

**Encryption and Password Cracking Report**

**Introduction**

Encryption and password cracking are essential techniques in cybersecurity. Encryption protects sensitive data by converting it into unreadable formats, ensuring confidentiality, while password cracking is used to test the strength of authentication mechanisms by recovering passwords through methods like brute-force attacks. In the cybersecurity industry, encryption helps secure data against unauthorized access, while password cracking is employed in penetration testing to identify weak passwords and vulnerabilities in systems. Both techniques play a crucial role in safeguarding data and assessing system security.

**Purpose**

This lab focuses on exploring encryption techniques, hash functions, password cracking, network traffic sniffing, and brute-force attacks using tools in Kali Linux. The lab will demonstrate the importance of encryption in securing data, how attackers exploit weak passwords, and how network sniffing can reveal vulnerabilities in unsecured protocols.

**Scope**

The scope of this lab covers symmetric and asymmetric encryption using OpenSSL and GPG, password cracking with John the Ripper, network traffic analysis with Wireshark, and a brute-force attack using Hydra. It includes practical tasks such as encrypting/decrypting files, generating password hashes, cracking those passwords, capturing network traffic, and brute-forcing SSH passwords to simulate real-world cybersecurity scenarios.

**Overview**

The evaluation identified several critical elements related to encryption, password strength, and network vulnerabilities. Both symmetric and asymmetric encryption methods were applied to protect data, while password cracking techniques demonstrated how weak password policies can expose systems to attack. Network sniffing uncovered vulnerabilities in unencrypted protocols like Telnet and FTP, showing the risks of transmitting data in cleartext. Additionally, the brute-force attack on an SSH service highlighted the importance of strong password policies and the use of secure protocols.

The lab is divided into several parts:

1. **Symmetric Encryption**: Using OpenSSL to encrypt and decrypt files with AES-256.
2. **Asymmetric Encryption**: Encrypting and decrypting files using GPG, leveraging public and private keys.
3. **Hash Functions**: Generating MD5 and SHA-256 hashes and comparing their outputs.
4. **Password Cracking**: Using John the Ripper to crack password hashes.
5. **Network Sniffing**: Using Wireshark to capture Telnet/FTP traffic and demonstrate the vulnerability of cleartext protocols.
6. **Brute Force Attack**: Using Hydra to brute-force SSH login credentials, highlighting the risks of weak passwords.

**Results**

**Part1:**

A screenshot of a computer screen

Description automatically generated

**Part 2:**

A computer screen shot of a computer code

Description automatically generated

A screenshot of a computer screen

Description automatically generated

**Part 3:**

A screen shot of a computer

Description automatically generated

**Part 3**

Different hash functions, like MD5 and SHA-256, produce distinct results due to differences in their algorithms, with MD5 generating a shorter, less secure hash (128-bit), while SHA-256 provides a longer, more secure hash (256-bit), making it more resistant to collisions and attacks. MD5, though faster, is considered vulnerable to collision attacks and is not recommended for cryptographic purposes. In contrast, SHA-256 is slower but offers much stronger security, making it suitable for applications like digital signatures and blockchain.

A screenshot of a computer screen

Description automatically generated

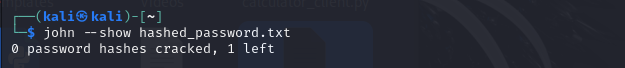
A screen shot of a computer

Description automatically generated

**Part 4**

A screenshot of a computer screen

Description automatically generated



**Part 5**

A screenshot of a computer

Description automatically generated

**Part 6**

A computer screen shot of white text

Description automatically generated

A computer screen shot of a program

Description automatically generated

A screenshot of a computer program

Description automatically generated

A computer screen shot of a computer code

Description automatically generated

**Questions/Answers:**

1. Discuss the differences between symmetric and asymmetric encryption in terms of speed and security.
2. Why are hashing algorithms used? How are they different from encryption algorithms?
3. What are the potential risks of password cracking tools like John the Ripper and Hydra if misused?
4. **Symmetric encryption** is significantly faster than **asymmetric encryption**, but it is generally considered less secure because it uses a single shared key for both encryption and decryption, making key distribution a critical challenge. On the other hand, asymmetric encryption, which uses a public/private key pair, prioritizes security by allowing anyone to encrypt data with the public key, while only the holder of the corresponding private key can decrypt it. However, this process is computationally slower due to the complex key management involved.
5. **Hashing algorithms** are used to verify the integrity of data, ensuring that a piece of information hasn't been altered, by generating a unique "fingerprint" of the data that cannot be reversed. In contrast, encryption algorithms are used to scramble data so that only authorized parties with the correct key can access its original form. Essentially, hashing is a one-way process focused on data integrity, while encryption is a two-way process focused on data confidentiality.
6. If misused, password cracking tools like **John the Ripper** and **Hydra** can pose significant risks, including unauthorized access to sensitive data, identity theft, financial loss, reputational damage to organizations, and the potential to breach sensitive systems. These tools allow malicious actors to gain access to user accounts with weak passwords through brute-force or dictionary attacks, particularly when targeting large datasets of compromised credentials.