■ Password Cracking Challenge using John the Ripper

1. Introduction

The project demonstrates the vulnerabilities of weak password protection by using John the Ripper to crack hashed passwords. The goal is to show how attackers can exploit insecure password practices and to highlight the need for stronger password policies and secure hashing mechanisms.

2. Abstract

This project focuses on cracking MD5 and SHA1 password hashes using John the Ripper, a popular open-source password cracking tool. By creating sample hashes, applying dictionary-based attacks, and analyzing the results, the project emphasizes the risks of using weak passwords and outdated hashing algorithms. The results provide insight into how quickly insecure credentials can be compromised, reinforcing the importance of implementing strong authentication policies in real-world systems.

3. Tools Used

- John the Ripper – password cracking tool.
- Hashcat – GPU-based alternative (optional for comparison).
- Wordlists – e.g., rockyou.txt, custom weak password lists.
- Operating System – Kali Linux / Ubuntu (Linux environment).
- Other Utilities – openssl, md5sum, sha1sum for generating test hashes.

4. Steps Involved in Building the Project

Step 1: Create a List of Hashed Passwords
Generate MD5 and SHA1 hashes for weak passwords:

```
echo -n "password123" | md5sum
echo -n "letmein" | sha1sum
```

Save these hashes in hashes.txt.

Step 2: Use John with Custom Wordlists or Rules

```
john --format=raw-md5 --wordlist=wordlist.txt md5_hashes.txt
john --format=raw-sha1 --wordlist=wordlist.txt sha1_hashes.txt
```

Apply rule-based attacks:

```
john --rules --wordlist=wordlist.txt hashes.txt
```

Step 3: Crack Hashes and Identify Weak Passwords
Example output:

```
password123 (user1)
letmein (user2)
qwerty (user3)
```

Step 4: Suggest Better Password Policies
• Minimum 12+ characters

• Use uppercase, lowercase, numbers, and symbols
• Avoid reusing old passwords
• Use secure hashing (bcrypt, scrypt, Argon2)
• Implement account lockout/rate-limiting
• Encourage password managers.

5. Screenshots/Outputs

Sample cracking output:

```
Loaded 5 password hashes with no different salts
password123 (user1)
letmein (user2)
qwerty (user3)
```

6. Conclusion

This project proved that weak passwords and outdated hashing methods like MD5 and SHA1 are highly insecure. Using John the Ripper, we successfully cracked several passwords within seconds. The exercise reinforces the need for stronger password creation policies, secure hashing algorithms, and modern authentication practices.

7. Challenges Faced (Optional)

• Configuring John the Ripper with custom wordlists.
• Understanding hash formats (MD5 vs SHA1).
• Ensuring wordlists included relevant weak passwords.

8. References

- John the Ripper official documentation: https://www.openwall.com/john/
- Hashcat documentation: https://hashcat.net/wiki/
- Wordlist resources (e.g., rockyou.txt).
- Tutorials on Linux openssl, md5sum, and sha1sum utilities.