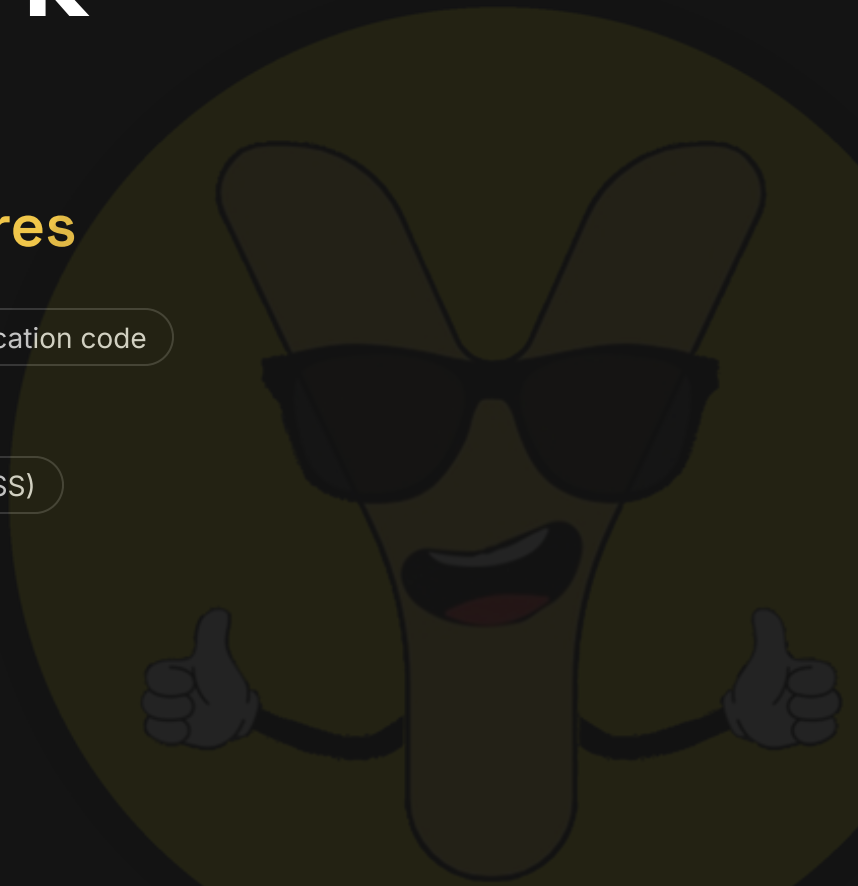
AKTU · 4TH YEAR · SEMESTER 7 · BCS072

Cryptography and Network Security

Unit 3 — Message Authentication and Digital Signatures

- Message Authentication Codes: Authentication requirements
- Authentication functions
- Message authentication code
- Hash functions
- Birthday attacks
- Security of hash functions
- Secure hash algorithm (SHA)
- Digital Signatures: Digital Signatures
- Elgamal Digital Signature Techniques
- Digital signature standards (DSS)
- Proof of digital signature algorithm



DECODED BY

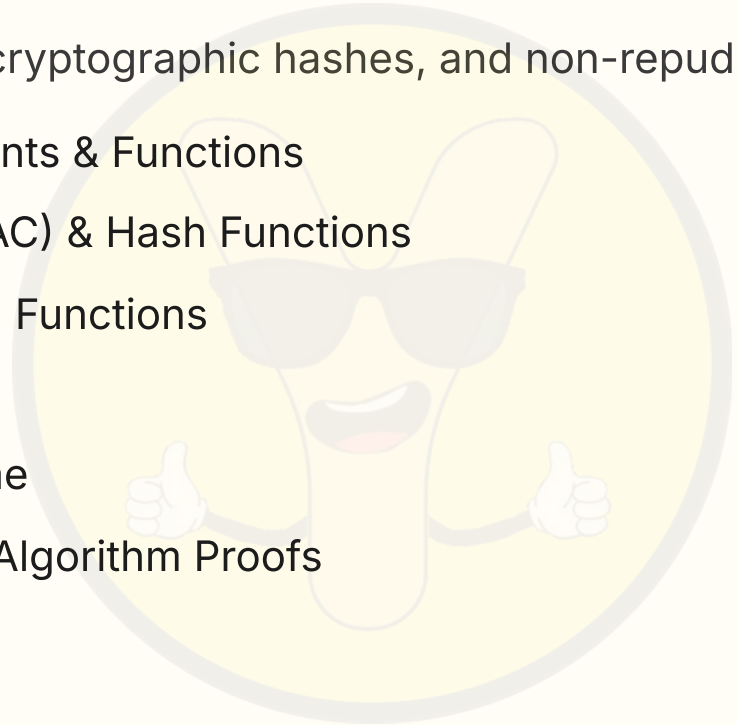
Chirag Jain

Founder, Ycotes · Gold Medalist in Artificial Intelligence, IIT Madras · taught 10,000+ students

Unit 3 Overview: Message Authentication and Digital Signatures

This unit covers integrity verification, cryptographic hashes, and non-repudiation mechanisms.

- Message Authentication Requirements & Functions
- Message Authentication Codes (MAC) & Hash Functions
- Birthday Attacks & Security of Hash Functions
- Secure Hash Algorithm (SHA)
- Digital Signatures & ElGamal Scheme
- Digital Signature Standard (DSS) & Algorithm Proofs



Authentication Requirements

Network security attacks necessitate robust mechanisms to protect message content and origin.

- Disclosure
  - Release of message contents to unauthorized entities.
  - Prevented by encryption.
- Traffic Analysis
  - Discovery of pattern, frequency, or length of communication.
  - Prevented by traffic padding.
- Masquerade
  - Insertion of messages from an unauthorized entity pretending to be an authorized one.
- Content Modification
  - Changes, deletions, additions, or reordering of message contents.
- Sequence Modification
  - Any modification to sequence of messages in a connection (insertion, deletion, reordering).
- Source Repudiation
  - Denial of transmission of a message by its actual sender.

Authentication Functions

MESSAGE AUTHENTICATION FUNCTION

A function that produces an authenticator, a value used to assure that a message has not been altered and comes from the alleged sender.

- Hash function: Maps an arbitrary-length message to a fixed-length hash value without using a secret key.
- Message Authentication Code (MAC): Uses a secret key to generate a fixed-size block dependent on both the message and the key.
- Encryption: Conventional encryption provides authentication if ciphertext structure makes unauthorized modification easy to detect.

Message Authentication Code (MAC)

MAC (TAGGED MESSAGE)

An authentication technique involving a secret key to generate a small block of data, known as a cryptographic checksum or MAC, appended to the message.

- Formula: $MAC = C(K, M)$ where K is the secret key, M is the message, and C is the MAC function.
- Receiver calculates $C(K, M)$ using the shared key and compares it with the received MAC.
- Provides both integrity and authenticity because only authorized parties share the key K.
- Does not provide non-repudiation by itself, as both sender and receiver share the same secret key.

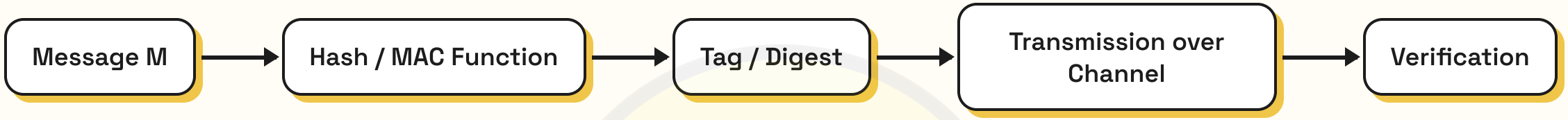
Hash Functions

CRYPTOGRAPHIC HASH FUNCTION

An algorithm that maps data of arbitrary size to a fixed-size bit string, called a hash value or digest.

- One-way property (Pre-image resistance): Given h, it is computationally infeasible to find x such that $H(x) = h$.
- Weak collision resistance (Second pre-image resistance): Given x, it is hard to find $y \neq x$ with $H(y) = H(x)$.
- Strong collision resistance (Collision resistance): It is hard to find any two random messages x and y such that $H(x) = H(y)$.

Message Authentication Architecture



General flow of generating and verifying message integrity using hash or MAC.

Birthday Attacks on Hash Functions

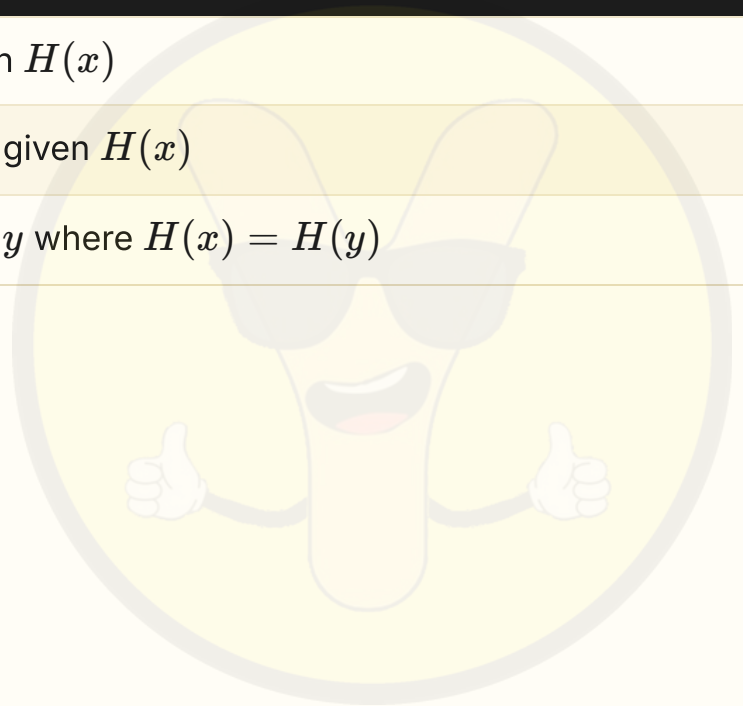
BIRTHDAY PARADOX

A probability theory result stating that in a set of N randomly chosen people, the probability that at least two share the same birthday exceeds 50% for $N = 23$.

- Applied to hash functions, it reduces the security bit-length against collision attacks.
- For a hash function with n-bit output, finding a collision requires about $2^{n/2}$ trials due to the birthday paradox.
- Therefore, a 128-bit hash offers only 64-bit security against collision attacks.
- Mandates the use of larger hash outputs like SHA-256 or SHA-512 in modern cryptographic systems.

Security of Hash Functions: Desired Properties

Property	Description	Required Difficulty
One-way	Find x given $H(x)$	2^n operations
Weak Collision	Find $y \neq x$ given $H(x)$	2^n operations
Strong Collision	Find any x, y where $H(x) = H(y)$	$2^{n/2}$ operations



Secure Hash Algorithm (SHA)

SHA is a family of cryptographic hash functions published by NIST as a U.S. Federal Information Processing Standard.

- SHA-0 and SHA-1
  - SHA-1 produces a 160-bit hash value.
  - SHA-1 found vulnerable to theoretical collisions; phased out.
- SHA-2 Family
  - Includes SHA-224, SHA-256, SHA-384, and SHA-512.
  - Uses 32-bit or 64-bit words and more rounds of operation.
- SHA-3 Family
  - Keccak algorithm selected via NIST competition.
  - Uses sponge construction differing from MD4-style structures.

SHA-512 Processing Steps

Detailed algorithmic flow for computing the SHA-512 hash digest.

- Step 1: Append padding bits so message length congruent to 894 mod 1024.
- Step 2: Append length field as a 128-bit integer representing original length.
- Step 3: Initialize 64-bit hash buffer values (eight variables A through H).
- Step 4: Process message in 1024-bit blocks through 80 rounds of compression.
- Step 5: Output final concatenated 512-bit digest from buffers A through H.

Digital Signatures

DIGITAL SIGNATURE

A mathematical scheme for verifying the authenticity of digital messages or documents, providing integrity, authentication, and non-repudiation.

- Created using the sender's private key; verified using the sender's public key.
- Properties: Must depend on the message, must use sender's secret information, must be easy to verify.
- Prevents repudiation: The signer cannot successfully deny signing the document.
- Distinguished from MAC: MAC is symmetric (shared key), whereas digital signatures are asymmetric (public-private key pair).

ElGamal Digital Signature Scheme

An asymmetric key signature scheme based on the difficulty of computing discrete logarithms in a finite field.

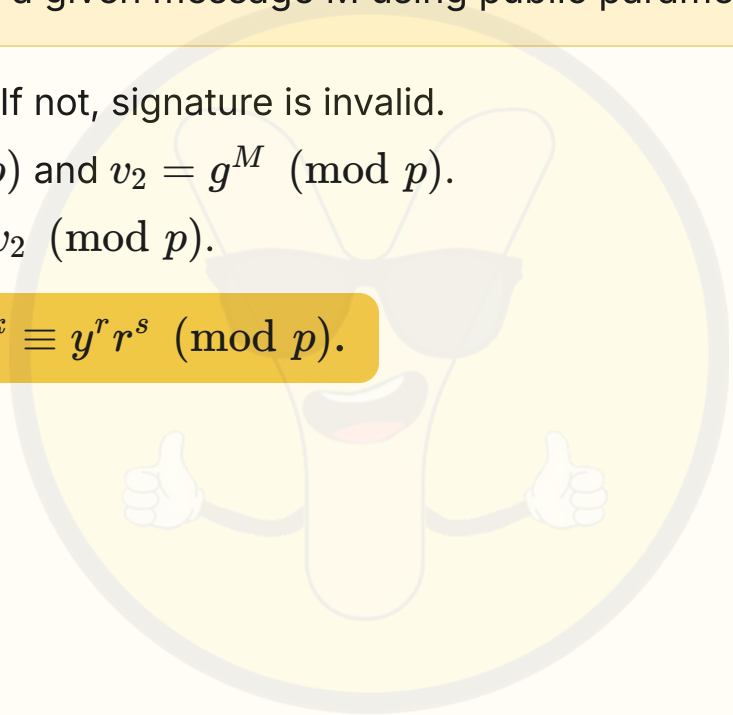
- Key Generation: Choose large prime p, generator g, secret key x, and public key $y = g^x \pmod p$.
- Signature Generation: Choose random integer k coprime to $p-1$. Compute $r = g^k \pmod p$ and $s = (M - xr)k^{-1} \pmod{p-1}$.
- Signature Pair: The signature for message M is the pair (r, s).
- Expansion: The signature size is twice the size of the prime p.

ElGamal Verification Equations

Verify an ElGamal signature pair (r, s) for a given message M using public parameters.

1. Check if $0 < r < p$ and $0 < s < p - 1$. If not, signature is invalid.
2. Compute two values: $v_1 = y^r r^s \pmod{p}$ and $v_2 = g^M \pmod{p}$.
3. Accept the signature if and only if $v_1 \equiv v_2 \pmod{p}$.

Verification holds because $g^M \equiv g^{xr+sk} \equiv y^r r^s \pmod{p}$.



Digital Signature Standard (DSS)

DIGITAL SIGNATURE ALGORITHM (DSA)

A NIST standard specified in FIPS 186 for digital signatures, based on the algebraic properties of modular exponentiation and discrete logarithms.

- Uses the Secure Hash Algorithm (SHA) to generate a message digest before signing.
- Components: Parameters p, q, g, private key x, and public key y.
- Generates a 320-bit signature (using standard 160-bit q and 1024-bit p) compared to ElGamal's larger size.
- Significantly faster for signature generation and verification in standard configurations.

DSS Generation and Verification Steps

Step-by-step mathematical flow of the Digital Signature Algorithm.

- Signing: Generate random k. Compute $r = (g^k \pmod p) \pmod q$.
- Compute $s = (k^{-1}(H(M) + xr)) \pmod q$. Signature is (r, s).
- Verification: Calculate $w = s^{-1} \pmod q$, $u_1 = (H(M)w) \pmod q$, $u_2 = (rw) \pmod q$.
- Compute $v = ((g^{u_1} y^{u_2} \pmod p)) \pmod q$.
- Valid if $v = r$.

Exam Corner

- **Q1** Differentiate between Message Authentication Code (MAC) and Hash Functions with suitable examples.
- **Q2** Explain the Birthday Attack on cryptographic hash functions and derive its complexity bound.
- **Q3** Describe the general architecture and round function steps of Secure Hash Algorithm (SHA-512).
- **Q4** What are the requirements for message authentication? Explain various attacks prevented by authentication.
- **Q5** Explain ElGamal Digital Signature Scheme with its key generation, signing, and verification algorithms.
- **Q6** Discuss the Digital Signature Standard (DSS) and write down its exact verification equations.

Unit 3 done. You are exam-ready.

Revise the Exam corner once more, then open Unit 4 on theycotes.com. Every AKTU subject, unit by unit, free. Notes by Chirag Jain, Ycotes.

