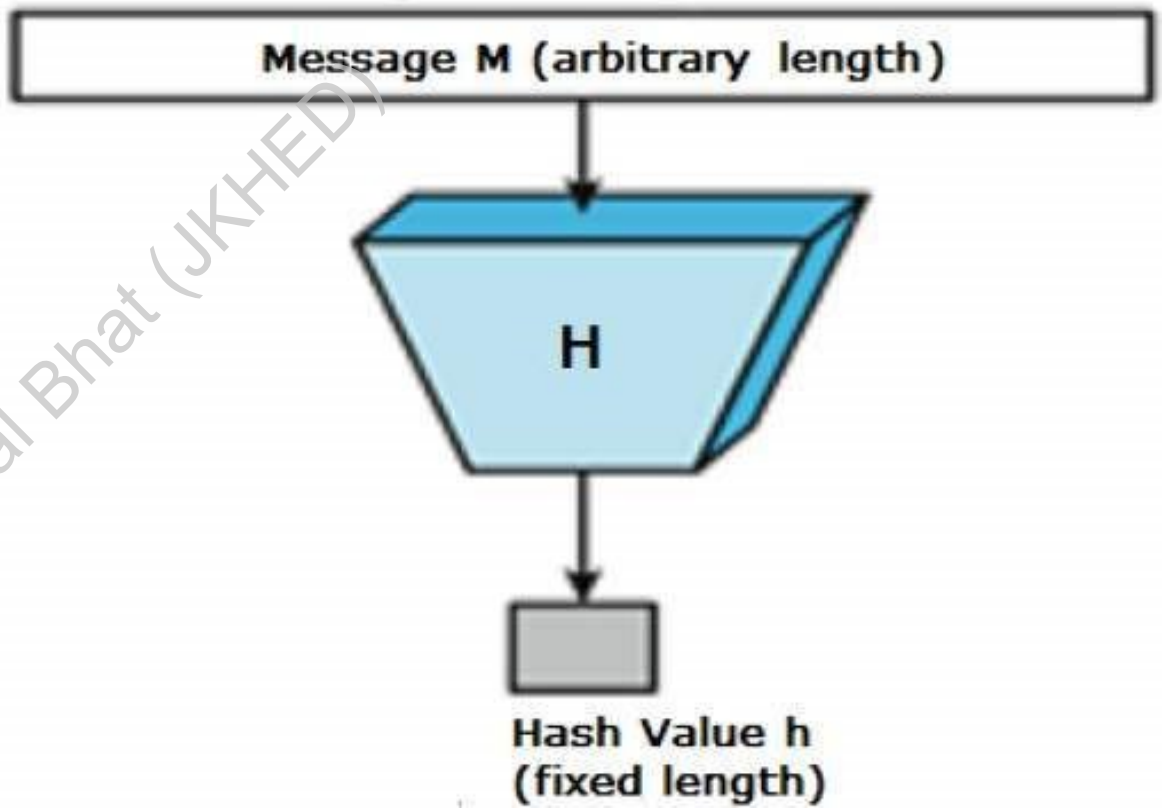
Hash Functions

By

Prof. Muhammad Iqbal Bhat

Government Degree College
Beerwah



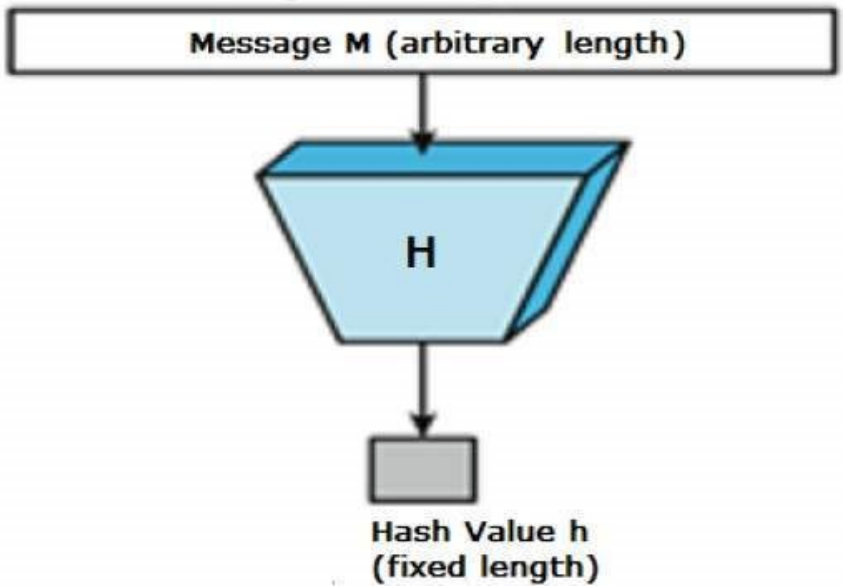
Topics

- Hash functions
- Features of Hash Functions:
- Properties of Hash functions

Hash Functions:

A hash function is a mathematical function that converts a numerical input value into another compressed numerical value. The input to the hash function is of arbitrary length but output is always of fixed length.

Values returned by a hash function are called **message digest** or simply **hash values**. The following picture illustrated hash function



Features of Hash Functions

- Hash functions take input data (often a message or file) and return a fixed-size string of characters, called a hash value or digest.
- Hash functions are deterministic, meaning they always produce the same output for a given input.
- Hash functions are designed to be one-way functions, meaning that it is computationally infeasible to reverse-engineer the input data from the hash value.
- Hash functions are designed to be collision-resistant, meaning that it is computationally infeasible to find two different inputs that produce the same hash value.
- Hash functions are designed to be preimage-resistant, meaning that given a hash value, it is computationally infeasible to find an input that produces that hash value.
- Hash functions are designed to be second preimage-resistant, meaning that given an input and its hash value, it is computationally infeasible to find a different input that produces the same hash value.
- Hash functions are often used in information security for purposes such as password storage, digital signatures, data integrity verification, and blockchain technology.

Commonly used Hash Functions:

Hash Function	Digest Size (bits)	Block Size (bits)	Collision Resistance	Preimage Resistance	Second Preimage Resistance
MD5	128	512	Vulnerable	Vulnerable	Vulnerable
SHA-1	160	512	Weak	Vulnerable	Vulnerable
SHA-256	256	512	Strong	Strong	Strong
SHA-512	512	1024	Strong	Strong	Strong
SHA-3-256	256	1088	Strong	Strong	Strong
SHA-3-512	512	576	Strong	Strong	Strong

Applications of Hash Functions

Password storage: Hash functions are often used to securely store passwords in a database. When a user creates a password, it is hashed and the hash value is stored in the database. When the user enters their password to log in, it is hashed again and the resulting hash value is compared to the stored hash value. This allows for password verification without storing the actual passwords, which could be stolen in a data breach.

Digital signatures: Hash functions are used in digital signature schemes to verify the integrity of a message. A message is hashed, and the hash value is signed with the signer's private key. The recipient can then verify the signature by hashing the message themselves and comparing it to the hash value that was signed. If the hash values match, the recipient can be sure that the message has not been tampered with and was signed by the person who claims to have signed it.

Data integrity verification: Hash functions can be used to ensure that data has not been tampered with during transmission or storage. A hash value is computed for the original data and sent or stored alongside the data. When the data is received or retrieved, the hash value is computed again and compared to the original hash value. If the hash values match, the data has not been tampered with. If they do not match, the data has been altered.

Blockchain technology: Hash functions are used extensively in blockchain technology, which is the underlying technology behind cryptocurrencies such as Bitcoin and Ethereum. Hash functions are used to create the blocks that make up the blockchain, and the hash values of each block are used to create the next block in the chain. This ensures the integrity and immutability of the blockchain.

File and data identification: Hash functions can be used to identify files or data by computing a unique hash value for each file or data set. This can be useful for deduplication or for verifying that the correct data or file has been transferred.

Message authentication codes (MACs): MACs are used to authenticate the integrity of a message and ensure that it has not been tampered with during transmission. Hash functions can be used to compute MACs by combining a secret key with the message and hashing the result. The recipient can then compute the MAC using the same secret key and verify that it matches the MAC that was sent.