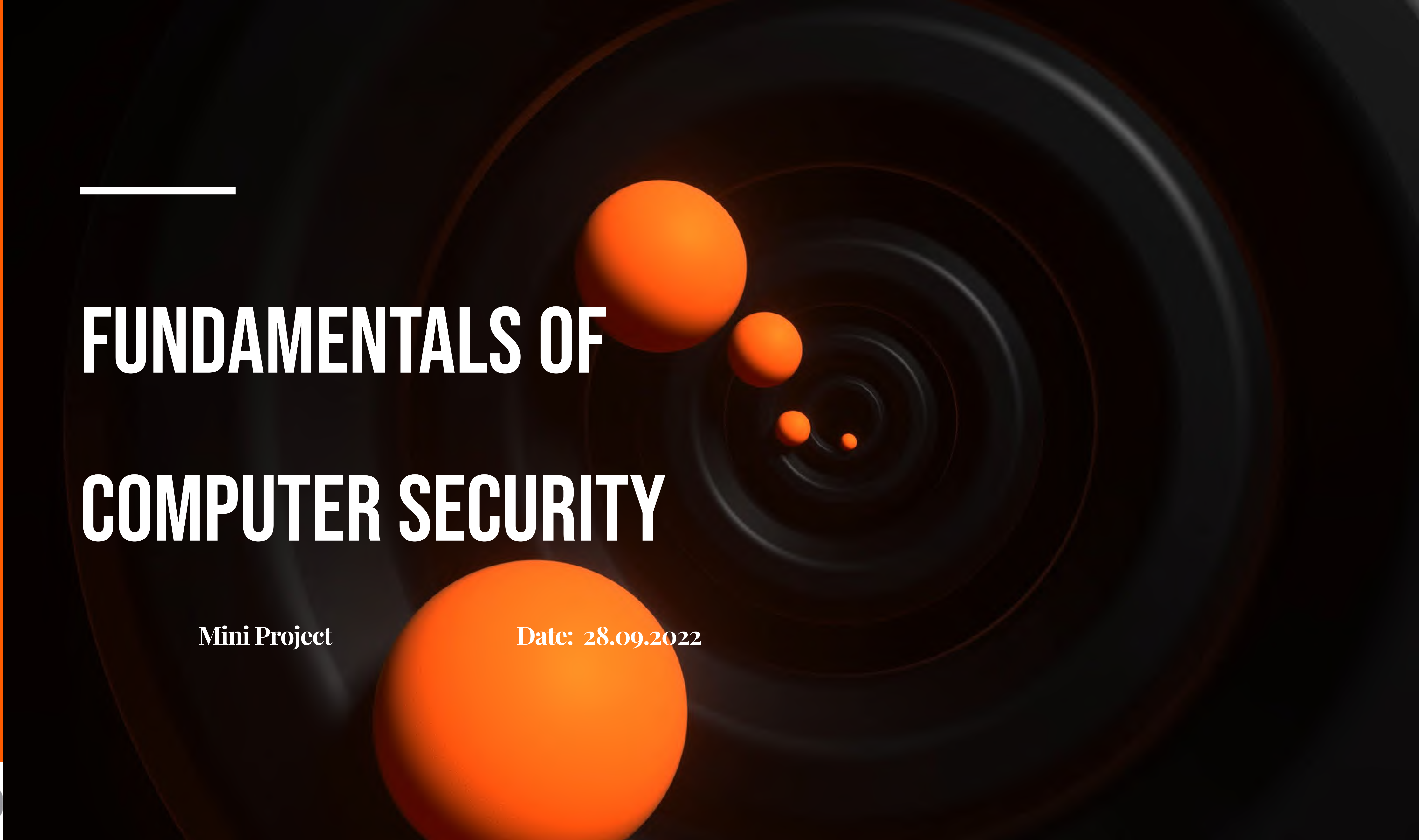
PASSWORD CRACKING USING JOHN_THE_RIPPER



# FUNDAMENTALS OF COMPUTER SECURITY

Mini Project

Date: 28.09.2022

PASSWORD CRACKING USING JOHN_THE_RIPPER

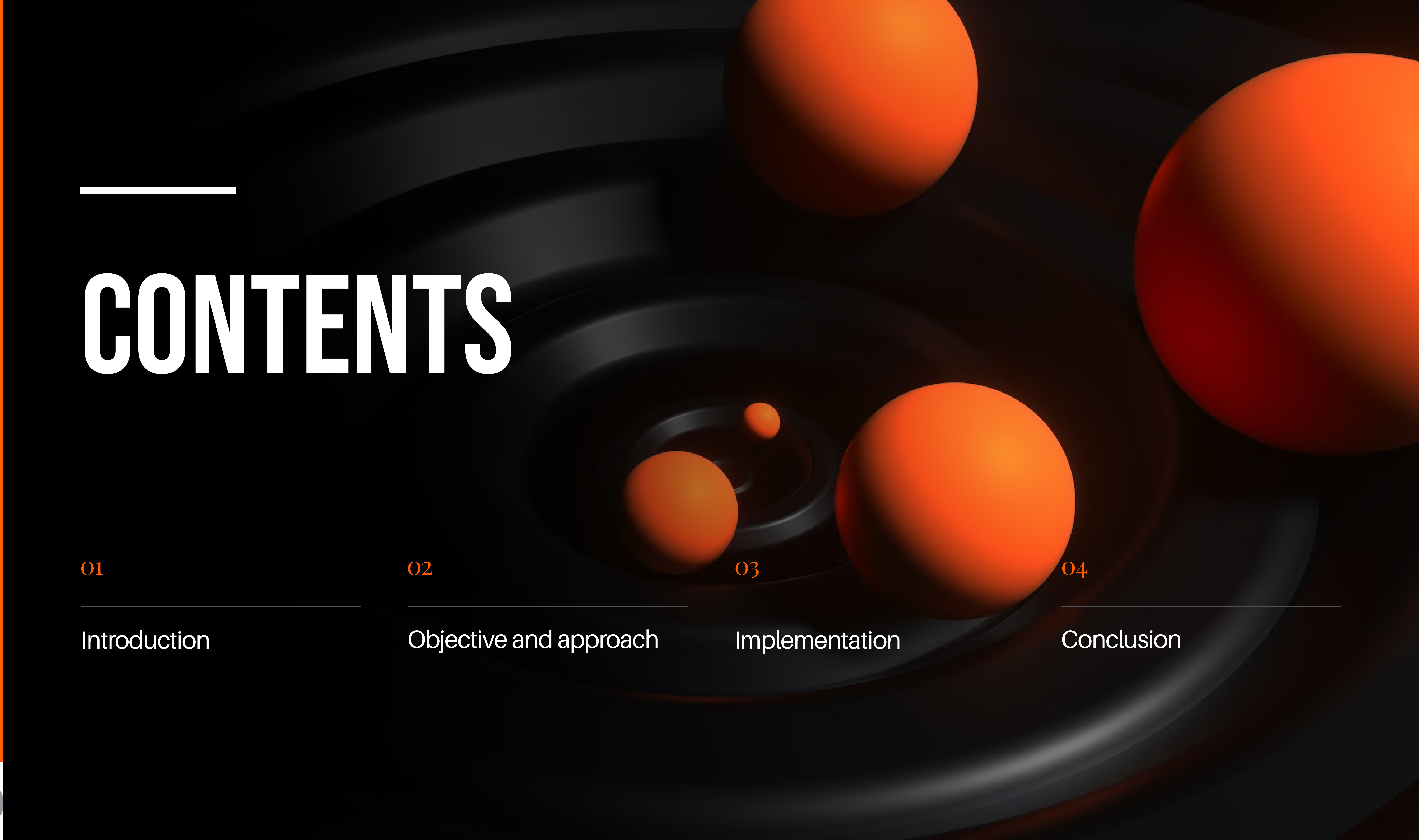
TOPIC

# PASSWORD CRACKING USING JOHN-THE-RIPPER TOOL

## GROUP MEMBERS:

1. YASHASVI SHARMA [9920103149]
2. PRAKHAR PRATAP SINGH [9920103124]
3. AAYUSH BHARDWAJ [9920103136]
4. IESH PRAVEEN PUNDE [9920103121]

## SUBMITTED TO:
## PROF. SHARIQ MURTUZA



# CONTENTS

01

Introduction

02

Objective and approach

03

Implementation

04

Conclusion

PASSWORD CRACKING USING JOHN_THE_RIPPER

# INTRODUCTION TO OUR TOPIC

## SO WHY WE USING THIS TOOL?

As Password cracking is the process of attempting to gain unauthorized access to restricted systems using common passwords or algorithms that guess passwords. In other words, it is the art of getting the right password that allows access to a system protected by an authentication method.

We are demonstrating a tool called Jack The Ripper tool which helps in the recovery of the password, in case you forget your password, mention ethical hacking professionals.

As we are doing it on small scale JtR can help IT staff spot weak passwords and poor password policies for their company.

# OBJECTIVE & APPROACH

## Objective

John the Ripper is  used in the enterprise to detect weak passwords that could put network security at risk, as well as other administrative purposes as it combines a number of password crackers into one package, autodetects password hash types, and includes a customizable cracker

## Approach

During the cracking process, John the Ripper uses a rainbow table approach where he takes words from the built-in dictionary that comes with it.

It then compiles variants of this dictionary and compares the hashed password to what is in the password file and tries to find a match. This is repeated until a match is found.

John the Ripper identifies all potential passwords in a hashed format. It then compares the hashed passwords to the initial hashed password and tries to find a match.

If a match is found in the password hash, John the Ripper will display the password in its raw form as a cracked password. The process of comparing hashes of passwords to find a match is called a dictionary attack.

# INSTALLATION

## Command prompt installation from Source

Open the terminal by holding Ctrl+Alt+T simultaneously and run the below command.

```
mkdir src
```

This will create a directory where we will store all our files.

Go to the src directory and clone the John the Ripper repository as shown below.

```
cd src
git clone https://github.com/openwall/john.git
```

Cloning the John the Ripper repository

This will create a directory called John. To make it active, we need to run the below command.

```
cd john
```

Go to the src directory where we will set up and configure the compilation sources.

```
cd src
./configure
```

# INSTALLATION

## Command prompt installation from package

```
./configure
```

Configuration files in the src directory

Run the make command to compile the source code into executable programs and libraries. This may take some time depending on your computer and the resources allocated to it.

```
make
```

Finally, run make install to install John the Ripper.

perform the installation

Run the make install command

Run the commands below to see if the installation was successful.

# INSTALLATION

## Command prompt installation from Source

```
cd ..
cd run
./john
```

If  the tool has been installed correctly without any errors then the installed version with other details of the tool will be shown on the terminal.

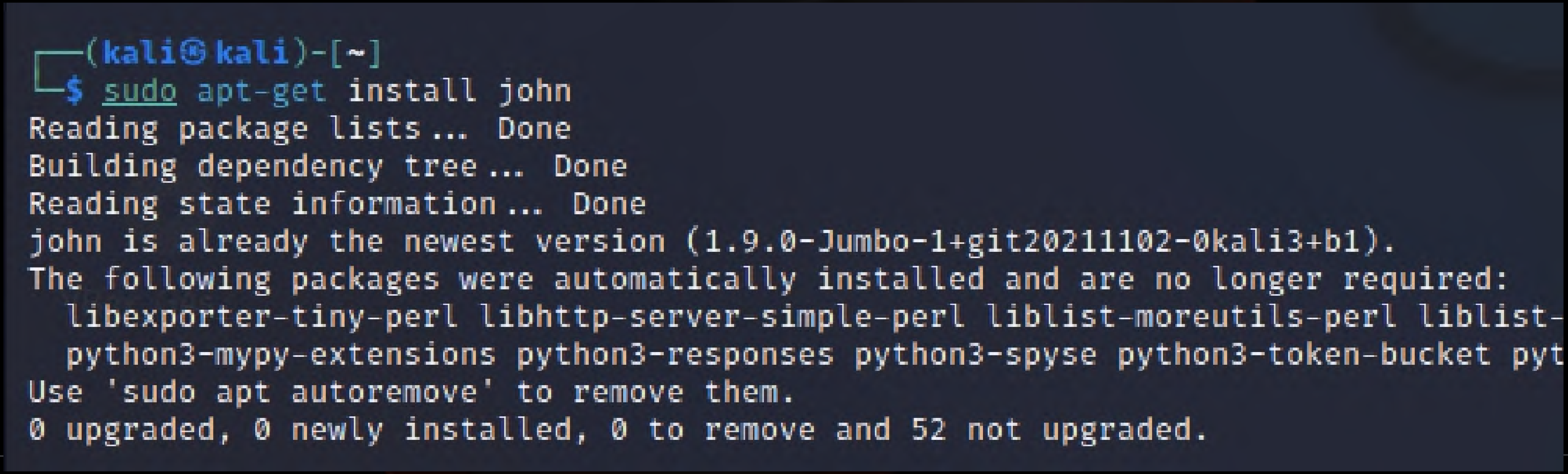
PASSWORD CRACKING USING JOHN_THE_RIPPER

# INSTALLATION

## Installation from Package

Since installing the tool from source is quite a lengthy process we can skip that by installing john the ripper from a pre loaded package using the command

```
sudo apt install john
```



```
┌──(kali㉿kali)-[~]
└─$ sudo apt-get install john
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
john is already the newest version (1.9.0-Jumbo-1+git20211102-0kali3+b1).
The following packages were automatically installed and are no longer required:
  libexporter-tiny-perl libhttp-server-simple-perl liblist-moreutils-perl liblist-
  python3-mypy-extensions python3-responses python3-spyse python3-token-bucket pyt
Use 'sudo apt autoremove' to remove them.
0 upgraded, 0 newly installed, 0 to remove and 52 not upgraded.
```

# USES OF JOHN THE RIPPER TOOL

John the Ripper is often used in the enterprise **to detect weak passwords that could put network security at risk, as well as for other administrative purposes.** In this presentation, we are going to use the john tool to crack the passwords of

Linux account

Zip file

# HASHING

Hashing is the process of generating a value from a text or a list of numbers using a mathematical function known as a hash function. Hashing turns your password (or any other piece of data) into a short string of letters and/or numbers using an encryption algorithm. There are many hashing functions but the ones we generally dealt with in old Linux versions were

MD5

SHA256

Now we deal with a more advanced type of encryption known as yescrypt.

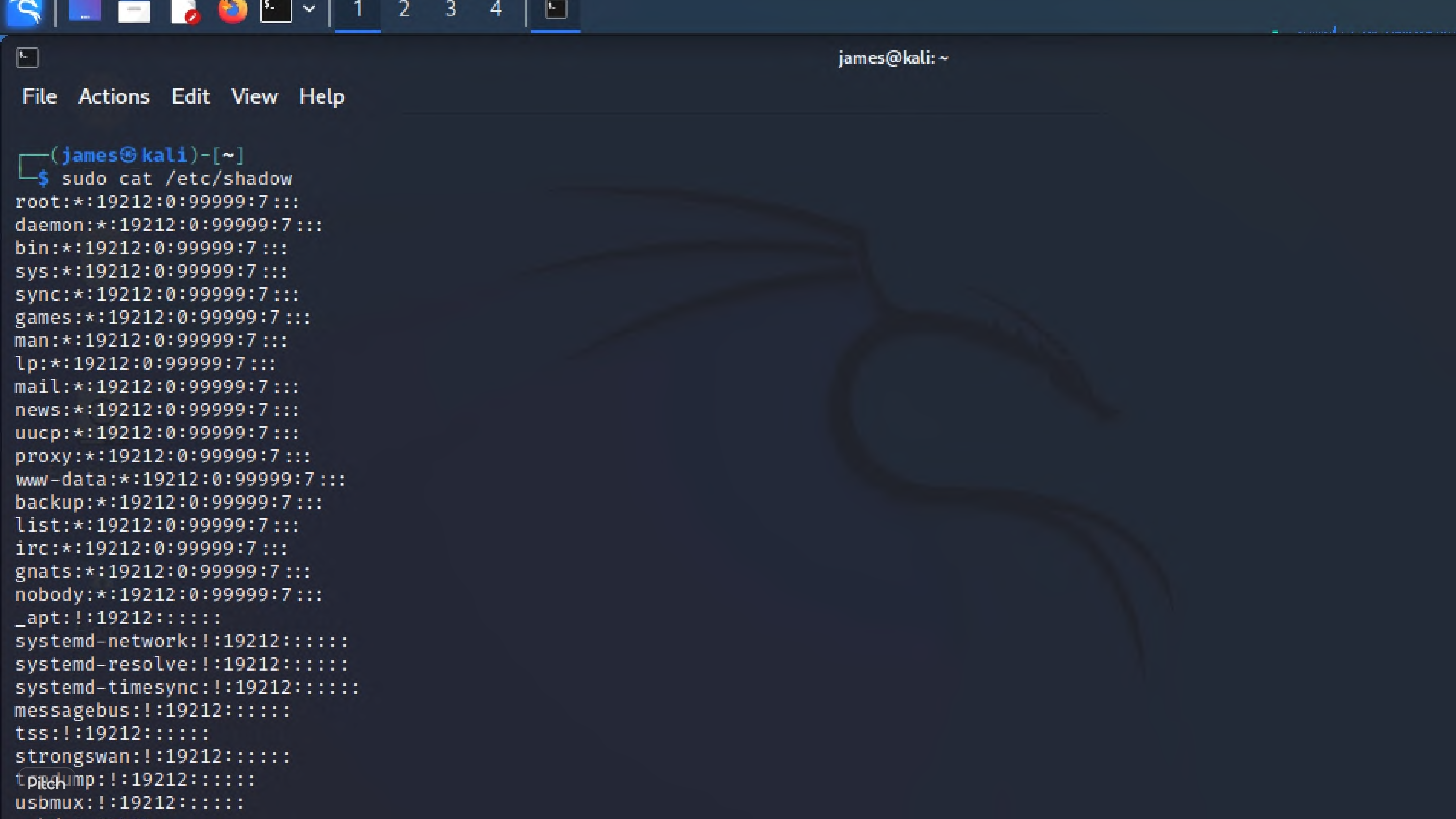
# /ETC/SHADOW AND /ETC/PASSWD

A shadow password file, also known as */etc/shadow,* is a system file in Linux that stores encrypted user passwords and is accessible only to the root user, preventing unauthorized users or malicious actors from breaking into the system.

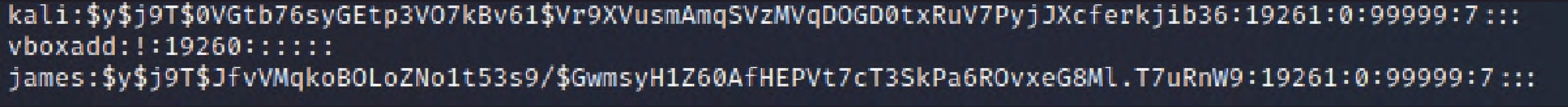
The common practice of storing passwords in the /etc/passwd file leaves the Linux system vulnerable to break-in attempts. To eliminate this vulnerability, newer Linux systems use the /etc/shadow file to store user passwords instead.

Traditional password files are maintained in /etc/passwd, but the actual hashed passwords are stored in /etc/shadow.

Hence John the ripper tool cracks the password stored in a shadow file in a hashed format.

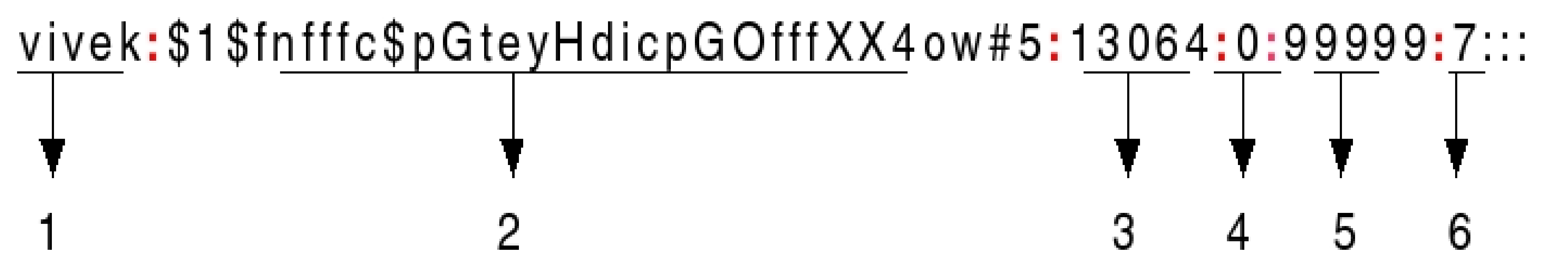


```
┌──(james㉿kali)-[~]
└─$ sudo cat /etc/shadow
root:*:19212:0:99999:7:::
daemon:*:19212:0:99999:7:::
bin:*:19212:0:99999:7:::
sys:*:19212:0:99999:7:::
sync:*:19212:0:99999:7:::
games:*:19212:0:99999:7:::
man:*:19212:0:99999:7:::
lp:*:19212:0:99999:7:::
mail:*:19212:0:99999:7:::
news:*:19212:0:99999:7:::
uucp:*:19212:0:99999:7:::
proxy:*:19212:0:99999:7:::
www-data:*:19212:0:99999:7:::
backup:*:19212:0:99999:7:::
list:*:19212:0:99999:7:::
irc:*:19212:0:99999:7:::
gnats:*:19212:0:99999:7:::
nobody:*:19212:0:99999:7:::
_apt:!:19212::::::
systemd-network:!:19212::::::
systemd-resolve:!:19212::::::
systemd-timesync:!:19212::::::
messagebus:!:19212::::::
tss:!:19212::::::
strongswan:!:19212::::::
tcpdump:!:19212::::::
usbmux:!:19212::::::
```



```
kali:$y$j9T$0VGtb76syGEtp3VO7kBv61$Vr9XVusmAmqSVzMVqDOGD0txRuV7PyjJXcferkjib36:19261:0:99999:7:::
vboxadd:!:19260::::::
james:$y$j9T$JfvVMqkoBOLoZNo1t53s9/$GwmsyH1Z60AfHEPVt7cT3SkPa6ROvxeG8Ml.T7uRnW9:19261:0:99999:7:::
```

# UNDERSTANDING A SHADOW FILE



```
vivek:$1$fnfffc$pGteyHdicpGOfffXX4ow#5:13064:0:99999:7:::
```

As with the /etc/passwd, each field in the shadow file is also separated with ":" colon characters as follows:

1. **Username** A valid account name, which exists on the system.
2. **Password:** Your encrypted password is in hash format. The password should be a minimum of 15-20 characters long including special characters, digits, lowercase alphabetic, and more.

The $id is the algorithm used On GNU/Linux as follows:

- **$1$** is MD5
- **$2a$** is Blowfish
- **$2y$** is Blowfish
- **$5$** is SHA-256
- **$6$** is SHA-512
- $y$ is yescrypt

3. **Last password change (last changed)**: The date of the last password change, expressed as the number of days since Jan 1, 1970 (Unix time). The value 0 has a special meaning, which is that the user should change her password the next time she will log in to the system. An empty field means that password aging features are disabled.
4. **Minimum**: The minimum number of days required between password changes i.e. the number of days left before the user is allowed to change her password again. An empty field and a value of 0 mean that there is no minimum password age.
5. **Maximum**: The maximum number of days the password is valid, after that user is forced to change her password again.
6. **Warn**: The number of days before the password is to expire the user is warned that his/her password must be changed
7. **Inactive**: The number of days after the password expires that the account is disabled.
8. **Expire** The date of expiration of the account is expressed as the number of days since Jan 1, 1970.

# MODES IN JOHN THE RIPPER TOOL

This tool provides at least four modes:

- **Single crack**: This mode can be helpful in cases when a user has set a password for an account based on commonly available information or phrase in the username (e.g. admin: admin888).
- **Wordlist**: As you can see, the dictionary attack method uses this mode.
- **Incremental**: This mode is used by the brute-force method.
- **External**: It's an optional mode. In this mode, John the Ripper may use program code to generate words.

# HOW TO USE JOHN THE RIPPER TOOL

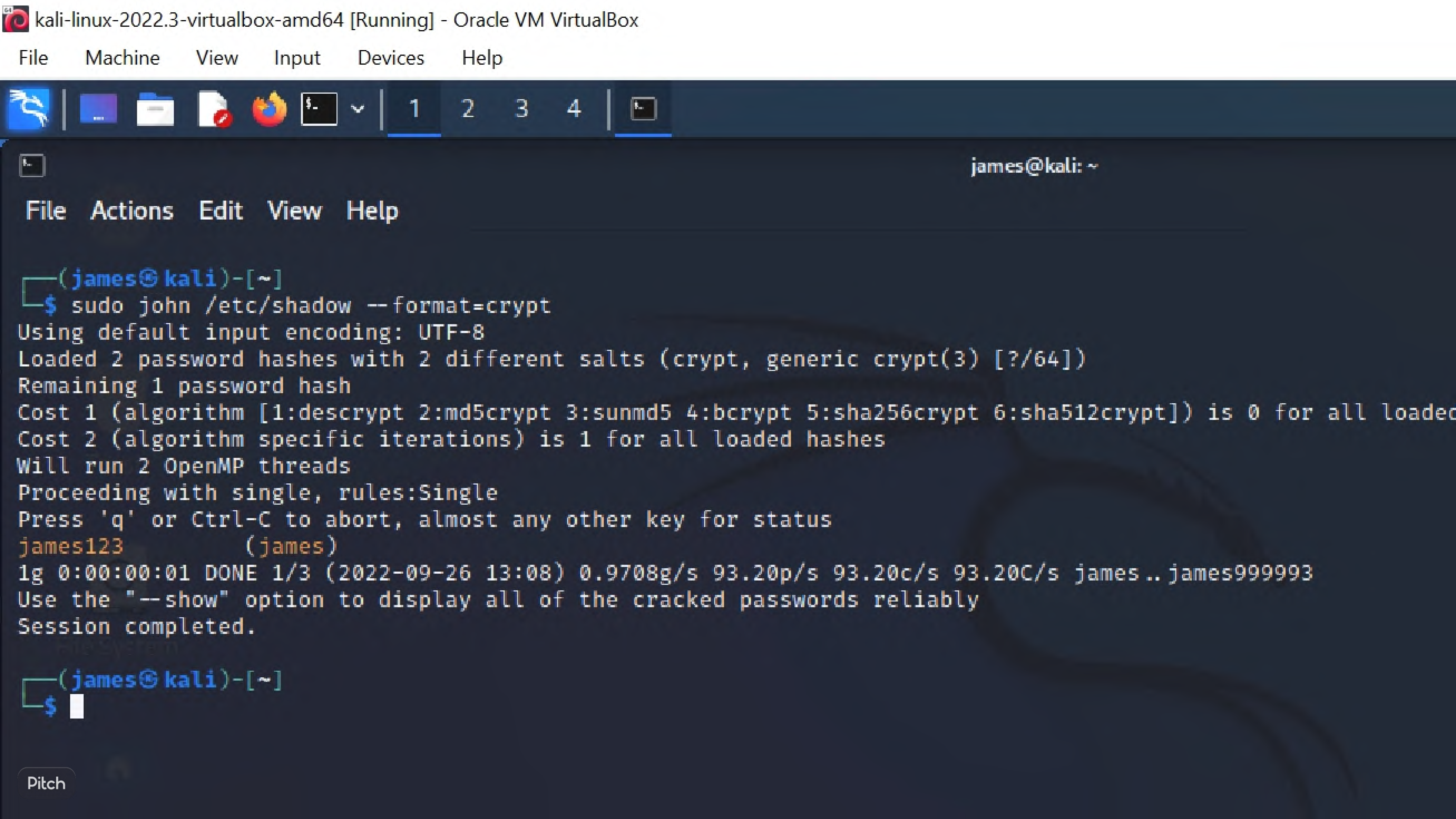
To use john the ripper tool we need to have an encrypted hash code of a password and depending upon the details we know about possible passwords we can use our preferred john the ripper tool mode. Let us take an example of cracking a Linux account password of a user named JAMES.the

# CRACKING A LINUX ACCOUNT PASSWORD

We have a Linux operating system logged in by user James with a password set as james123 by looking we can judge that it is quite a weak password. We will use Single crack mode which is by default if no method is specified in the command.

Open the terminal and enter the command

```
sudo john /etc/shadow —format=crypt
```



```
┌──(james㉿kali)-[~]
└─$ sudo john /etc/shadow --format=crypt
Using default input encoding: UTF-8
Loaded 2 password hashes with 2 different salts (crypt, generic crypt(3) [?/64])
Remaining 1 password hash
Cost 1 (algorithm [1:descrypt 2:md5crypt 3:sunmd5 4:bcrypt 5:sha256crypt 6:sha512crypt]) is 0 for all loade
Cost 2 (algorithm specific iterations) is 1 for all loaded hashes
Will run 2 OpenMP threads
Proceeding with single, rules:Single
Press 'q' or Ctrl-C to abort, almost any other key for status
james123         (james)
1g 0:00:00:01 DONE 1/3 (2022-09-26 13:08) 0.9708g/s 93.20p/s 93.20c/s 93.20C/s james..james999993
Use the "--show" option to display all of the cracked passwords reliably
Session completed.

┌──(james㉿kali)-[~]
└─$ 
```

# CONCLUSION

In a real-world scenario, it would be a good idea to set some options in the configuration files before running any of the examples on practical datasets. For example, these options can be used to specify the text file that JtR should use in wordlist mode, or to specify the range of password lengths (minimum and maximum) along with the character sets that the tool should use when running in incremental mode.

**In this tutorial we learned the following:**

1. **Installing John the Ripper on a Kali Linux PC.**
2. **Crack password-protected zip/rar file.**

Now we have a Demo video that will show the working of john the ripper tool in cracking a Linux account password and how to crack a zip file password.