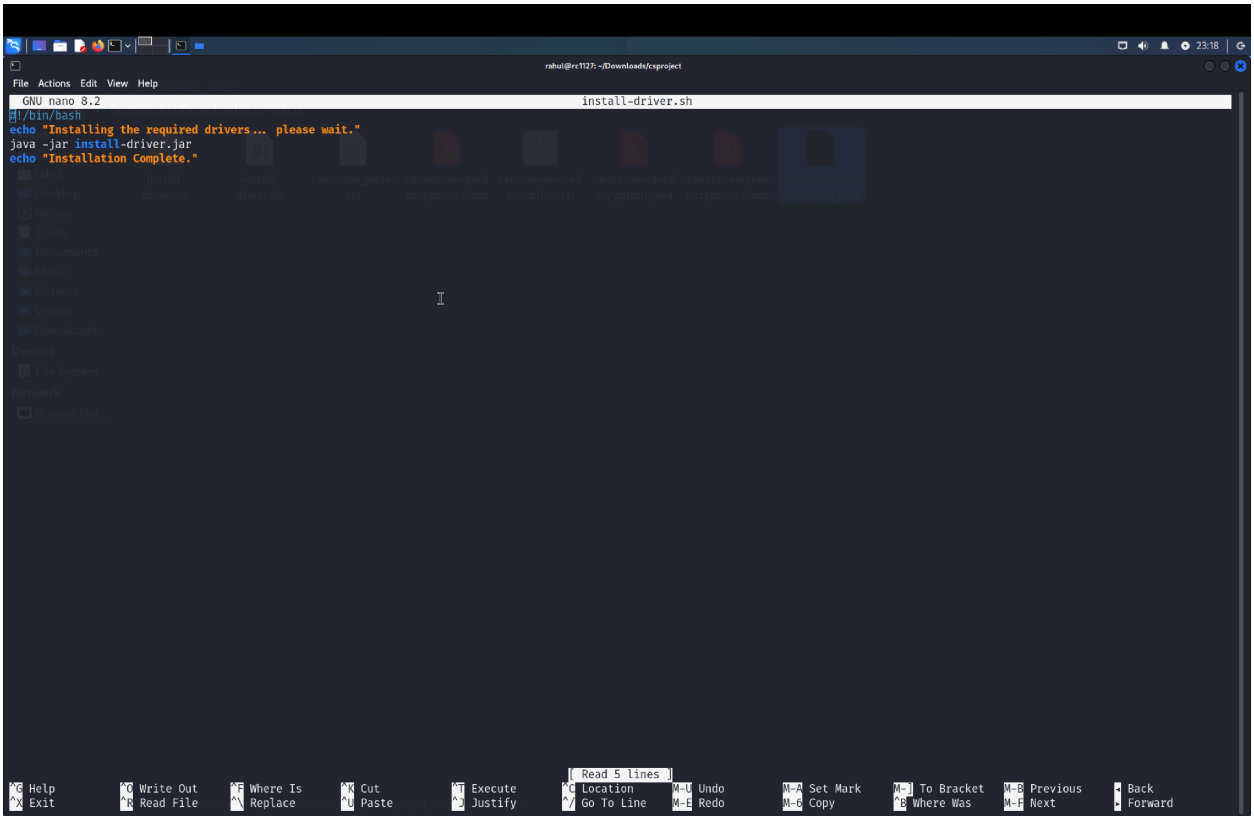
Group: 12

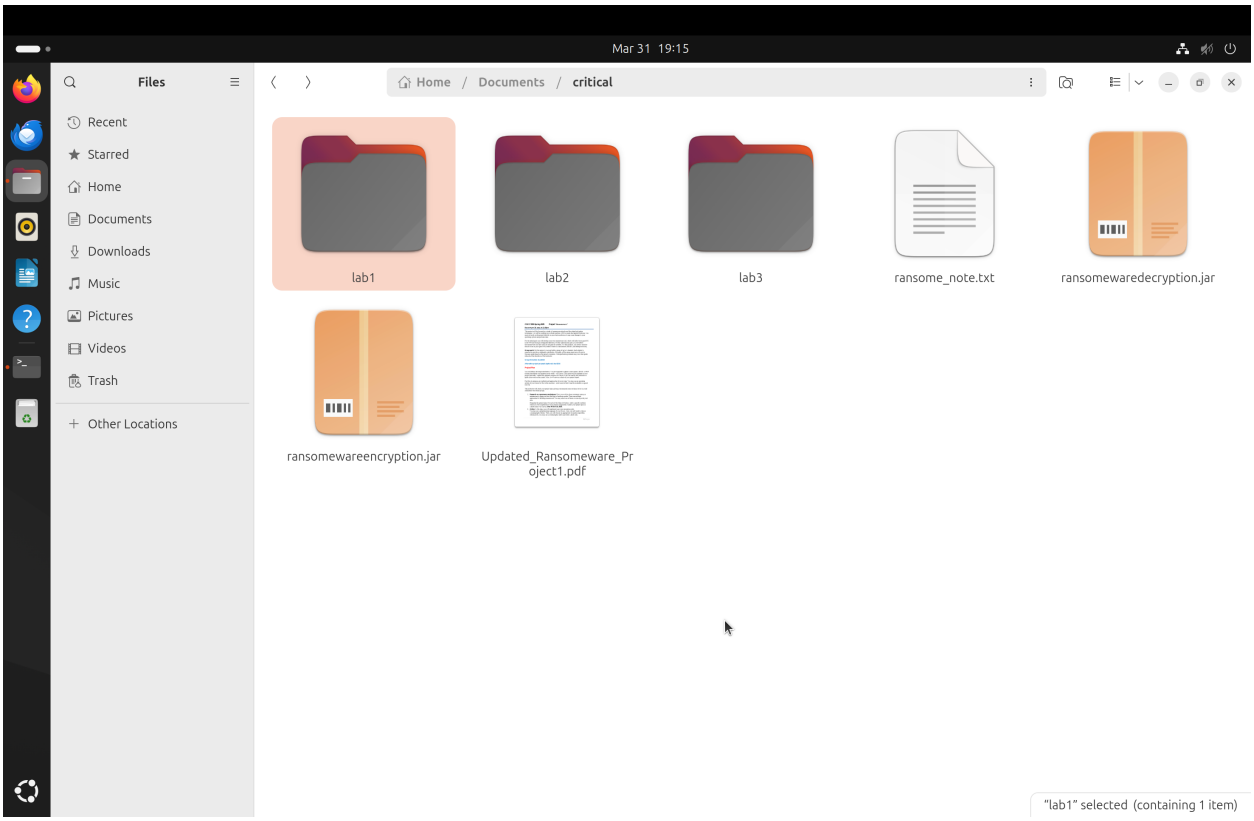
Proposed Infection Method: Malicious Bash Script Disguised as a Utility

Install-driver.sh code

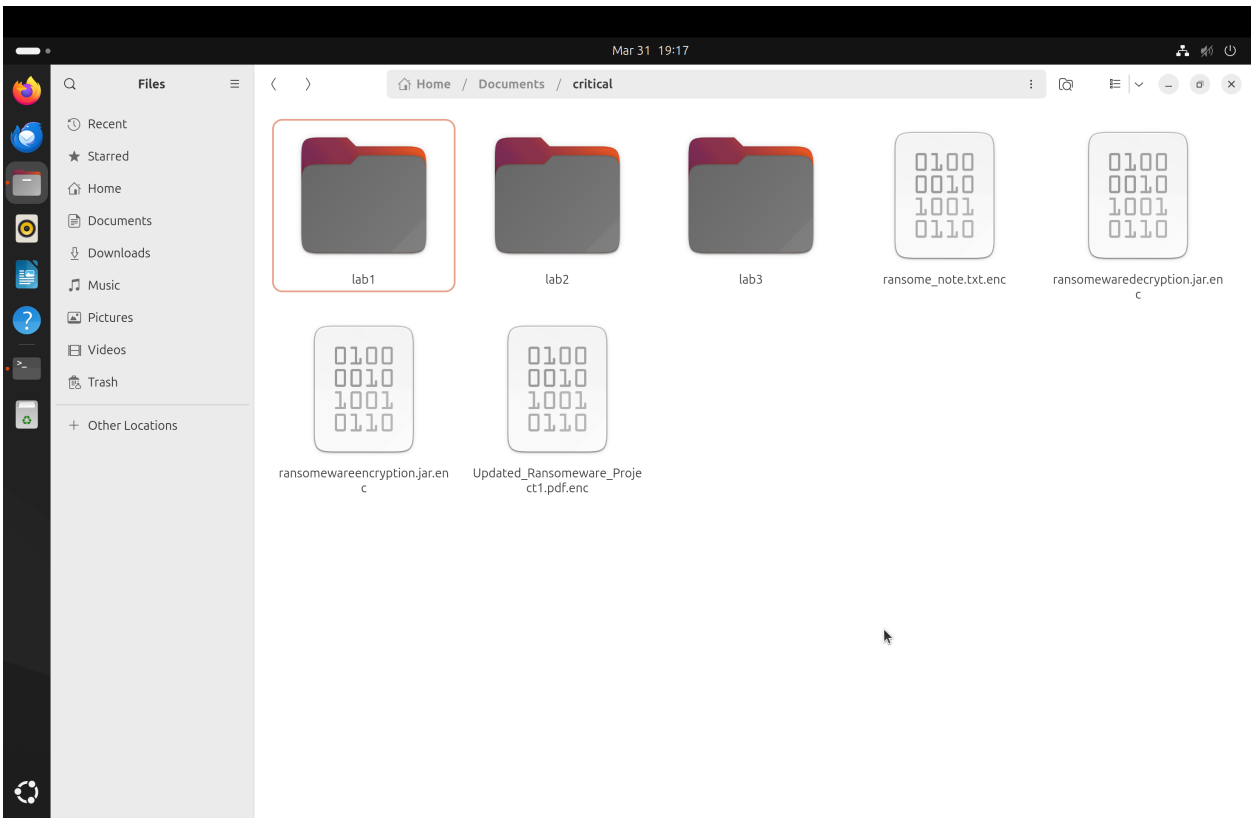


Before the ransomware implementation in the victim's machine.

Group: 12



After the implementation of ransomware in victims' machines.



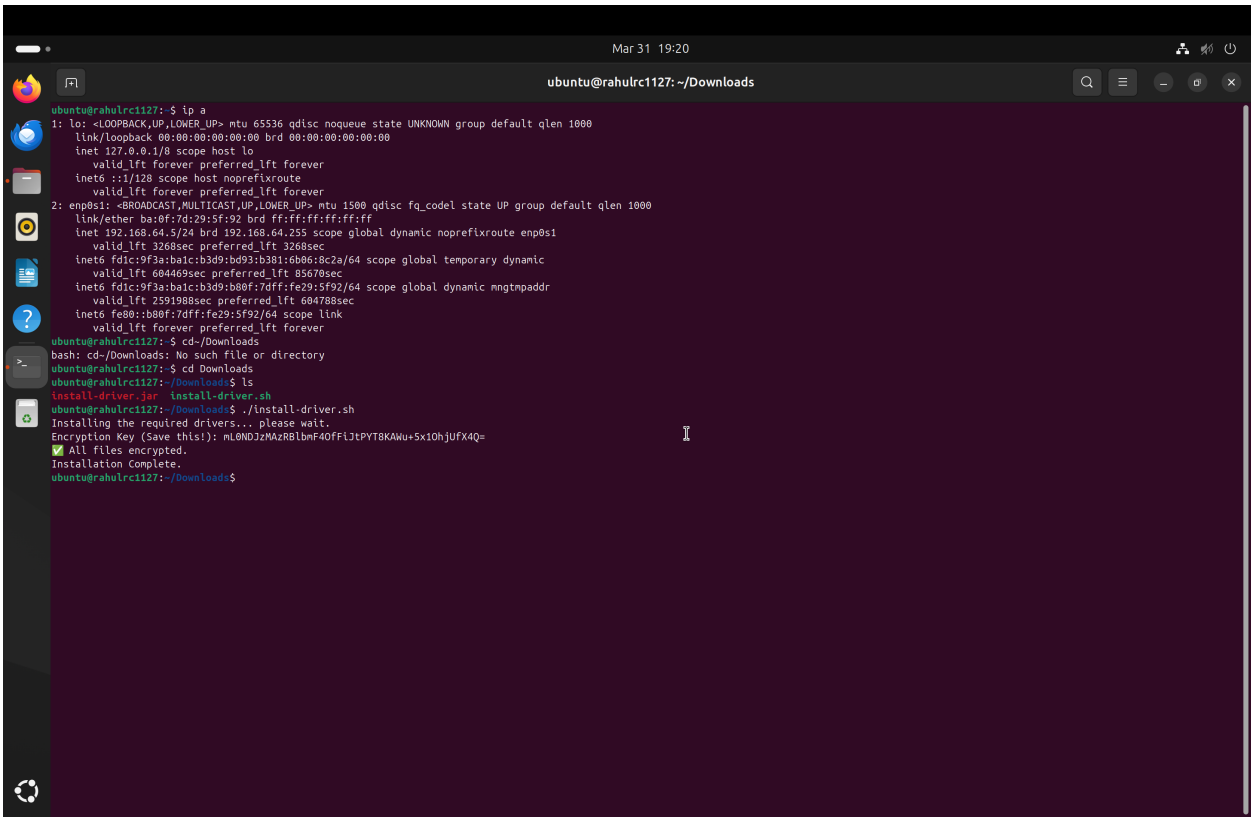
Group: 12

The attack was done by an attacker through Kali Linux:

```
rahul@rc1127: ~/Downloads/csproject
File  Actions  Edit  View  Help
┌──(rahul㉿rc1127)-[~]
└─$ mv ransomewareencryption.jar install-driver.jar
mv: cannot stat 'ransomewareencryption.jar': No such file or directory

┌──(rahul㉿rc1127)-[~]
└─$ ls
'Computer Security Lab 1.pcapng'  'Computer Security Lab.pcapng'  Desktop  Documents  Downloads  Music  Pictures  Public  Templates  Videos

┌──(rahul㉿rc1127)-[~]
└─$ cd /home/rahul/Downloads/csproject/

┌──(rahul㉿rc1127)-[~/Downloads/csproject]
└─$ ls
ransome_note.txt  ransomewaredecryption.class  ransomewaredecryption.jar  ransomewaredecryption.java  ransomewareencryption.class  ransomewareencryption.jar  ransomewareencryption.java

┌──(rahul㉿rc1127)-[~/Downloads/csproject]
└─$ mv ransomewareencryption.jar install-driver.jar

┌──(rahul㉿rc1127)-[~/Downloads/csproject]
└─$ ls
install-driver.jar  ransome_note.txt  ransomewaredecryption.class  ransomewaredecryption.jar  ransomewaredecryption.java  ransomewareencryption.class  ransomewareencryption.java

┌──(rahul㉿rc1127)-[~/Downloads/csproject]
└─$ nano install-driver.sh

┌──(rahul㉿rc1127)-[~/Downloads/csproject]
└─$ chmod +x install-driver.sh

┌──(rahul㉿rc1127)-[~/Downloads/csproject]
└─$ nano install-driver.sh

┌──(rahul㉿rc1127)-[~/Downloads/csproject]
└─$ scp install-driver.sh ubuntu@192.168.64.5:/home/ubuntu/Downloads
The authenticity of host '192.168.64.5 (192.168.64.5)' can't be established.
ED25519 key fingerprint is SHA256:pB7hTYclG1Hgs+yUu8CkeLFVJS9I0Hd70W461sojvaU.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '192.168.64.5' (ED25519) to the list of known hosts.
ubuntu@192.168.64.5's password:
install-driver.sh                                                    100%  127   128.9KB/s   00:00

┌──(rahul㉿rc1127)-[~/Downloads/csproject]
└─$ scp install-driver.jar ubuntu@192.168.64.5:/home/ubuntu/Downloads
ubuntu@192.168.64.5's password:
install-driver.jar                                                   100% 2347   741.0KB/s   00:00

┌──(rahul㉿rc1127)-[~/Downloads/csproject]
└─$
```

The Victim's Machine:

Group: 12



Objective:

The memorandum supplies information about how ransomware will spread during the simulation research. The design presents a realistic educational delivery model that uses a legitimate Bash script to transmit ransomware to victim systems.

Proposed Infection Method:

A Bash script titled install-driver.sh carries the ransomware and takes the form of a system utility script meant to accomplish device driver installation. The ransomware payload containing encryption code (install-driver.jar) takes the form of a Java archive produced during Step 2 of the project development.

The victim executes the Bash script while seeing a pretended system notification showing "Installing drivers... please wait" to present a regular utility process. During its execution the script quietly operates the ransomware JAR file to start file encryption tasks in the attacked directory.

Implementation Process:

Disguising the Payload:

The initial JAR ransomware program gets renamed as install-driver.jar to make it appear like an ordinary installer application.

Group: 12

Creating the Script:

The Bash script install-driver.sh contains simple echo messages in addition to running the JAR file through java -jar execution command.

Delivery to Victim:

The files arrive at the victim computer's Downloads directory thus generating the appearance of a system copy or download.

Victim Execution:

The victim operates the script under the belief that it is an authentic utility. The ransomware attacks occur as it encrypts all data stored in the ~/Documents/critical directory.

This method models a realistic social engineering scenario where users are tricked into running malicious scripts, especially in Linux-based environments where Bash scripts are commonly used. It does not rely on advanced exploits or phishing infrastructure, making it ideal for controlled simulation and educational demonstration.

Conclusion:

The infection approach successfully shows how harmful scripts can trick users into running them through utility scripts. The infection methodology implements genuine attack sequences that suit the ransomware simulation project requirements for its infective phase.