Null Session Enumeration from a Windows-Based System

The first step in enumerating CIFS/SMB is to connect to the service using the so-called **null session** command, which you will do in the following exercise.

Exercise 1: Creating a null session from your Windows attack system:

1. From a Windows attack system command shell, type the following (only type what's in **bold**):

   C:\>**net use \\victim_IP_address\ipc$ "" /u:""**

   Syntax breakdown:
   net use: program name
   **\\victim_IP_address\ipc$**: the Universal Naming Convention (UNC) used to connect to the victim system's hidden interprocess communication share (IPC$)
   "": use a null password
   /u:"": program option for the built-in anonymous user

   Notice the similarity between this command and the standard *net use* syntax for mounting a network drive (they're identical)

2. If the command completes successfully, you now have an open channel over which to attempt various techniques to pillage as much information as possible from the target (e.g., network information, shares, users, groups, Registry keys, etc.)

3. To disconnect your null session connection, type the following:

   C:\>**net use \\victim_IP_address\ipc$ /d**

Null Session Enumeration Using WinScanX

WinScanX is a free tool used for querying Windows service information over SMB. It combines many of the essential tools used during a penetration test into a single utility.

In addition to a GUI version of the utility (which requires the Microsoft .NET Framework), it can also be run from a command-line.

Exercise 1: Null session enumeration using WinScanX: in this exercise, you will run the GUI version of WinScanX to further enumerate your target system:

1. On your Windows attack system, disconnect any null session connections

2. Install WinScanX. WinScanX is available in the Blackboard to download.

3. Next open WinScanX: Start/All Programs/WinScanX/WinScanX GUI

4. Enter the Windows target system's IP address in the Target Host field (upper left-hand corner of application)

5. Under the WinScanX Connection Settings, select Use Anonymous Connection

6. In addition, select the following settings:

    - Get Account Policy Information
    - Get Audit Policy Information
    - Get Display Policy Information
    - Get Domain Information
    - Get Administrative Local & Global Group Information
    - Get Local & Global Group Information
    - Get Installed Programs
    - Get Logged On Users
    - Get Patch Information
    - Get Registry Information
    - Get Scheduled Task Information
    - Get Server Information
    - Get Service Information
    - Get Share Information
    - Get Share Permissions
    - Get User Information
    - Get User Information via RA Bypass
    - Get User Rights Information
    - Save Remote Registry Hives

7. Click the Start Scan button

8. Scroll through the output in the WinScanX Output window

9. You can also view the output of the scan by clicking the View Reports button and selecting the various reports of interest (these reports are saved in the following location: C:\Temp\Class_Tools\WinScanX\Reports)

10. Repeat step #3, this time selecting Specify Username & Password (use the Administrator user account and password you enumerated in lab #23 and PTEST as the domain name)

11. Which scan yielded more information?

SNMP Enumeration Using snmpcheck

snmpcheck is a PERL script used to enumerate information on systems that are running SNMP. It's located in the BackTrack distro */pentest/enumeration/snmpcheck/* folder.

Exercise 1: **SNMP Enumeration using snmpcheck**: in this exercise, you'll use snmpcheck from the BackTrack distribution to dump information about specific parameters on the SNMP agent (device):

1. From a BackTrack shell, type the following (only type what's in **bold**, on one line):

user1@pentest:~# **perl /pentest/enumeration/snmpcheck/snmpcheck.pl -t target_IP_address -c public > /root/ceh/snmpcheck**

Syntax breakdown:
perl: PERL scripting language program name
/pentest/enumeration/snmpcheck/snmpcheck.pl: location of script name
-t target_IP_address: the IP address of the target system
-c public: the community string is public
> /root/ceh/snmpcheck: redirect the output to a file called snmpcheck in the /root/ceh directory

2. Examine your results:

user1@pentest:~#**cat /root/ceh/snmpcheck | less**

3. List items of interest:

4. Repeat step #1: **NOTE**: replace *target_IP_address* with that of another victim system's IP address. Also, change the name of the redirected output to */root/ceh/snmpcheck2*

5. Examine your results:

user1@pentest:~#**cat /root/ceh/snmpcheck2 | less**

6. List items of interest:

MySQL Enumeration

WHAT YOU WILL SEE MAY BE SLIGHTLY DIFFIERENT FROM THE FOLLOWING.

Exercise 1: **guessing MySQL root account password**: in this exercise, you will guess passwords for the MySQL root user account:

1. Download and install MySQL database in your victim machine (Window2K3) Test if you are able to run basic commands such as *show databases*.

2. From a BackTrack shell, type the following (only type what's in **bold**):

user1@pentest:~# **mysql -h target_IP_address -u root -p**

Syntax breakdown:
mysql: program name
-h target_IP_address: program option specifying the IP address of the target system
-u root: program option specifying the root username
-p: program option prompting for root password

3. When prompted, type in a password. Continue until you have guessed the MySQL root password and record it here:

4. You know you successfully guessed the MySQL password if you see the following (**NOTE**: your output may vary depending on the version of MySQL):

Welcome to the MySQL monitor. Commands end with ; or \g.
Your MySQL connection id is 2
Server version: 5.0.77 Source distribution

Type 'help;' or '\h' for help. Type '\c' to clear the buffer.

mysql>

Exercise 2: **enumerating MySQL**: in this exercise, you will use various MySQL commands to try and enumerate database names, tables within a database, the structure of the table, and other valuable information (e.g., usernames):

1. From your MySQL prompt, type the following (only type what's in **bold**):

mysql>**show databases;**

Syntax breakdown:
show databases; program option to list all databases on the MySQL server

Your output will look as follows:

```
+--------------------------------+
| Database                       |
+--------------------------------+
| information_schema             |
| Avalon                         |
| test                           |
+--------------------------------+
3   rows in set (0.11 sec)
```

There are three databases on the victim MySQL server: *information_schema, Avalon, and test*

2. From your MySQL prompt, type the following (only type what's in **bold**):

mysql>**use Avalon;**

Syntax breakdown:
use Avalon; program option to switch to the database named Avalon

Your output will look as follows:

Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A

Database changed

3. From your MySQL prompt, type the following (only type what's in **bold**):

mysql>**show tables;**

Syntax breakdown:
show tables; program option to list all the tables in the Avalon database

Your output will look as follows:

```
+---------------------------------+
| Tables_in_Avalon                |
+---------------------------------+
| Customers                       |
+---------------------------------+
1 row in set (0.00 sec)
```

There is one table in the Avalon database named *Customers*

4. From your MySQL prompt, type the following (only type what's in **bold**):

mysql>**describe Customers;**

Syntax breakdown:
describe Customers; program option to list the Customers table's format

Your output will look as follows:

```
+--------------+-------------+------+-----+---------+----------------------------------------+
| Field        |        Type | Null |       Key | Default | Extra |
+--------------+-------------+------+-----+---------+----------------------------------------+
| firstname    |     varchar(20) | YES |                   | NULL|          |
| lastname     |     varchar(40) | YES |                   | NULL|          |
| organization|     varchar(60) | YES |                   | NULL|          |
| address1     |     varchar(40) | YES |                   | NULL|          |
| address2     |     varchar(40) | YES |                   | NULL|          |
| city         |     varchar(40) | YES |                   | NULL|          |
| state        |     char(2)     | YES |                   | NULL|          |
| postal_code|     char(10)    | YES |                   | NULL|          |
| ssn          |     char(11)    | YES |                   | NULL|          |
+--------------+-------------+------+-----+---------+----------------------------------------+
9 rows in set (0.00 sec)
```

There are nine fields of varying types (e.g., ssn is a field for Social Security Numbers of size 11 bytes/characters)

5. From your MySQL prompt, type the following (only type what's in **bold**):

mysql>**select * FROM Customers;**

Syntax breakdown:
select * FROM Customers; program option to show all the data in the Customers table

6. Notice any data of value?

7. From your MySQL prompt, type the following (only type what's in **bold**):

mysql>**exit;**

Syntax breakdown:
exit; program option to exit MySQL command line

Exercise 3: **copying victim's MySQL databases to your local system**: in this exercise, you will make a copy of the victim's MySQL databases on your system:

1. From a BackTrack shell, type the following (only type what's in **bold**, on one line):

user1@pentest:~# **mysqldump -h target_IP_address -u root -p -A > /root/ceh/alldatabases.sql**

Syntax breakdown:
mysqldump: program name
-h target_IP_address: program option specifying the IP address of the target system
-u root: program option specifying the root username
-p: program option prompting for root password
-A > /root/ceh/alldatabases.sql: program option to dump all the databases on victim MySQL server to the */root/ceh/* folder in a file called *alldatabases.sql*

You now have a local copy of the victim's MySQL databases.

Determining the Password Policy by Guessing the Guest Account Password

The cleanest way to determine the lockout policy of a remote Windows system is to enumerate it via a null session (and run another tool like WinScanX). If you can't do that, attempting password guesses against the Guest account is the next best step to perform. The Guest account was disabled by default starting with WindowsXP. However, if you reach the lockout threshold, you will be notified nonetheless. Once the lockout threshold has been exceeded, the next guess tells you that the Guest account is locked out, even though it is disabled.

Exercise 1: Guessing the Guest account password to determine the password policy on the target system: in the following exercise, you will use a slight variation of the null session connection syntax to guess the number of failed login attempts before an account is locked (if such a policy exists):

1. From your Windows victim system (Windows2k3) set the password policy to lock guest account after 3 failed password guessing. If you do not know how to do this, do some research to figure out.

2. From a Windows attack system command shell, type the following (only type what's in **bold**):

C:\> **net use \\target_IP_address\ipc$ * /u:guest**

Syntax breakdown:
net use: program name
**\\target_IP_address\ipc$**: the Universal Naming Convention (UNC) used to connect to the target system's hidden interprocess communication share (IPC$)
*****: prompt for a password
/u: program option to specify a user account to use for the IPC$ share connection
guest: use the Guest user account for the IPC$ share connection

3. When prompted for a password, type in whatever you want (don't use a null password)

4. Repeat steps #1-2 – all the while, counting the number of failed password guessing attempts you are allowed before given the following message:
   System error 1909 has occurred.
   The referenced account is currently locked out and may not be logged on to.

5. How many failed login attempts do you get on the target system? Does it match with your setting in the first step?

6. Switch to your Windows target system

7. Click Start/All Programs/Administrative Tools/Event Viewer

8. Open the Security log and examine the failed login attempts, as well as the Account Lockout message, if any.