seemed to be using the default configurations. In addition, mailing services (SMTP, IMAP, and POP3) were using weak ciphers. FTP and SSH are possibly vulnerable to brute force attack, since there may not be any back-off mechanism configured with these services.

Using firewall, it is possible to protect many information inside the private network invisible to the external world. But for example, when somebody from inside organization need to have access to resources from outside, we need virtual private network (VPN). VPN lets user from outside of a private network to take a detour around the firewall and make it possible to have access to internal network. This happen through using combination of software and security measures. VPN works based on encryption of all the traffic entering and leaving by making encrypted tunnel between a user and remote server, so all date traffics are going to be routed through this tunnel. So, our data stay secure and out IP address and identity and location have mask. At the end the security is improved, and VPN improves security by offering confidentiality and integrity.

- The first step is to have a strong firewall system. No doubt, antivirus is the first thing in mind when it comes to the cyber security. Enable a firewall or endpoint protection to protect these ports from attackers. Most solutions include a blacklist to prevent connections from known attackers IP addresses.
- Keeping the system updated with all the recent versions or patches is extremely important to block the attacker from getting access to the system.
- Creating strong snort rules as we did up in this report will help to prevent the organisation from intended attacker.

- d. Install a VPN to encrypt and protect network traffic.
- e. Implement VLANs to isolate internal network traffic.
- f. Use MAC address filtering to keep unknown systems from accessing the network. This tactic requires significant management to keep the list maintained.
- g. In addition to the network specific protections above, we can implement a data centric security plan to protect the most important resource – the data that lives on our SMB file shares.