DAY10_advent_of_the_cyber

10.2. Today's Learning Objectives:
Learn about the basics of network file sharing protocols before getting hands-on with Samba, where you will be enumerating "*tbfc-smb-01*": a vulnerable Samba server to gain un-authorised access.

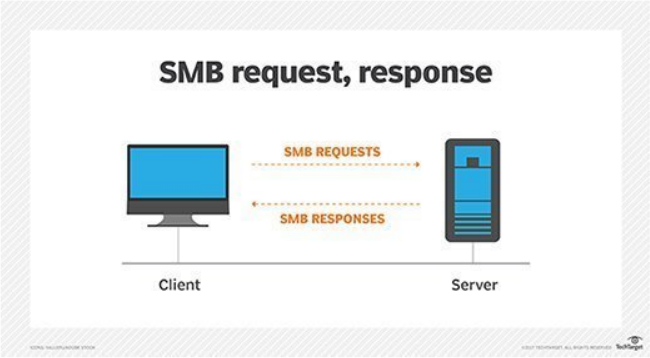
Made with ❤ by CMNatic

10.3. What is Samba & where is it Used?

Whilst we learnt about one of the most commonplace protocols that are used for file-sharing on Day 10, we'll be covering an alternative technology for file-sharing that is most used within organisation/company networks. Offering encryption as standard, this technology consists of two protocols:

- SMB (Server Message Block) - Natively supported by Windows and not Linux
- NFS (Network File System) - Natively supported by Linux and not Windows

Protocols such as SMB send "requests" and "responses" when communicating with each other, as illustrated below:



(TechTarget., 2017)

What makes Samba so popular and useful is that it removes the differences between these two protocols, meaning that the two operating systems can now share resources including files amongst each other. Simply, Samba connects to a "**share**" (think of this as a virtual folder) and is capable of day-to-day activities like deleting, moving or uploading files.

Samba is flexible in the sense it can be useful for both you and me or businesses with thousands of employees. For example, employees can access documents from a central computer rather than each employee storing their own copy. As previously mentioned, this technology is encrypted enabling sensitive data like username and passwords used in the authentication process (and the data itself) to be communicated between client/server securely.

Unlike FTP, other IT devices such as network printers can also be shared between client/server.

10.4. Searching for Samba Shares

We're going to be using the *enum4linux* tool that is already provided to you on the THM AttackBox. Let's get our hands dirty!

1. Open a terminal prompt and navigate to enum4linux: `cd /root/Desktop/Tools/Miscellaneous`
2. Run enum4linux and list all the possible options we could use, take time to study these for anything interesting: `./enum4linux.pl -h`

```
root@ip-10-10-171-174:~# cd /root/Desktop/Tools/Miscellaneous
root@ip-10-10-171-174:~/Desktop/Tools/Miscellaneous# ./enum4linux.pl -h
enum4linux v0.8.9 (http://labs.portcullis.co.uk/application/enum4linux/)
Copyright (C) 2011 Mark Lowe (mrl@portcullis-security.com)

Simple wrapper around the tools in the samba package to provide similar
functionality to enum.exe (formerly from www.bindview.com).  Some additional
features such as RID cycling have also been added for convenience.

Usage: ./enum4linux.pl [options] ip

Options are (like "enum"):
    -U        get userlist
    -M        get machine list*
    -S        get sharelist
    -P        get password policy information
    -G        get group and member list
    -d        be detailed, applies to -U and -S
    -u user   specify username to use (default "")
    -p pass   specify password to use (default "")

The following options from enum.exe aren't implemented: -L, -N, -D, -f
```

Note how we can use options like `-S` to list shares or `-U` (note the uppercase) to list possible users. In my example, I want to find out who can be used to access the server through Samba: `./enum4linux.pl -U 10.10.54.135`

```
 ============================
|    Users on 192.168.1.200    |
 ============================
index: 0x1 RID: 0x3e8 acb: 0x00000010 Account: jjohns   Name:   Desc:
index: 0x2 RID: 0x3ea acb: 0x00000010 Account: lbutton  Name:   Desc:
index: 0x3 RID: 0x3e9 acb: 0x00000010 Account: jfrost   Name:   Desc:
index: 0x4 RID: 0x3eb acb: 0x00000010 Account: cmnatic  Name:   Desc:

user:[jjohns] rid:[0x3e8]
user:[lbutton] rid:[0x3ea]
user:[jfrost] rid:[0x3e9]
user:[cmnatic] rid:[0x3eb]
enum4linux complete on Thu Nov 12 00:53:47 2020

root@kali:~#
```

Note how *enum4linux* has discovered four users in my example...One of these users may have a weak password such as "password123" that we can log in with and access sensitive data as.

1. jjohns
2. lbutton
3. jfrost
4. cmnatic

And as a result of further enumeration with *enum4linux*, we've discovered the following three shares!

1. homes
2. share1
3. IPC$

```
 ========================================
|    Share Enumeration on 192.168.1.200    |
 ========================================

        Sharename       Type      Comment
        ---------       ----      -------
        homes           Disk      Home Directories
        share1          Disk      A Shared Directory
        IPC$            IPC       IPC Service (tbfc-smb server (Samba, Ubuntu))
SMB1 disabled -- no workgroup available
```

*Now it's your turn, scan your Instance (10.10.54.135) to answer **Question #1** and **Question #2**. Remember the options that* we can use with `enum4linux.pl` *!*

10.5. Connecting to a Share

We've already learnt two key pieces of information from the previous section:

- Usernames to authenticate as
- Shares that we can access (remembering that shares most likely contain data)

However, a very common and easy to cause vulnerability by administrators is wrong permissions. You may be able to access a share and its data without logging in at all, such as we will demonstrate below:

1. Remember that the IP address of the Samba server is that of the Instance you deployed (**10.10.54.135**)

2. Use the *smbclient* tool to begin accessing the Samba server and its shares, replacing "**sharename**" with the name of the share you wish to access: `smbclient //REPLACE_INSTANCE_IP_ADDRESS/**sharename**`

3. You will be asked for a password, the easiest password is no password! **We can just press "Enter"** to test this theory. If successful, this means that the share requires no authentication and we are now logged in.

*For example, accessing **"share1"** on another device:*

```
root@kali:~# smbclient //192.168.1.200/share1
Enter WORKGROUP\root's password:
Try "help" to get a list of possible commands.
smb: \>
```

You can use the `help` command to list some of the commands you can run whilst connected to the Samba share. Here's a quick rundown of the fundamentals:

Command	Description
ls	List files and directories in the current location
cd <directory>	Change our working directory
pwd	Output the full path to our working directory
more <filename>	Find out **more** about the contents of a file. To close the open file, you press `:q`
get <filename>	Download a file from a share
put <filename>	Upload a file from a share

You can now proceed to answer **Question #3** and **Question #4**

10.6. Conclusion, where to go from here and additional Material:

You've learned the fundamentals of how a very commonplace protocol used by computing devices works, and ultimately, can be leveraged through the use of enumeration and misconfiguration. With this said, you might be surprised to learn that even printers can use the protocols behind Samba. Swafox has created a lovely room on Printer Hacking 101.

There's no truer statement in pentesting that practice makes perfect. Not only can you use the tools within this room, why not give a few others a try and apply your knowledge in the "Kenobi" Walkthrough room or the "Anonymous"Challenge room (CTF)

“users” on samba server:

```
 ==========================================================
|    Enumerating Workgroup/Domain on 10.10.54.135          |
 ==========================================================
[+] Got domain/workgroup name: TBFC-SMB-01

 ======================================
|    Session Check on 10.10.54.135     |
 ======================================
[+] Server 10.10.54.135 allows sessions using username '', password ''

 ============================================
|    Getting domain SID for 10.10.54.135     |
 ============================================
Domain Name: TBFC-SMB-01
Domain Sid: (NULL SID)
[+] Can't determine if host is part of domain or part of a workgroup

 ============================
|    Users on 10.10.54.135   |
 ============================
index: 0x1 RID: 0x3e8 acb: 0x00000010 Account: elfmcskidy       Name:   Desc:
index: 0x2 RID: 0x3ea acb: 0x00000010 Account: elfmceager       Name: elfmceager        Desc:
index: 0x3 RID: 0x3e9 acb: 0x00000010 Account: elfmcelferson    Name:   Desc:

user:[elfmcskidy] rid:[0x3e8]
user:[elfmceager] rid:[0x3ea]
user:[elfmcelferson] rid:[0x3e9]
enum4linux complete on Wed Dec 16 01:13:23 2020

[kafka@kafka ~]$
```

“shares” on samba server:

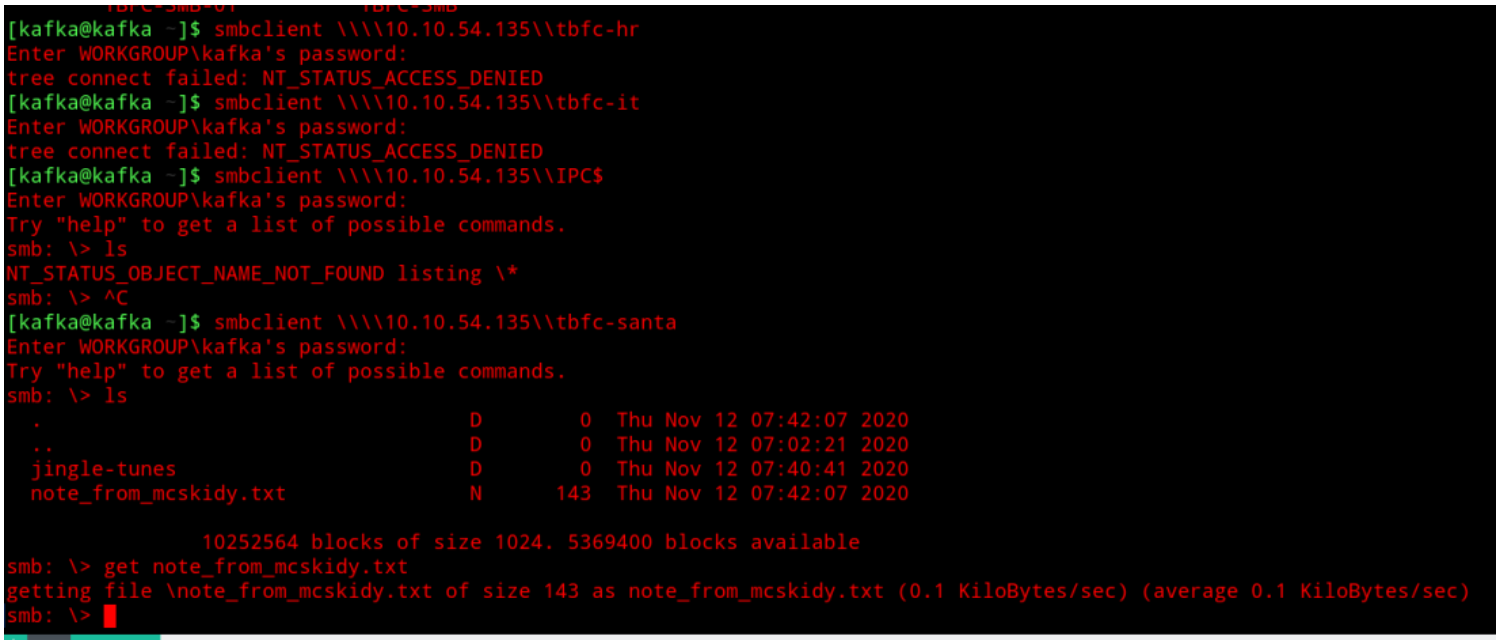
```
[kafka@kafka ~]$ smbclient -L 10.10.54.135
Enter WORKGROUP\kafka's password:

        Sharename       Type      Comment
        ---------       ----      -------
        tbfc-hr         Disk      tbfc-hr
        tbfc-it         Disk      tbfc-it
        tbfc-santa      Disk      tbfc-santa
        IPC$            IPC       IPC Service (tbfc-smb server (Samba, Ubuntu))
Reconnecting with SMB1 for workgroup listing.

        Server               Comment
        ---------            -------

        Workgroup            Master
        ---------            -------
        TBFC-SMB-01          TBFC-SMB
[kafka@kafka ~]$
```

trying to login with easy/no passwords:


[kafka@kafka ~]$ smbclient \\\\10.10.54.135\\tbfc-hr
Enter WORKGROUP\kafka's password:
tree connect failed: NT_STATUS_ACCESS_DENIED
[kafka@kafka ~]$ smbclient \\\\10.10.54.135\\tbfc-it
Enter WORKGROUP\kafka's password:
tree connect failed: NT_STATUS_ACCESS_DENIED
[kafka@kafka ~]$ smbclient \\\\10.10.54.135\\IPC$
Enter WORKGROUP\kafka's password:
Try "help" to get a list of possible commands.
smb: \> ls
NT_STATUS_OBJECT_NAME_NOT_FOUND listing *
smb: \> ^C
[kafka@kafka ~]$ smbclient \\\\10.10.54.135\\tbfc-santa
Enter WORKGROUP\kafka's password:
Try "help" to get a list of possible commands.
smb: \> ls
. D 0 Thu Nov 12 07:42:07 2020
.. D 0 Thu Nov 12 07:02:21 2020
jingle-tunes D 0 Thu Nov 12 07:40:41 2020
note_from_mcskidy.txt N 143 Thu Nov 12 07:42:07 2020
10252564 blocks of size 1024. 5369400 blocks available
smb: \> get note_from_mcskidy.txt
getting file \note_from_mcskidy.txt of size 143 as note_from_mcskidy.txt (0.1 KiloBytes/sec) (average 0.1 KiloBytes/sec)
smb: \>