Day 6

Cryptography: -

Cryptography is the practice and study of techniques used to secure communication and information from adversaries. It involves converting plain text into unreadable formats, known as ciphertext, using algorithms and keys. Cryptography plays a crucial role in ensuring data confidentiality, integrity, authentication, and non-repudiation in various applications such as secure communication, online transactions, and data protection.

Key Concepts in Cryptography:

1. **Encryption:**
    - **Process:** Converts plain text into ciphertext using an encryption algorithm and a key.
    - **Purpose:** Ensures data confidentiality by making it unreadable to unauthorized users.
    - **Algorithms:** Various encryption algorithms exist, such as AES, DES, and RSA, each with its own strengths and use cases.
2. **Decryption:**
    - **Process:** Reverses the encryption process by converting ciphertext back into plain text using the decryption algorithm and key.
    - **Key:** Decryption requires the correct key that corresponds to the one used for encryption.

Types of Encryption:

1. **Symmetric Encryption:**
    - Uses a single key for both encryption and decryption.
    - Examples include Advanced Encryption Standard (AES) and Data Encryption Standard (DES).
    - Fast and efficient for encrypting large amounts of data.
2. **Asymmetric Encryption:**
    - Utilizes a pair of keys (public and private) for encryption and decryption.
    - Examples include RSA (Rivest-Shamir-Adleman) and Elliptic Curve Cryptography (ECC).
    - Enables secure communication and digital signatures.

Encryption and Decryption Process:

1. **Encryption:**
    - Data is encrypted using an encryption algorithm and a key.
    - The plain text is transformed into ciphertext, which appears as random characters.
    - The encrypted data is transmitted or stored securely.

2. **Decryption:**
    - The recipient uses the corresponding decryption key to decrypt the ciphertext.
    - The ciphertext is transformed back into plain text, revealing the original data.
    - Decryption is essential for accessing and interpreting encrypted information.
3. **Cryptographic Hash Functions:**
    - Generate fixed-length output (hash) for input data.
    - Used for data integrity verification and password storage. Examples include SHA-256 and MD5.
4. **Digital Signatures:**
    - Provide authentication and non-repudiation.
    - Created using the private key and verified using the corresponding public key.

Common Cryptographic Algorithms:

1. **AES (Advanced Encryption Standard):**
    - Widely used symmetric encryption algorithm for securing data.
    - Supports key sizes of 128, 192, and 256 bits.
2. **RSA (Rivest-Shamir-Adleman):**
    - Popular asymmetric encryption algorithm for secure communication.
    - Key generation involves the use of public and private key pairs.
3. **SHA-256 (Secure Hash Algorithm 256-bit):**
    - Cryptographic hash function used for data integrity verification.
    - Generates a fixed-length hash value of 256 bits.

Hashcat Overview:

Hashcat is a powerful command-line utility designed for advanced password cracking and hash recovery. It is known for its speed and efficiency in decrypting hashed passwords using various attack modes. Here are some key points about Hashcat :

1. **Functionality:**
   - Hashcat offers multiple stand-alone binaries that are useful for cracking passwords.
   - It supports five unique modes of attack for over 300 highly-optimized hashing algorithms.
2. **Features:**
   - **Speed:** Hashcat is recognized as one of the fastest password recovery utilities available.
   - **Versatility:** It can crack hashes using different attack modes, such as dictionary attacks, brute force attacks, and rule-based attacks.
   - **Customization:** Users can customize their cracking strategies by specifying rules and parameters.
3. **Use Cases:**
   - **Password Cracking:** Hashcat is commonly used by security professionals and researchers to recover passwords from hashed data.
   - **Security Testing:** It is an essential tool for penetration testing and assessing the strength of password security measures.
   - **Research:** Hashcat is utilized in academic research and cybersecurity studies for analyzing password security vulnerabilities.
4. **Resources:**
   - **Hashcat-utils:** These are a set of small utilities that complement Hashcat and enhance its capabilities for advanced password cracking.
   - **GitHub Repository:** Hashcat has an active GitHub repository where users can access the latest updates, documentation, and community support.

Hashcat Command Syntax and Example:

Command Syntax:

The general syntax for running Hashcat commands is as follows:

hashcat <options> <hashfile> <mask|wordfiles|directories>.

Example:

Let's consider an example where we want to crack an MD5 hashed password using a dictionary attack with Hashcat.

1. **Command Syntax:**

plaintext

hashcat -m 0 -a 0 hashes.txt rockyou.txt

- -m 0: Specifies the hash type (in this case, **MD5**).

- -a 0: Specifies the attack mode (in this case, **dictionary attack**).
- hashes.txt: File containing the hashes to be cracked.
- rockyou.txt: Dictionary file containing a list of common passwords.

2. **Explanation:**
    - In this example, we are using Hashcat to crack MD5 hashes stored in the file hashes.txt.
    - We are employing a dictionary attack by providing the dictionary file rockyou.txt containing a list of common passwords.
    - Hashcat will attempt to match the hashes in hashes.txt with entries in rockyou.txt to find the corresponding passwords.

Tools for Identifying Hashes:

Here are some tools that can be used to identify different types of hashes used to encrypt data, especially passwords:

1. **hash-identifier | Kali Linux Tools:**
    - **Description:** Software to identify the different types of hashes used to encrypt data and passwords. Dependencies: python3.
2. **Hash Type Identifier - Identify unknown hashes:**
    - **Description:** Use this tool to identify, detect, and analyze hashes online. Identify hash types and detect unknown hashes using this tool.
3. **Hash Analyzer - TunnelsUp:**
    - **Description:** An online tool aimed at helping identify hash types by analyzing the characters that make up the hash.
4. **Name-That-Hash: A tool to identify hashes - Hackercool Magazine:**
    - **Description:** Name That Hash is a hash identifying tool that can identify over 300 types, including MD5 and SHA256.
5. **hashid | Kali Linux Tools:**
    - **Description:** hashID is a tool written in Python 3.x that supports the identification of over 175 unique hash types using regular expressions.

These tools can assist in identifying and analysing various types of hashes, making them valuable resources for security professionals and researchers.

There are some other websites too which are: -

https://icyberchef.com/

https://hashes.com/