جامـــعـة المــــسـتـقـبـل
AL MUSTAQBAL UNIVERSITY

قســم الامـــن الـــــســيبراني

Department of Cyber Security

Subject:

Public key encryption

Class:

third

Lecturer:

Asst. Lecturer Qusai Al-Durrah

Lecture (9 & 10):

Digital Signature Standard (DSS)

1. Introduction

As digital communication becomes increasingly central to modern systems, ensuring the **authenticity**, **integrity**, and **non-repudiation** of electronic documents is essential. Traditional handwritten signatures cannot secure digital content against tampering or impersonation. Therefore, specialized cryptographic techniques are required to authenticate digital information.

The **Digital Signature Standard (DSS)** is a federal standard (FIPS) developed by the **National Institute of Standards and Technology (NIST)**. DSS specifies approved algorithms for generating and verifying **digital signatures**. Unlike encryption systems, DSS's primary purpose is **authentication**, not confidentiality. It ensures that a received message truly originated from the claimed sender and was not altered during transmission.

The core algorithm included in DSS is the **Digital Signature Algorithm (DSA),** which works alongside secure hashing (SHA-2 or SHA-3) and public-key infrastructure (PKI). DSS has become a global benchmark for secure digital communication and is widely used in government, finance, e-commerce, and secure software distribution.

2. Learning Outcomes

By the end of this lecture, students will be able to:

1. **Define**: the Digital Signature Standard (DSS) and explain its purpose.
2. **Describe**: how DSA, SHA, and PKI work together to produce secure digital signatures.
3. **Explain**: the **signature generation and verification** steps in DSS.
4. **Differentiate**: DSS from encryption-based systems and symmetric/asymmetric key methods.
5. **Evaluate:** the advantages, limitations, and security properties of DSS.
6. **Understand**: how DSS ensures authenticity, integrity, and non-repudiation in digital transactions.

3. What Is the Digital Signature Standard (DSS)?

According to NIST, the **Digital Signature Standard (DSS)** is a federal standard that specifies the methods for:

- Generating digital signatures
- Verifying digital signatures
- Managing the required cryptographic keys

DSS provides a standardized framework for guaranteeing the authenticity and integrity of electronic documents. It does **not** provide encryption or key exchange capabilities; its focus is solely **digital authentication**.

DSS relies on three main components:

1. **Digital Signature Algorithm (DSA)**
2. **Secure Hash Algorithm (SHA-2 or SHA-3)**
3. **Public Key Infrastructure (PKI)**

Together, these components guarantee that only the legitimate owner of a private key can produce a valid digital signature.

4. Components of DSS

4.1 Digital Signature Algorithm (DSA)

DSA is the core signature algorithm defined by DSS. It uses:

- A private key → to generate a signature
- A public key → to verify a signature

DSA signs the **hash** of a message, not the message itself. This provides:

- Efficiency
- Security
- Fixed-length signatures

DSA produces two values:
(r, s)
which together form the digital signature.

4.2 Secure Hash Algorithm (SHA)

Before signing, DSS computes a **cryptographic hash** of the message using SHA.

Properties of SHA:

- Produces a fixed-size output ("fingerprint")
- Any change to the message results in a different hash
- Detects tampering instantly

DSS requires modern secure versions such as:

- **SHA-256**
- **SHA-384**
- **SHA-512**

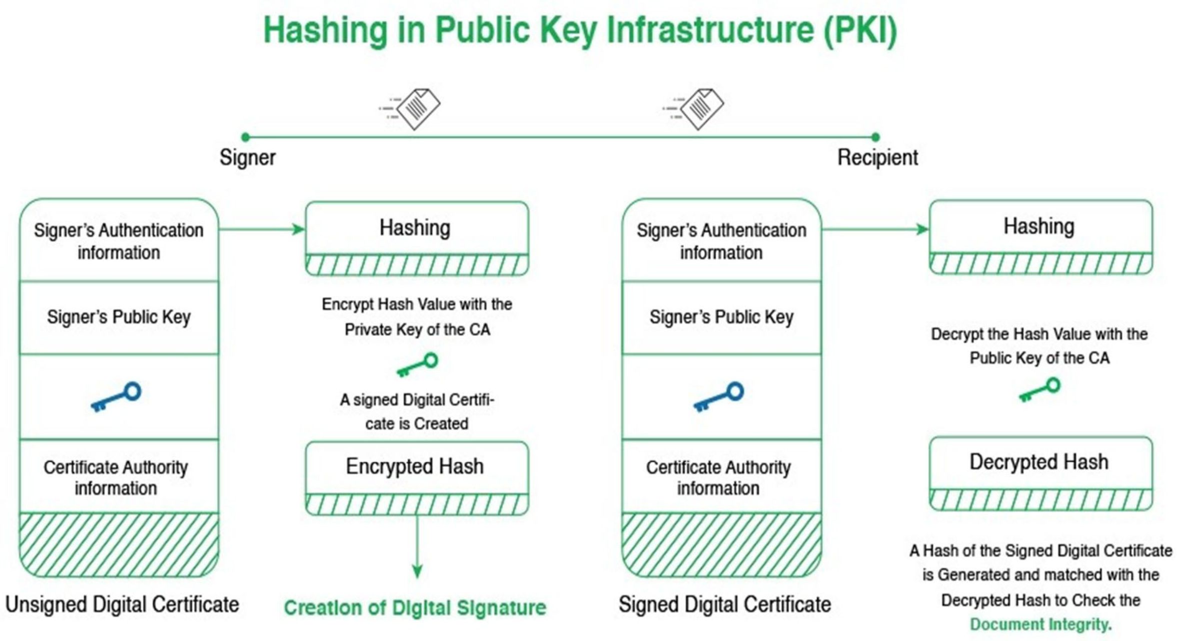
4.3 Public Key Infrastructure (PKI)

DSS operates within a larger trust framework known as PKI, which includes:

- **Private keys** – used for signing
- **Public keys** – used for verification
- **Certificate Authorities (CAs)** – validate user identities

PKI ensures:

- Verified identity of the signer
- Secure distribution of public keys


Hashing in Public Key Infrastructure (PKI)
Signer
Recipient
Signer's Authentication information
Signer's Public Key
Certificate Authority information
Unsigned Digital Certificate
Hashing
Encrypt Hash Value with the Private Key of the CA
A signed Digital Certificate is Created
Encrypted Hash
Creation of Digital Signature
Signer's Authentication information
Signer's Public Key
Certificate Authority information
Signed Digital Certificate
Hashing
Decrypt the Hash Value with the Public Key of the CA
Decrypted Hash
A Hash of the Signed Digital Certificate is Generated and matched with the Decrypted Hash to Check the Document Integrity.

5. Why Digital Signatures Are Important

Digital signatures provide several key security properties:

5.1 Authenticity

Ensures the signature was created using the sender's private key.

5.2 Integrity

If the message changes, signature verification fails.

5.3 Non-Repudiation

The sender cannot later deny generating the signature.

5.4 Legal Validity

Digital signatures that comply with DSS, ESIGN, UETA, and eIDAS have the same legal standing as handwritten signatures.

6. How DSS Works

The DSS process consists of two major phases:

6.1 Signature Generation (Sender Side)

Step 1: Hash the message

The sender computes:
H = SHA(M)

Step 2: Generate random value (k)
This (k) must be unique for each signature.

Step 3: Use the private key to compute signature values
(r, s)

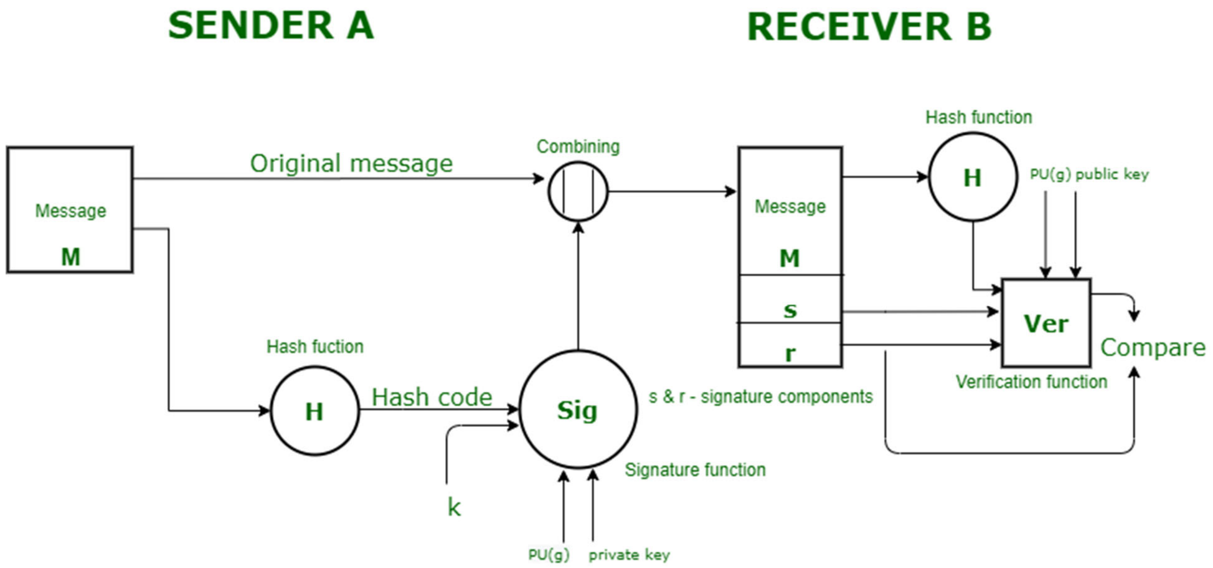
Step 4: Send the message and signature
The transmitted packet is:
(M, r, s)

6.2 Signature Verification (Receiver Side)

Step 1: Hash the message
H' = SHA (M)

Step 2: Use sender's public key and signature values (r, s)
These values are fed into a verification function.

Step 3: Compare
If:
H = H'
the signature is valid.
Otherwise, the message or signature was tampered with.



7. DSS vs. Symmetric and Asymmetric Encryption

Feature	**DSS**	**Symmetric Encryption**	**Asymmetric Encryption**
Purpose	Authentication only	Confidentiality	Key exchange / confidentiality
Keys Used	Private + Public	One shared secret key	Public/private key pair
Affected Data	Hash of message	Entire message	Message
Algorithm Type	Signature algorithm	Block/stream ciphers	RSA/ElGamal

DSS does **not** encrypt data — it only authenticates.

8. Example Scenario of DSS in Practice

Imagine a company emailing an electronically signed contract:

1. The sender's system computes SHA-256 of the contract.
2. DSA signs the hash using the sender's private key.
3. The contract + signature are emailed.

4. The receiver verifies using the sender's public key.
5. If verification succeeds:
   - The document is unchanged
   - The sender's identity is confirmed

This process is widely used in:

- Government e-services
- Banking
- E-commerce
- Software updates
- Secure email

9. Advantages of DSS

1. **Extremely secure** due to strong hashing and asymmetric keys
2. **Detects any tampering** instantly
3. **Legally recognized**
4. **Prevents impersonation**
5. **Provides a timestamp** proving when signature was created
6. **Cannot be forged or duplicated**
7. **Ensures identity validation** through PKI
8. **Tamper-evident and traceable**

10. Limitations of DSS

1. Requires updated software and drivers for compatibility
2. Relies on secure storage of private keys
3. Key loss → signature invalid forever
4. Requires purchasing digital certificates
5. Some business services must be DSS-compliant
6. Verification software may add cost
7. Technical complexity for non-technical users
8. Algorithms must be updated when standards evolve
9. Requires secure PKI infrastructure

11. Summary

The Digital Signature Standard (DSS) provides a robust framework for verifying digital identities and ensuring document integrity in electronic communication. It combines the Digital Signature Algorithm (DSA), secure hash functions (SHA), and PKI to generate signatures that cannot be forged or modified. DSS enables secure authentication, supports global legal compliance, and protects organizations from fraud and tampering.

Digital signatures are essential in modern cybersecurity, forming the backbone of secure online transactions, software distribution, and digital document management.