***Name – Mangal gupta***

***Reg. No. - 20223141***

**Explore the other Hash Algorithms such as SHA-1, SHA-224, SHA-256, SHA-334 and MD5**

**1. MD5 (Message Digest Algorithm 5)**

* **Introduction**: MD5 is a widely known cryptographic hash function that was designed by Ronald Rivest in 1992. It is the fifth iteration of the Message Digest algorithms and was once the most commonly used hash function for ensuring the integrity of data.
* **Working**: MD5 takes an input message of any length and processes it in 512-bit chunks. Before processing, the message is padded so that its length becomes a multiple of 512. This padding includes adding a single '1' bit, followed by enough '0' bits, and the original length of the message is appended as the final 64-bit block. MD5 then uses a **Merkle-Damgård structure**, performing mathematical operations on the blocks to generate a 128-bit hash value.
* **Output Size**: The output of MD5 is a 128-bit (16-byte) hash value, typically represented as a 32-character hexadecimal number.
* **Common Uses**: Initially, MD5 was used for checksums, verifying file integrity, and storing hashed passwords. It was also used in digital signatures, certificates, and other security protocols.
* **Security Issues**: Despite its initial popularity, MD5 is no longer considered secure due to its vulnerability to **collision attacks**. A collision attack occurs when two different inputs produce the same hash value. In 2004, researchers showed that it was possible to create two different files with the same MD5 hash, which compromised its security. Thus, MD5 should not be used for cryptographic purposes today.
* **Example Hash**:
  + Input: hello
  + MD5 Hash: 5d41402abc4b2a76b9719d911017c592

**2. SHA-1 (Secure Hash Algorithm 1)**

* **Introduction**: SHA-1 was developed by the National Security Agency (NSA) in 1993 as part of the Secure Hash Algorithm (SHA) family. It became widely used in various security protocols such as SSL (Secure Socket Layer), TLS (Transport Layer Security), and in applications like Git for version control.
* **Working**: Similar to MD5, SHA-1 processes input data in 512-bit blocks. Before processing, the input message is padded to ensure its length is divisible by 512. The padded message is then processed using a series of logical functions and bitwise operations to generate a 160-bit hash value. SHA-1 also uses the **Merkle-Damgård construction**.
* **Output Size**: SHA-1 produces a 160-bit (20-byte) hash value, typically shown as a 40-character hexadecimal number.
* **Common Uses**: SHA-1 was widely used in SSL/TLS certificates, digital signatures, and version control systems like Git. It was also used in data integrity verification in certain applications.
* **Security Issues**: SHA-1 has been vulnerable to **collision attacks** since 2005, but it wasn’t until 2017 that Google and CWI Amsterdam publicly demonstrated a successful SHA-1 collision attack. This attack confirmed that SHA-1 should no longer be trusted for secure cryptographic operations. Many organizations have since moved to more secure algorithms like SHA-256.
* **Example Hash**:
  + Input: hello
  + SHA-1 Hash: aaf4c61ddcc5e8a2dabede0f3b482cd9aea9434d

**3. SHA-224 (Secure Hash Algorithm 224)**

* **Introduction**: SHA-224 is part of the **SHA-2 family**, which was introduced by the NSA in 2001. SHA-224 is essentially a truncated version of SHA-256, designed to offer a shorter hash output while maintaining strong security.
* **Working**: SHA-224 works by first padding the input message to a length that is a multiple of 512 bits. The message is then divided into blocks and processed using a set of logical and bitwise functions. Like other SHA algorithms, SHA-224 uses a **Merkle-Damgård construction**, which involves chaining the output of one block as the input to the next.
* **Output Size**: SHA-224 produces a 224-bit (28-byte) hash value, making it smaller than the 256-bit output of SHA-256 but still more secure than older algorithms like MD5 and SHA-1.
* **Common Uses**: SHA-224 is used in environments where a smaller hash size is sufficient but security is still important. It is less commonly used than SHA-256 but can be useful in systems with limited storage.
* **Strengths**: SHA-224 is resistant to **collision** and **pre-image attacks**, providing good security for most applications. It offers a balance between shorter hash size and security.
* **Example Hash**:
  + Input: hello
  + SHA-224 Hash: ea09ae9cc6768c50fcee903ed054556e5bfc8347907f12598aa24193

**4. SHA-256 (Secure Hash Algorithm 256)**

* **Introduction**: SHA-256 is part of the SHA-2 family and is one of the most widely used and trusted cryptographic hash functions today. It was introduced by the NSA in 2001 and is used in a variety of security applications, including digital certificates and blockchain technologies.
* **Working**: SHA-256 processes the input message in 512-bit blocks. Like other SHA algorithms, it first pads the input to ensure its length is divisible by 512. It then processes each block through a series of mathematical operations, including **bitwise shifts**, **XORs**, and **additions**, to generate a unique 256-bit hash value. SHA-256 uses a **Merkle-Damgård construction**.
* **Output Size**: SHA-256 produces a 256-bit (32-byte) hash value. This makes it secure against most known attacks, including collision and pre-image attacks.
* **Common Uses**: SHA-256 is widely used in digital certificates, SSL/TLS protocols, and **blockchain technologies** (most notably in cryptocurrencies like **Bitcoin**). It is also used in password hashing and file integrity verification.
* **Strengths**: SHA-256 is considered highly secure and is resistant to all known cryptographic attacks. It is the default choice for many applications requiring strong security.
* **Example Hash**:
  + Input: hello
  + SHA-256 Hash: 2cf24dba5fb0a30e26e83b2ac5b9e29e1b161e5c1fa7425e73043362938b9824

**5. SHA-384 (Secure Hash Algorithm 384)**

* **Introduction**: SHA-384 is another member of the SHA-2 family, designed to provide a higher level of security than SHA-256 by producing a longer hash output. It is often used in high-security environments where a larger hash size is necessary.
* **Working**: SHA-384 is a truncated version of **SHA-512**. It processes data in 1024-bit blocks, which is larger than the 512-bit blocks used by SHA-256. Like other hash functions, it pads the input message to ensure its length is a multiple of the block size. SHA-384 applies a series of logical and bitwise operations to each block and generates a 384-bit hash value.
* **Output Size**: SHA-384 produces a 384-bit (48-byte) hash value, offering a higher level of security than SHA-256 but with a larger output size.
* **Common Uses**: SHA-384 is used in applications that require a high level of security, such as government systems, military communications, and large-scale secure data storage. It is also used in digital certificates and SSL/TLS protocols for highly sensitive data.
* **Strengths**: SHA-384 provides a stronger level of security than SHA-256, making it suitable for environments where even the smallest possibility of a collision or attack must be avoided.
* **Example Hash**:
  + Input: hello
  + SHA-384 Hash: 59e1748777448c69de6b800d7a33bbfb9e78128c3984310763ed804cfb7cbf15