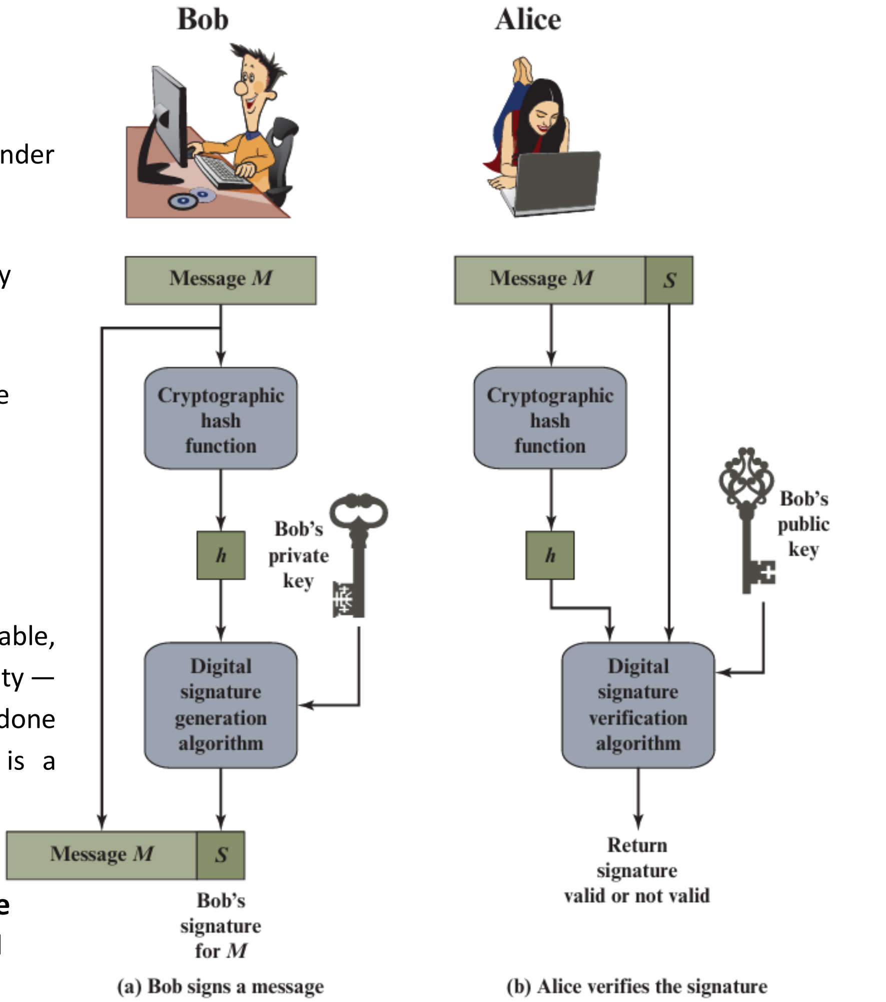
# Course: Data Security
## Lecturer: Dr. Mahmoud Yehia
## TA: Eng. Eslam Osama
# Digital Signatures

**Digital Signature** is a cryptographic technique that verifies the **authenticity**, **integrity**, and **origin** of a digital message or document. It is created using the sender’s **private key** and verified using their **public key**.


Bob
Alice
Message $M$
Cryptographic hash function
$h$
Bob’s private key
Digital signature generation algorithm
Message $M$ | $S$
Bob’s signature for $M$
Message $M$ | $S$
Cryptographic hash function
$h$
Bob’s public key
Digital signature verification algorithm
Return signature valid or not valid

(a) Bob signs a message

(b) Alice verifies the signature

**Figure 13.1** Simplified Depiction of Essential Elements of Digital Signature Process

## Key Features of Digital Signatures

1. **Authenticity**

   Confirms the message is from the claimed sender (verified via public key).

2. **Integrity**

   Ensures the message hasn’t been altered; any change breaks the signature.

3. **Non-repudiation**

   The sender cannot deny sending the message after signing it.

## Important Notes:

### - Third-Party Authentication in Digital Signatures

A digital signature must be publicly verifiable, meaning trusted third party can check its validity — not just the sender and receiver. This is often done using a **Certificate Authority (CA)**, which is a trusted third party.

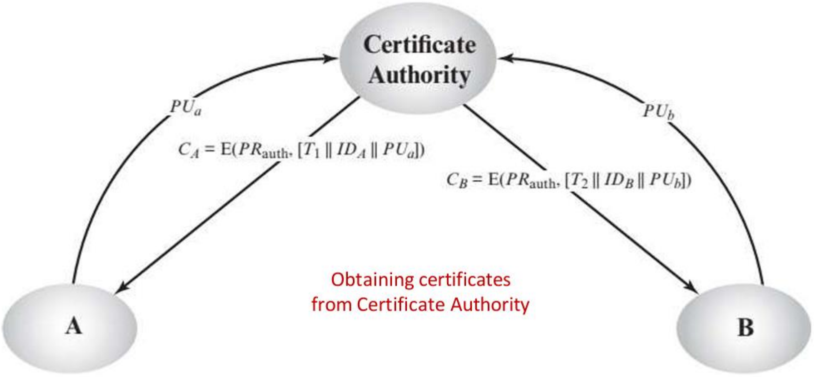
### - Signatures Are Not Unconditionally Secure

→Digital signatures are **not 100% secure** in all conditions. Here's why:

- An attacker might eventually:
- Break the math behind the algorithm.
- exploit weak random number generators.
- gain computing power (quantum computing in the future).

→try **forgery attacks** using public info or signed messages.

→That’s where **expiration Dates** comes into picture as it reduces risk over time, ensures systems regularly refresh and upgrade cryptographic methods.


Certificate Authority
$PU_a$
$PU_b$
$C_A = E(PR_{auth}, [T_1 \| ID_A \| PU_a])$
$C_B = E(PR_{auth}, [T_2 \| ID_B \| PU_b])$
A
B

Obtaining certificates from Certificate Authority

**Course: Data Security**
**Lecturer: Dr. Mahmoud Yehia**
**TA: Eng. Eslam Osama**
**Digital Signatures**

## Attacks and Forgeries

Assume **Alice is a server that runs an automated document-signing service** — a system that signs digital contracts when clients send requests. (**the entity tended to be attacked and forging its signature**).

| Attack Type | What the Attacker Does | Example with Alice | What the Attacker Knows | Threat Level |
|---|---|---|---|---|
| **Key-only Attack** | Tries to forge a signature knowing **only Alice's public key**. | Mallory downloads Alice's public key from her website and tries to create fake signed documents. | Public Key only | Low (very hard) |
| **Known Message Attack** | Uses a **set of valid signed documents** Mallory already has seen to help forge a new one. | Mallory collects some of Alice's old signed contracts and studies the signatures to learn patterns. | Public Key + Some (message, signature) pairs | Moderate |
| **Generic Chosen Message Attack** | Prepares a **fixed list of messages**, gets signatures on them from Alice (without knowing her key), then tries to break system. | Mallory sends 100 prepared contracts to Alice's API to be signed, hoping they'll reveal exploitable information. | Prepared message list (before any interaction) | Moderate–High |
| **Directed Chosen Message Attack** | Similar to Generic, but messages are chosen **after** seeing Alice's public key. | After getting Alice's public key, Mallory designs 100 specific contracts tailored to what the key reveals. | Public Key + Chooses messages after seeing it | High |
| **Adaptive Chosen Message Attack** | Mallory sends a message → sees signature → sends another, adjusting based on what she learns. | Mallory sends 1 contract, sees signature, then sends another, adjusting each to extract info and eventually forge. | Public Key + Can adapt messages mid-attack | Very High (realistic threat) |

## Notes:

**Key Difference: Generic Chosen Message Attack vs Known Message Attack**

1- For **Generic**, messages are **chosen by Mallory** (attacker), but for **known**, messages are **chosen by Alice** or occur naturally.

2- For **Generic**, Mallory **sends messages to Alice** to be signed, but for **known**, Mallory only **observes existing signed messages**.

**Key Difference: Generic vs Directed Chosen Message Attack**

1- For **Generic**, messages are chosen **before** knowing Alice's public key, but for **Directed**, they are chosen **after**.

2- For **generic**, messages are **random or generic** — not tailored, but for **Directed**, Messages are **strategically crafted** to exploit potential weaknesses in Alice's public key.

Course: Data Security
Lecturer: Dr. Mahmoud Yehia
TA: Eng. Eslam Osama

# Digital Signatures

## Key Difference: Adaptive Chosen Message Attack vs Directed Chosen Message Attack

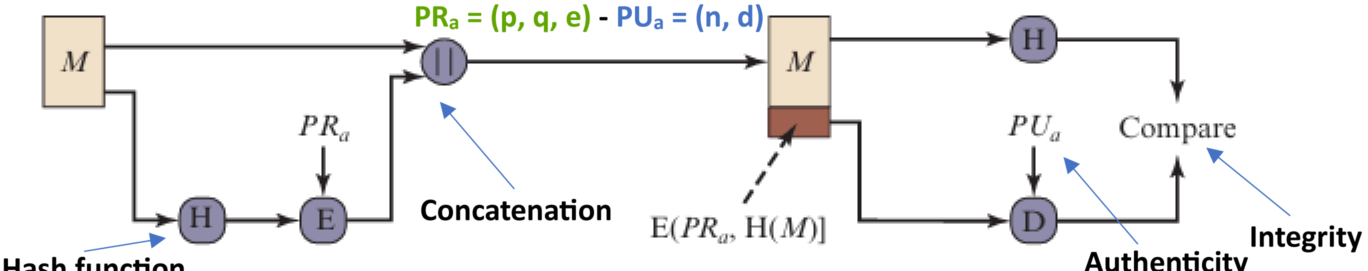
1- For **Directed**, all messages are chosen **after seeing** the public key but **before** seeing any signatures, but for **Adaptive**, Messages are chosen **dynamically**, one-by-one, based on signatures **already received**.

2- For **Directed**, static list — all messages sent at once, but for **Adaptive**, dynamic — attacker learns and **adjusts next message** based on previous results.

## Forgery Types Based on What the Attacker Achieves

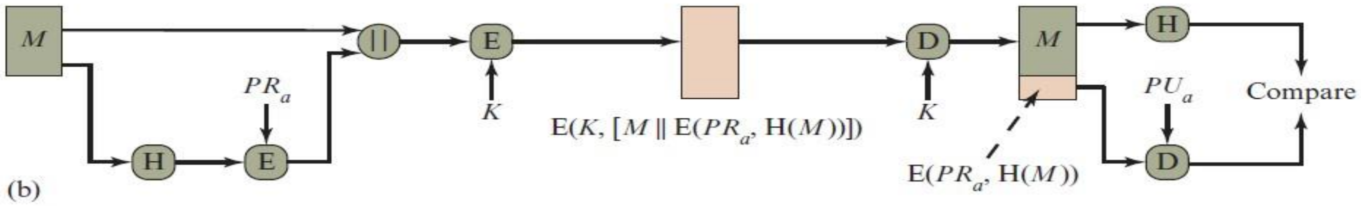
| Forgery Type | What It Means | Attacker's Goal / Example | What Attacker Knows | Threat Level |
|---|---|---|---|---|
| **Total Break** | Attacker recovers the signer's private key. | Mallory fully learns Alice's private key and can sign any document as Alice. | Everything (private key!) | Critical |
| **Universal Forgery** | Attacker creates a signing algorithm without needing private key. | Mallory can sign any document without the key, by finding a flaw in the algorithm. | Public key & flaws in algorithm | Severe |
| **Selective Forgery** | Attacker forges a signature for a specific message chosen ahead of time. | Mallory targets a specific contract ("ApproveLoan") and creates a valid fake signature for it. | Public key (and maybe prior messages) | High |
| **Existential Forgery** | Attacker forges a signature for at least one arbitrary message, not chosen by the attacker. | Mallory finds a weird message (e.g., a system log) that she didn't choose, but can create a valid signature for it. | Public key, maybe known signatures | Low (but dangerous) |

## RSA signature scheme (Direct Digital Signature)



$PR_a = (p, q, e)$ - $PU_a = (n, d)$

Note that, **No Confidentiality is provided**



**Confidentiality is ensured** by encrypting both the message and its digital signature**.** This encryption uses **a shared secret key**, known as symmetric encryption**.**

**Course: Data Security**
**Lecturer: Dr. Mahmoud Yehia**
**TA: Eng. Eslam Osama**
**Digital Signatures**

## Review Questions

### (1) List two disputes that can arise in the context of message authentication?

**Dispute 1:** The sender denies having sent the message (non-repudiation issue).

**Dispute 2:** The receiver claims the message was altered or forged after signing (integrity issue).

### (2) What are the properties a digital signature should have?

**Authenticity:** Confirms the sender's identity.

**Integrity:** Detects any changes to the signed message.

**Non-repudiation:** Prevents the sender from denying the signature later.

### (3) What requirements should a digital signature scheme satisfy?

**Verifiability:** Third parties must be able to verify signatures.

**Security:** It should be computationally infeasible to forge signatures without the private key as only the signer can generate a valid signature for a message.

### (4) What is the difference between direct and arbitrated digital signature?

**Direct digital signature:** The receiver verifies the signature directly using the sender's public key, without a trusted third party.

**Arbitrated digital signature:** A trusted third party (arbitrator) is involved in verifying and managing signatures, which can help resolve disputes.

### (5) In what order should the signature function and the confidentiality function be applied to a message, and why?

**First:** The message is digitally signed (signature function).

**Then:** The signed message is encrypted for confidentiality.

**Reason:** Signing first ensures the signature covers the original message and can be verified after decryption. Encrypting first would prevent signature verification without decrypting first, which can cause security and efficiency issues.

### (6) What are some threats associated with a direct digital signature scheme?

**Key-only attack:** Attacker tries to forge signatures knowing only the public key.

**Known message attack:** Attacker uses known signed messages to attempt forgery.

**Chosen message attacks:** Attacker obtains signatures on messages of their choice to help forge other signatures (generic, directed, adaptive).