INSTITUTE OF INFORMATION TECHNOLOGY

Jahangirnagar University

জাহাঙ্গীরনগর বিশ্ববিদ্যালয়

IT-4259: Computer Network Security

for

4th Year 2nd Semester of B.Sc (Honors) in IT (5th Batch)

Lecture: 08

Cryptographic Hash Function

Prepared by:

K M Akkas Ali

akkas_khan@yahoo.com, akkas@juniv.edu

Associate Professor

Institute of Information Technology (IIT)

Jahangirnagar University, Dhaka-1342

Objectives of this Lecture:

- To introduce general ideas behind hashing and hash functions.
- Cryptographic hash function and its importance.
- Use and application of hash function.
- Desirable properties of a hash function.
- Simple hash functions.

Prepared by: K M Akkas Ali, Associate Professor, IIT, JU

What is Hashing?

- Hashing is the transformation of a string of characters into a usually shorter fixed-length value or key that represents the original string.
- Hashing is a cryptographic technique that produces hash values using an algorithm or hash function for accessing data or for security purposes.
- A hash value (or simply hash), also called a message digest, is a number generated from a string of text. The hash is substantially smaller than the text itself.
- In hashing, a fixed-length message digest is created out of a variable-length message. The digest is normally much smaller than the message.
- It plays a vital role in security system that creates a unique, fixed-length signature for a message or data set.
- People commonly use them to compare sets of data. Since a hash is unique to a specific message, even minor changes to that message result in a dramatically different hash. Therefore it is very resistant to tampering.

Prepared by: K M Akkas Ali, Associate Professor, IIT, JU

What is Hashing (cont…)

- Hashing also refers to a search technique or a method of accessing data records, where search time is independent of the number of the elements in the collection.

- Hashing is used to index and retrieve items in a database because it is faster to find the item using the shorter hashed key than to find it using the original value.

- For example, consider a list of names in a database:
    - Rassel Akram
    - Asif Afzal Khan
    - Khan Ataus Samad
    - Anuradha Mondol

- Each of these names would be the key in the database for that person's data. A database search mechanism would first have to start looking character-by-character across the name for matches until it found the match.

- But if each of the names were hashed, it might be possible to generate a unique four-digit key or index for each name. So you might get something like:
    - 1345 Rassel Akram
    - 3097 Asif Afzal Khan
    - 4060 Khan Ataus Samad
    - 7350 Anuradha Mondol

- A search for any name would first consist of computing the hash value (using the same hash function used to store the item) and then comparing for a match using that value. This is much more efficient than searching through all the records till the matching record is found. Because, to find a match across four digits, each having only 10 possibilities is faster, than across an unpredictable value length where each character had 26 possibilities.

Prepared by: K M Akkas Ali, Associate Professor, IIT, JU

Importance of Hashing

- Hashing is used to index and retrieve items in a database because it is faster to find the item using the shorter hashed key than to find it using the original value.
- In addition to faster data retrieval, hashing is also used to encrypt and decrypt digital signatures (used to authenticate message senders and receivers).
- Hashing plays vital a role in security systems where it is used to ensure that transmitted messages have not been tampered with.
- The sender generates a hash of the message, encrypts it, and sends it with the message itself. The recipient then decrypts both the message and the hash, produces another hash from the received message, and compares the two hashes. If they are the same, there is a very high probability that the message was transmitted intact.

Prepared by: K M Akkas Ali, Associate Professor, IIT, JU

Hash Function

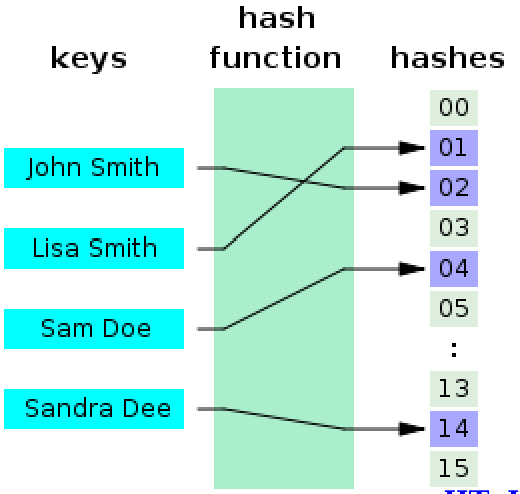
- A hash function is a formula or an algorithm that-
  - takes large data sets of variable length as input, and
  - returns smaller data sets of fixed length as output.

- Since, the output is smaller than the input data, a hash function compresses an ***n***-bit message string to create an ***m***-bit string where ***n*** is normally greater than ***m***.

- The values returned by a hash function are called hash values, hash codes, hash sums, checksums or simply hashes.

- Hash function creates hash value in such a way that it is extremely unlikely that some other text will produce the same hash value.

- A hash table (also called hash map) is used to implement an associative array that can map keys to values. A hash table uses a hash function to compute an index into an array of buckets or slots, from which the correct value can be found.



Prepared by: K M Akkas Ali, Associate Professor, IIT, JU

Cryptographic Hash Function

- A cryptographic hash function is a hash function that takes an arbitrary block of data as input and returns a fixed-size bit string as output. The returned value is called the cryptographic hash value.

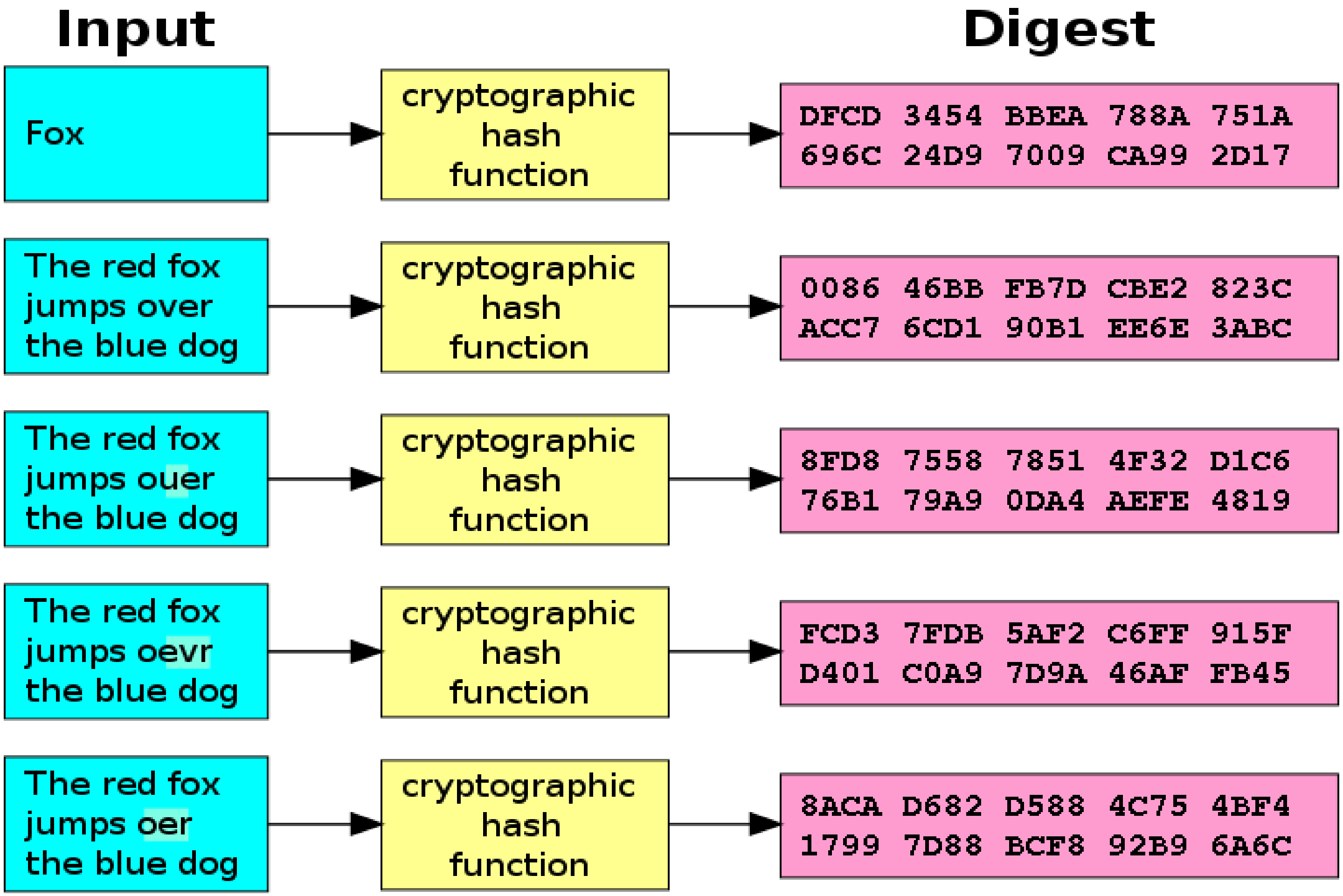
- Cryptographic hash function creates hash value in such a way that any (accidental or intentional) change to the data will change the hash value. Therefore, it is extremely unlikely that some other text will produce the same hash value.

- The data to be encoded are often called the message, and the hash value is sometimes called the message digest or simply digest.

Prepared by: K M Akkas Ali, Associate Professor, IIT, JU

Cryptographic Hash Function

- In cryptographic hash function, even a small changes in the input would cause a large change in the output.
- Figure below shows how the slight changes input (here in the word "over") drastically change the resulting output.



Input

Digest

Fox → cryptographic hash function → DFCD 3454 BBEA 788A 751A 696C 24D9 7009 CA99 2D17

The red fox jumps over the blue dog → cryptographic hash function → 0086 46BB FB7D CBE2 823C ACC7 6CD1 90B1 EE6E 3ABC

The red fox jumps ouer the blue dog → cryptographic hash function → 8FD8 7558 7851 4F32 D1C6 76B1 79A9 0DA4 AEFE 4819

The red fox jumps oevr the blue dog → cryptographic hash function → FCD3 7FDB 5AF2 C6FF 915F D401 C0A9 7D9A 46AF FB45

The red fox jumps oer the blue dog → cryptographic hash function → 8ACA D682 D588 4C75 4BF4 1799 7D88 BCF8 92B9 6A6C

Prepared by: K M Akkas Ali, Associate Professor, IIT, JU

Illustration: Cryptographic Hash Function

- An illustration of the potential use of a cryptographic hash is as follows:
  - Alice poses a tough math problem to Bob and claims she has solved it.
  - Bob would like to try it himself, but would yet like to be sure that Alice is not bluffing. Therefore, Alice writes down her solution, computes its hash and tells Bob the hash value (whilst keeping the solution secret). Then, when Bob comes up with the solution himself a few days later, Alice can prove that she had the solution earlier by revealing it and having Bob hash it and check that it matches the hash value given to him before.

Prepared by: K M Akkas Ali, Associate Professor, IIT, JU

Use of Hash Function

- Cryptographic hash functions have many information security applications, such as in-
  - digital signatures
  - message authentication codes (MACs)
  - other forms of authentication
- Hash functions are primarily used to generate fixed-length output data that acts as a shortened reference to the original data. This is useful when the output data is too cumbersome to use in its entirety.
  - For example, consider a list of person’s names. Here, name of each person is of variable length. Searching for a person's name in the list is slow; time required to retrieve each name may also vary. But if each name could be hashed to a fixed length integer, then searching and retrieving each name will be performed in faster with constant time.
- Hash functions are also used to accelerate table lookup or data comparison tasks such as finding items in a database, detecting duplicated or similar records in a large file, finding similar stretches in DNA sequences, and so on.

Prepared by: K M Akkas Ali, Associate Professor, IIT, JU

Hash Functions Used in Cryptography

- The two commonly used hash functions are MD5 and SHA-1.
  - **MD5:**
    - MD stands for Message Digest.
    - Several MD hash algorithms designed by Ron Rivest are MD2, MD4 and MD5.
    - The last version MD5 is more secured than the previous versions.
    - It divides the message into blocks of 512 bits and creates a 128-bit digest.
  - **SHA-1:**
    - SHA stands for Secure Hash Algorithm.
    - This standard was developed by NIST (National Institute of Standards and Technology).
    - This standard is mostly based on MD5.
    - Several versions of SHA standard were realsed: SHA-1, SHA-224, SHA-256, SHA-384 and SHA-512.
    - SHA-1 returns a string of 160 bits.
    - Both MD5 and SHA-1 hash functions are built with the Merkle-Damgard construction.

Prepared by: K M Akkas Ali, Associate Professor, IIT, JU

Hash Functions Used in Cryptography

Merkle-Damgard Scheme:

- The Merkle-Damgard construction method takes an arbitrary sized input and breaks the input into fixed size blocks of the same size as the output. It applies a one way compression function to each of the blocks in turn, combining a block of input with the output of the previous block. The last block has bits representing the length of the entire message.
  - A one way compression function takes two fixed size inputs - the key and the plain text - and returns one single output - the cipher text which is the same size as the plain text.
  - An example of such a function is the Davis-Meyer compression function. It feeds the previous hash value (H_{i-1}) as the plaintext to be encrypted. It uses the each block of the message (m_i) as the key. The output ciphertext is then XORed with the previous hash value (H_{i-1}) to produce the next hash value (H_i). In the first round when there is no previous hash value it uses a predefined inital value (H_0).

Prepared by: K M Akkas Ali, Associate Professor, IIT, JU

Application of Hash Function in Cryptography

Hash functions are used for:

- Verifying the integrity of message and file
- Verifying password for secure login
- fingerprints of keys
- authentication
- digital signatures

Verifying the integrity of files or messages:

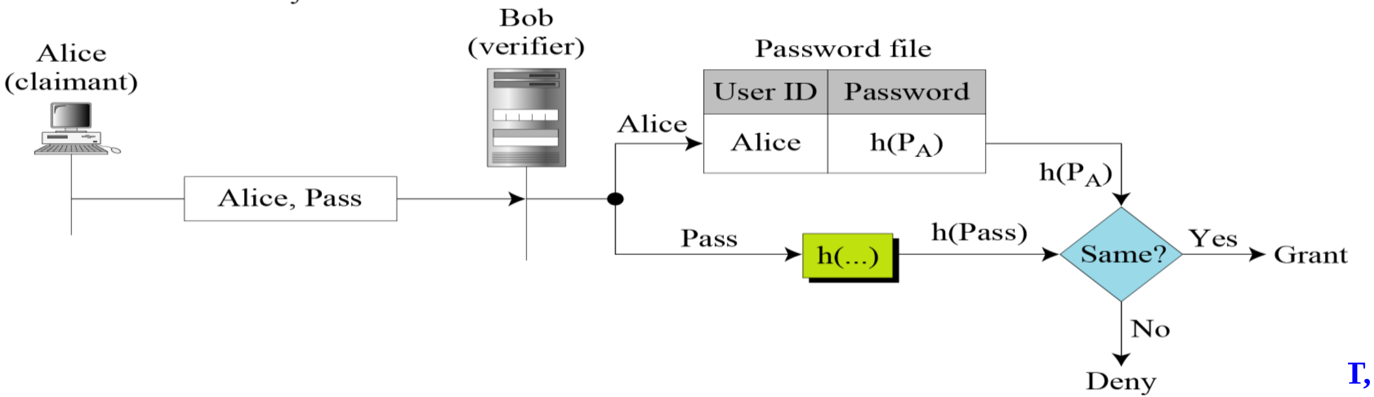
- An important application of secure hashes is verification of message integrity. Determining whether any changes have been made to a message (or a file), for example, can be accomplished by comparing message digests calculated before, and after, transmission (or any other event).
- For this reason, most digital signature algorithms only confirm the authenticity of a hashed digest of the message to be "signed". Verifying the authenticity of a hashed digest of the message is considered proof that the message itself is authentic.

Application of Hash Function in Cryptography

Verifying password for secure login:

- A related application of hash function is password verification.
- Storing all user passwords as plaintext character can result in a massive security breach if the password file is compromised.
- One way to reduce this danger is to only store the hash digest of each password instead of the plaintext password in the table (a file) that is stored by user identification.
- Any user can read the contents of the file, but, because the hash function is a one-way function, it is almost impossible to guess the value of the password.
- When the password is created , the system hashes it and stores the hash in the password file.
- When the user sends her user ID and password, the system creates a hash of the password and then compare the hash value with the one stored in the file.
- If there is a match, the user is granted access; otherwise, access is denied.

P_A: Alice's stored password
Pass: Password sent by claimant



Password file

User ID	Password
Alice	$h(P_A)$

Application of Hash Function in Cryptography

- **File or data identifier:**
  - A message digest can also serve as a means of reliably identifying a file;
  - One of the main applications of a hash function is to allow the fast look-up of a data in a hash table. Being hash functions of a particular kind, cryptographic hash functions lend themselves well to this application too.

- **Authentication:**
  - Authentication is the assurance that the communicating entity is the one that it claims to be.
  - Cryptographic hash function can be used for provide authentication.

Prepared by: K M Akkas Ali, Associate Professor, IIT, JU

Application of Hash Function in Cryptography

Digital Signature:

- Digital signature, first proposed in 1976 by Whitfield Diffie of Stanford University, is a digital code (encrypted message digest) that can be attached to an electronically transmitted message that uniquely identifies the sender.
- Like a written signature, the purpose of a digital signature is to guarantee that the individual sending the message really is who he or she claims to be. It is linked to the data in such a manner that if the data is changed, the digital signature is invalidated.
- When making a digital signature, cryptographic hash functions are generally used to construct the message digest.
- A digital signature servers three important purposes:
  - Verifies data integrity.
  - Provides authentication of the sender.
  - Provides non-repudiation

Prepared by: K M Akkas Ali, Associate Professor, IIT, JU

Properties of Cryptographic Hash Function

- ➢ A cryptographic hash function must be able to withstand all known types of cryptanalytic attack.
- ➢ A desirable cryptographic hash function should have the following properties:
- ❖ A hash function produces a fixed length value from a variable length source.
- ❖ It is easy to compute the hash value for any given message.
- ❖ **Pre-image resistance:** Given a hash h, it should be difficult to find any message m such that h = hash(m). That is, it is infeasible to generate a message that has a given hash.
- ❑ A function with this property is called a one-way function.
- ❑ Functions that lack this property are vulnerable to preimage attacks.
- ❖ **Second pre-image resistance:** Given a message m_1, it should be difficult to find another message m_2 such that m1 ≠ m2 and hash(m1) = hash(m2).
- ❑ Functions that lack this property are vulnerable to second-preimage attacks.
- ❑ It is infeasible to modify a message without changing the hash.

Prepared by: K M Akkas Ali, Associate Professor, IIT, JU

Properties of Cryptographic Hash Function

- **Collision resistance:** It should be difficult to find two different messages m_1 and m_2 with the same hash h. Such a pair is called a cryptographic hash collision.
    - This property is sometimes referred to as strong collision resistance. It requires a hash value at least twice as long as that required for preimage-resistance; otherwise collisions may be found by a birthday attack.
- A hash function must be referentially transparent, i.e., if called twice on input that is "equal" (for example, strings that consist of the same sequence of characters), it should give the same result.
- A hash procedure must be deterministic—meaning that for a given input value, it must always generate the same hash value.
- Hash functions are destructive, as the original data is lost when hashed.
- A small change in the input m would cause a large change in the output of the hash function.
- It should be impossible for an adversary to find two messages with substantially similar digests; or to infer any useful information about the data, given only its digest. Therefore, a cryptographic hash function should behave as much as possible like a random function while still being deterministic and efficiently computable.

- The above properties of a cryptographic hash function imply that a malicious adversary cannot replace or modify the input data without changing its digest. Thus, if two strings have the same digest, one can be very confident that they are identical.

Prepared by: K M Akkas Ali, Associate Professor, IIT, JU

Simple Hash Function

Some Popular Hash Function:

Here are some relatively simple hash functions that have been used:

- Division-remainder method
- Mid-square method
- Folding method

Division-remainder method:

- Using this method, choose a number ***m*** that is larger than the number ***n*** of keys in ***K*** (***K*** is a set of keys). Generally, the number ***m*** is chosen to be a prime number. The hash function ***H*** is defined as:

  H(k)= k (mod m) or,   H(k)= k (mod m)+1

- Here k (mod m) denotes the remainder when k is divided by m.
- The second formula is used when we want the hash addresses to range from 1 to m rather than from 0 to m-1.

Prepared by: K M Akkas Ali, Associate Professor, IIT, JU

Simple Hash Function

Example: Division-remainder method:

Suppose a company with 68 employees assigned a 4-digit employee number to each employee which is used as the primary key. Apply the division method of hash function to each of the following employee number:
3205, 7148, 2345

Solution:

- Since, there are 68 employees in the company, two digit employee number is sufficient to represent them.

- Highest 2 digit number is 99 and 97 is the nearest 2 digit prime number of 99. So, we divide each of the 4 digit employee number by 97.

  H(3205)= 3205 (mod 97)= 04.

  H(7148)= 7148 (mod 97)= 67

  H(2345)= 2345 (mod 97)= 17

- In the case that the memory addresses begin with 01 rather than 00, we choose that the function H(k) = k(mod m)+1 to obtain. H(3205)= 3205 (mod 97)+1= 4+1=05

Prepared by: K M Akkas Ali, Associate Professor, IIT, JU

Simple Hash Function

Mid-square method:

- Using this method, the key k is squared. Then the hash function H is defined by:

  H(k)=l,

  Where l is obtained by deleting digits from both ends of k^2.

Example: Mid-square method

Suppose a company with 68 employees assigned a 4-digit employee number to each employee which is used as the primary key. Apply the mid-square method of hash function to each of the following employee number:
3205, 7148, 2345

Solution:

K	3205	7148	2345
K^2	102**72**025	510**93**904	54**99**025
H(k) = I	72	93	99

- Observe that the 4th and 5th digits counting from right are chosen for the hash address.

Prepared by: K M Akkas Ali, Associate Professor, IIT, JU

Simple Hash Function

Folding method:

- Using this method, the key k is portioned into a number of parts, k_1, k_2, k_3, …..k_r, where each part, except possibly the last, has the same number of digits as the required address. Then the parts are added together, ignoring the last carry if any. That is:
- $H(k)=k_1+k_2+k_3+\ldots..+k_r$, where the leading-digit carries, if any, are ignored.

Example: Folding method

Suppose a company with 68 employees assigned a 4-digit employee number to each employee which is used as the primary key. Apply the folding method of hash function to each of the following employee number:
3205, 7148, 2345

Solution:
Chop the key into two parts and then add them.

K	3205	7148	2345
$H(k)=K_1+k_2$	32+05=37	71+48=119	23+45=77
H(k)	37	19	77

- Observed that the leading digit from the 2nd function is ignored.
- Alternatively, reverse folding method may be used as:

H(3205)= 32+50=82

Prepared by: K M Akkas Ali, Associate Professor, IIT, JU