

전자서명의 이론적 안전성 이해

이형태

중앙대학교 소프트웨어학부

2026. 02. 24 @ 2026 KpqC Winter Camp

Part I. Foundations

- What is a Digital Signature?
- Correctness
- EUF-CMA Security
- Strong Unforgeability

Part II. Code-Based Signatures

- Error-Correcting Codes
- Syndrome Decoding (SD)
- CFS Signature Scheme
- Security of CFS

Goal of This Talk

Understand the formal definition of digital signatures, their theoretical security notions, and how these concepts appear in a concrete example.

What is a Digital Signature?

Handwritten Signature

- Unique to the person who signs
- Easy to verify by looking at it
- Only works on physical documents

Digital Signature

- Generated using a private key
- Unique per message — can't be reused
- Anyone can verify using the public key

Authentication

Proves who sent the message. The signer's identity is verified.

Integrity

Detects if the document was changed after signing.

Non-repudiation

Signer cannot deny later. Math proves who signed.

Think of it as a seal of approval

Only you can create it, but anyone can check it is real.

Three Components of a Digital Signature Scheme

KeyGen(λ)

Input:

- Security level λ

Output:

- Signing key sk (private)
- Verification key vk (public)

Run once to set up the scheme

Sign(sk, m)

Input:

- Signing key sk
- Message m

Output:

- Signature σ

Run by the sender to sign

Verify(vk, m, σ)

Input:

- Verification key vk
- Message m , signature σ

Output:

- Accept (1) / Reject (0)

Run by anyone to check validity

Correctness of a Digital Signature Scheme

Informal Meaning

“If you sign a message honestly, the verifier will always accept it.”

In other words, a legitimately generated signature must never be rejected.

Formal Definition

A signature scheme $(\text{KeyGen}, \text{Sign}, \text{Verify})$ is **correct** if for all security parameters λ , all messages m , and all key pairs (sk, vk) output by $\text{KeyGen}(\lambda)$:

$$\Pr \left[\text{Verify}(\text{vk}, m, \text{Sign}(\text{sk}, m)) = 1 \right] = 1.$$

- For schemes with negligible error, the probability may be $\geq 1 - \text{negl}(\lambda)$, not exactly 1.

What Correctness Does NOT Guarantee

- It says nothing about *forged* signatures being rejected.
- Security (unforgeability) is a separate property.

Real-Life Threat Scenario: Online Banking Forgery

Scenario

- Alice authorizes bank transfers using digital signatures.
- Mallory installs malware on Alice's device.
- Mallory can submit transfer requests on Alice's behalf.

1 Attacker Chooses Messages

- Mallory submits carefully chosen transfer requests:
 - "Send \$123.45 to Bob"
 - "Send \$987.65 to Carol"
- Alice's system automatically signs them.

2 Signature Collection

- Mallory obtains valid signatures on her chosen messages.
- She repeats this process many times.

3 Forgery Attempt

- Mallory attempts to forge a signature on "Send \$1,000,000 to Mallory".

EUF-CMA Security: Formal Game Definition

- (Informal) Existential Unforgeability against Chosen Message Attacks (EUFCMA)
 - An attacker who can get signatures on messages of their choice **cannot** forge a valid signature on any **new** message.
 - Most digital signature schemes aim for this level of security.

Definition (EUFCMA Security)

A signature scheme $(\text{KeyGen}, \text{Sign}, \text{Verify})$ is *existentially unforgeable under chosen-message attacks (EUFCMA)* if for every PPT adversary \mathcal{A} , the adversary's winning probability in the following game is negligible in the security parameter λ :

- 1 **Setup:** The challenger runs $(\text{vk}, \text{sk}) \leftarrow \text{KeyGen}(\lambda)$ and gives vk to \mathcal{A} .
- 2 **Query Phase:** \mathcal{A} adaptively queries a signing oracle on messages m_i of its choice and receives $\sigma_i \leftarrow \text{Sign}(\text{sk}, m_i)$ for $1 \leq i \leq q$.
- 3 **Forgery:** \mathcal{A} outputs a pair (m^*, σ^*) .

The adversary wins if $\text{Verify}(\text{vk}, m^*, \sigma^*) = 1$ and $m^* \notin \{m_1, \dots, m_q\}$.

Stronger Security Notion: Strong Unforgeability

Strong Unforgeability under Chosen Message Attack (SUF-CMA)

The adversary wins if it outputs (m^*, σ^*) such that

$$\text{Verify}(\text{vk}, m^*, \sigma^*) = 1 \quad \text{and} \quad (m^*, \sigma^*) \notin \{(m_i, \sigma_i)\}.$$

- **Key difference from EUF-CMA:** Even generating a different valid signature for a previously signed message is considered a forgery.

Why Strong Unforgeability Matters in Practice

Scenario: Blockchain Transaction

- Alice signs a transaction: “Send 10 coins to Bob”
- The transaction is identified by the *hash of the signature*.

What Could Go Wrong?

- Many real-world signature schemes are randomized.
- The same message can have *multiple valid signatures*.
- An attacker observes Alice’s valid signature.
- The attacker generates a **different valid signature** on the same message.

Security Problem

- The system may treat the new signature as a *new transaction*.
- This can lead to replay attacks or double execution. → Need strong unforgeability!

Error Correcting Codes

sender \mathbf{c} $\xrightarrow{\text{noise } \mathbf{e}}$ $\mathbf{s} = \mathbf{c} + \mathbf{e}$ receiver

- \mathcal{C} : a **linear code** of length n and dimension k
 \Rightarrow a k -dimensional subspace of \mathbb{F}_q^n
e.g., Hamming codes, Reed-Solomon codes, Golay codes, Goppa codes and so on.

- a **generator matrix** \mathbf{G} for a code \mathcal{C} : a $k \times n$ matrix such that

$$\mathcal{C} = \{\mathbf{m} \cdot \mathbf{G} : \mathbf{m} \in \mathbb{F}_q^k\}$$

- a **parity check matrix** \mathbf{H} : a matrix such that $\mathbf{H} \cdot \mathbf{G}^\top = 0$ (and so $\mathbf{H} \cdot \mathbf{c}^\top = 0$ for all codewords \mathbf{c})
- We can easily compute a $(n - k) \times n$ parity check matrix \mathbf{H} from given \mathbf{G} using Gaussian elimination.

The Hard Problem: Syndrome Decoding (SD)

Syndrome Decoding (SD) Problem

Given:

- Matrix \mathbf{H} of size $(n-k) \times n$ over \mathbb{F}_2
- Target syndrome vector $\mathbf{s}^\top = \mathbf{H} \cdot \mathbf{e}^\top$
- Target weight threshold w

Find: Error vector \mathbf{e} with $\text{wt}(\mathbf{e}) \leq w$, where $\mathbf{H} \cdot \mathbf{e}^\top = \mathbf{s}^\top \pmod{2}$.

- Matrix Equation $\mathbf{H} \cdot \mathbf{e}^\top = \mathbf{s}^\top$:
$$\begin{pmatrix} 1 & 0 & 1 & 1 & 0 \\ 0 & 1 & 0 & 1 & 1 \\ 1 & 1 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} e_1 \\ e_2 \\ e_3 \\ e_4 \\ e_5 \end{pmatrix} = \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix} = \begin{pmatrix} s_1 \\ s_2 \\ s_3 \end{pmatrix}.$$

Find \mathbf{e} with few 1s (weight $\leq w = 3$). For large n , this is **HARD**!

- It is known that **the SD problem is NP-hard for random \mathbf{H} !**

Key Idea of CFS Signature

- CFS Signature: A representative code-based signature scheme
 - N. Courtois, M. Finiasz, and N. Sendrier, How to achieve a McEliece-based digital signature scheme, Asiacrypt 2001.

Key Idea

- Design a signature scheme based on the hardness of the SD problem

$$\underbrace{\mathbf{H}}_{\text{verification key vk}} \cdot \underbrace{\mathbf{e}^T}_{\text{signature}} = \underbrace{\mathcal{H}(m)}_{\text{hash of message}} \quad \text{with wt}(\mathbf{e}) \text{ small}$$

- For a random matrix \mathbf{H} , finding a low-weight solution \mathbf{e} is hard.
- Use a Goppa code parity-check matrix \mathbf{H}_G with trapdoor decoding capability, concealed via suitable transformations.

$$\text{verification key: } \mathbf{H} = \mathbf{S} \cdot \mathbf{H}_G \cdot \mathbf{P}$$

– \mathbf{S} : a $(n - k) \times (n - k)$ invertible matrix, \mathbf{P} : a $n \times n$ permutation matrix

Description of CFS Signature

KeyGen

- 1 Generate a Goppa-code parity check matrix $\mathbf{H}_G \in \mathbb{F}_2^{(n-k) \times n}$.
- 2 Generate random $(n-k) \times (n-k)$ invertible matrix \mathbf{S} and $n \times n$ permutation matrix \mathbf{P} .
- 3 $\text{vk} = \mathbf{H} (= \mathbf{S} \cdot \mathbf{H}_G \cdot \mathbf{P}) \in \mathbb{F}_2^{(n-k) \times n}$ and $\text{sk} = (\mathbf{S}, \mathbf{H}', \mathbf{P})$.

Sign

- 1 Compute $\mathbf{x} = D_{\mathbf{H}'}(\mathbf{S}^{-1} \cdot \mathcal{H}(M \parallel \alpha))$ until \mathbf{x} is found for $\alpha \xleftarrow{\$} \mathbb{F}_2^{n-k}$.
- 2 Output (α, \mathbf{s}) where $\mathbf{s} = \mathbf{x} \cdot \mathbf{P}$.

Verify

Output 1 iff $\text{wt}(\mathbf{s}) \leq t$ and $\mathbf{H} \cdot \mathbf{s}^\top = \mathcal{H}(M \parallel \alpha)$.

$$(\because \mathbf{H} \cdot \mathbf{s}^\top = (\mathbf{S} \cdot \mathbf{H}' \cdot \mathbf{P})(\mathbf{x} \cdot \mathbf{P})^\top = \mathbf{S} \cdot \mathbf{H}' \cdot \mathbf{x}^\top = \mathbf{S} \cdot \mathbf{S}^{-1} \cdot \mathcal{H}(M \parallel \alpha))$$

Security of CFS Signature

- Security relies on the hardness of the **Syndrome Decoding (SD)** problem:

Given $\mathbf{H} \in \mathbb{F}_2^{(n-k) \times n}$ and \mathbf{y} , find \mathbf{s} such that $\mathbf{H} \cdot \mathbf{s}^\top = \mathbf{y}$ and $\text{wt}(\mathbf{s}) \leq t$.

- If \mathbf{H} is uniformly random, finding such a low-weight \mathbf{s} is computationally hard.
- In CFS, however, \mathbf{H} has special structure ($\mathbf{H} = \mathbf{S} \cdot \mathbf{H}_G \cdot \mathbf{P}$). To argue security, we additionally assume that the decisional McEliece problem defined below is computationally hard.

This problem is also known as the Distinguishing Goppa Code problem.

Decisional McEliece Problem: DMcE(n, k, t)

Given $\mathbf{H} \in \mathbb{F}_2^{(n-k) \times n}$, distinguish whether

- \mathbf{H} is uniformly random over $\mathbb{F}_2^{(n-k) \times n}$, or
- $\mathbf{H} = \mathbf{S} \cdot \mathbf{H}_G \cdot \mathbf{P}$ for a hidden Goppa code.

Security of CFS Signature (Cont.)

- We construct a reduction that uses an adversary breaking the EUF-CMA security of CFS to solve the Syndrome Decoding (SD) problem.

Goal

Use an EUF-CMA forger \mathcal{A} to solve a target **Syndrome Decoding (SD)** instance.

- **Given:** An SD challenge $(\mathbf{H}, \mathbf{y}^*)$, and auxiliary instances $(\mathbf{s}_i, \mathbf{y}_i)_{1 \leq i \leq q}$ for simulation.
- **Public Key:** Set $\text{vk} = \mathbf{H}$, which is computationally indistinguishable from a valid CFS public key under the DMcE assumption.
- **Answer Signing Queries:** For each signing query M_i , choose α_i and program the random oracle as $\mathcal{H}(M_i \| \alpha_i) = \mathbf{y}_i$. Use the known solution \mathbf{s}_i such that $\mathbf{H}\mathbf{s}_i^\top = \mathbf{y}_i$ to respond.
- **Target Programming:** Program the random oracle at (M^*, α^*) so that $\mathcal{H}(M^* \| \alpha^*) = \mathbf{y}^*$.
- **Extraction:** If \mathcal{A} outputs a valid forgery $(M^*, \alpha^*, \mathbf{s}^*)$, then

$$\mathbf{H}\mathbf{s}^{*\top} = \mathbf{y}^*,$$

thus solving the target SD instance.

Security of CFS Signature (Cont.)

Theorem (Security of CFS Signature)

The CFS signature scheme is EUF-CMA secure if the decisional McEliece problem and the syndrome decoding problem is computationally infeasible in the random oracle model.

Summary

What We Learned

- A digital signature must satisfy **correctness** and **unforgeability**.
- EUF-CMA models realistic chosen-message attacks.
- Strong unforgeability prevents even re-signing attacks.
- As an example, the CFS signature reduces security to the hardness of the Syndrome Decoding problem (and additional assumptions).

Big Picture

Modern digital signature security is established by reducing forgery attacks to well-studied computationally hard problems.

Questions?

Thank you for your attention!