A Linearly Homomorphic Signature Scheme From Weaker Assumptions

TECHNISCHE UNIVERSITÄT DARMSTADT

Lucas Schabhüser, Patrick Struck, Johannes Buchmann

Organization

TECHNISCHE UNIVERSITÄT DARMSTADT

- Motivation
- Homomorphic Signature Schemes
- Our Construction
- Combining our Construction with Homomorphic Encryption
- Conclusion

Motivation

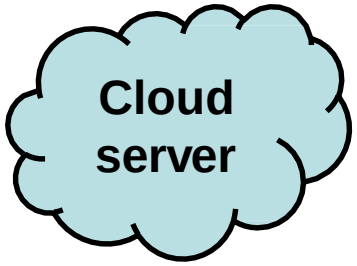
Alice

Cloud server

Motivation

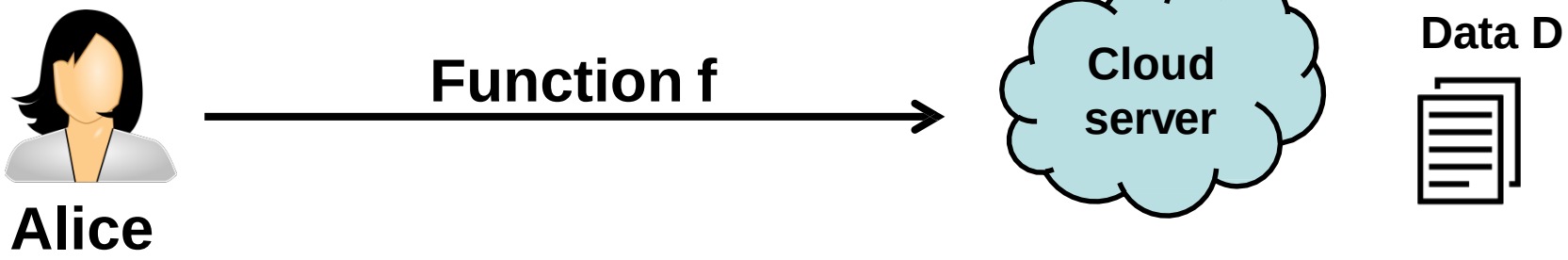
Motivation

Alice

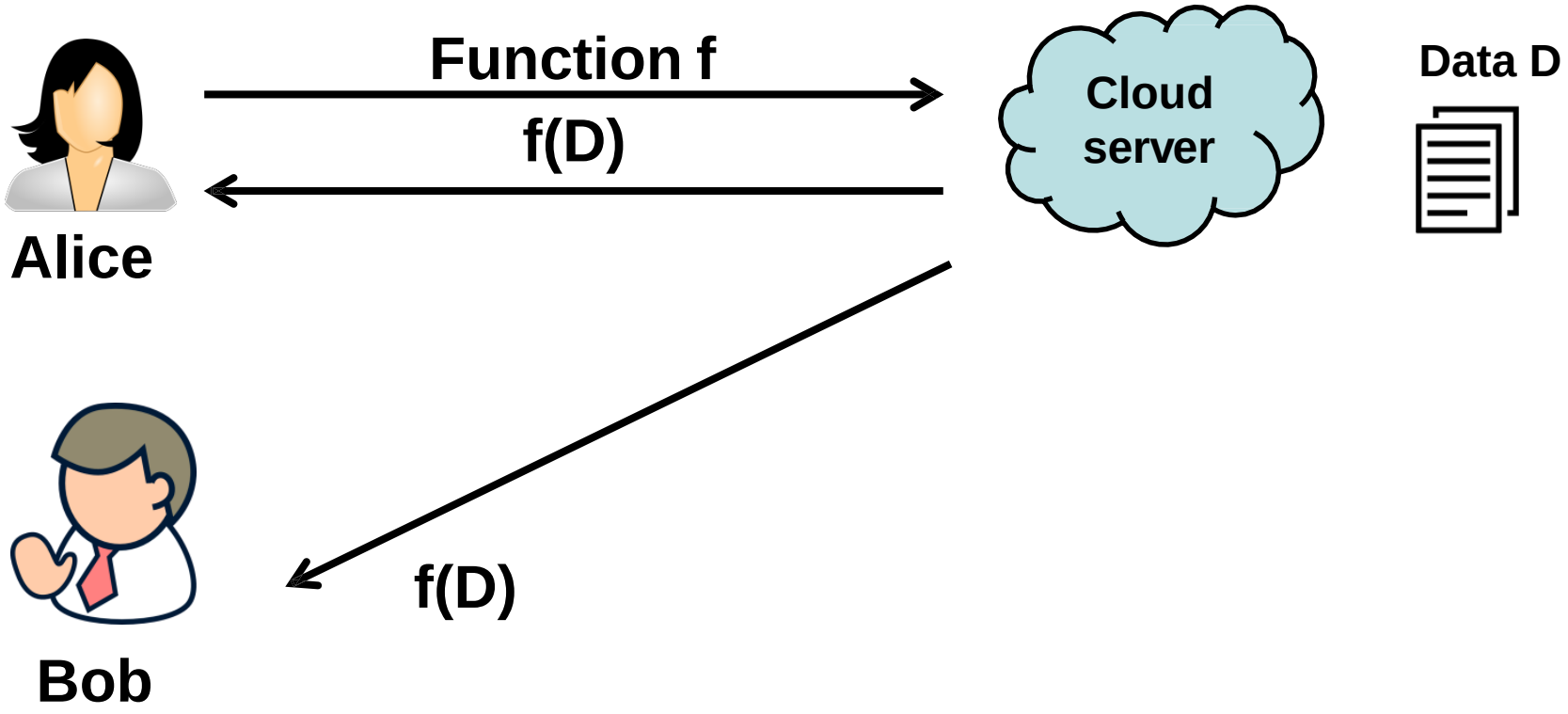


Data D

Motivation



Motivation



Motivation

TECHNISCHE UNIVERSITÄT DARMSTADT

- Alice or Bob does not trust the cloud
  - Computation result has to be verifiable
  - **Solution**: homomorphic signature scheme
- Sensitive data
  - Data has to be confidential
  - **Solution**: homomorphic encryption scheme

Homomorphic Signature Schemes

TECHNISCHE UNIVERSITÄT DARMSTADT

- Messages are stored in datasets identified by an identifier Δ
- Typically, the dataset size is fixed by a value k
- Functions can only be evaluated over messages in the same dataset

	Δ_1	Δ_2	Δ_3
$i = 1$:	m_1	m^*_1	m'_1
$i = 2$:	m_2	m^*_2	m'_2
$i = 3$:	m_3	m^*_3	m'_3
...	...	...	...
...	...	...	...
...	...	...	...
$i = k$:	m_k	m^*_k	m'_k

Homomorphic Signatures

TECHNISCHE UNIVERSITÄT DARMSTADT

$\mathsf{HKeyGen}(1^\lambda)$: Output: key pair $(\mathsf{sk}, \mathsf{pk})$.

$\mathsf{HSign}(\mathsf{sk}, \Delta, \tau, m)$: Output: Signature σ.

$\mathsf{HEval}(\mathsf{pk}, f, \boldsymbol{\sigma})$: Output: Signature σ.

$\mathsf{HVerify}(\mathsf{pk}, (f, \tau_1, \ldots, \tau_n, \Delta), m, \sigma)$: Output: 0 or 1.

Homomorphic Signatures

TECHNISCHE UNIVERSITÄT DARMSTADT

Security Intuition: No signatures for $m \neq f(m_1, \ldots, m_k)$ can be derived.

Weak security: Adversary can not query messages adaptively.

Strong security: Adversary **can** query messages adaptively.

Homomorphic Signatures

TECHNISCHE UNIVERSITÄT DARMSTADT

Existing solutions for strongly secure linearly homomorphic signatures all use „strong“ assumptions.

[BFKW09]: 2-out-of-3 CDH Assumption
[CFW11]: Strong RSA Assumption
[ALP13]: k-simultaneous Flexible Pairing Assumption
[CFN15]: Flexible DH Inversion
…
This work: CDH Assumption

„Weak“ vs „Strong“Assumptions

TECHNISCHE UNIVERSITÄT DARMSTADT

Unique solution, example CDH:

Given: $(\mathbb{G}, g, g^a, g^b)$

Compute: g^{ab}

Class of solutions, example 2-out-of 3-CDH:

Given: $(\mathbb{G}, g, h, g^a)$

Compute: (g^r, h^{ar}) for some r

Preliminaries

TECHNISCHE UNIVERSITÄT DARMSTADT

Bilinear Groups: $e : \mathbb{G}_1 \times \mathbb{G}_2 \rightarrow \mathbb{G}_T$

$$e(g_1, g_2) \neq 1_{\mathbb{G}_T} \forall g_1 \neq 1_{\mathbb{G}_1}, g_2 \neq 1_{\mathbb{G}_2}$$
$$e(g_1^a, g_2^b) = e(g_1, g_2)^{ab}$$

If $\mathbb{G}_1 \simeq \mathbb{G}_2$, we write: $\mathbb{G}$

We will describe our scheme in symmetric groups (for notations sake).
It works in (more efficient) asymmetric groups as well.

Messages: $m = \begin{pmatrix} m[1] \\ m[2] \\ \dots \\ m[T] \end{pmatrix} \in \mathbb{Z}_q^T$

Our Solution

TECHNISCHE UNIVERSITÄT DARMSTADT

$\mathsf{HKeyGen}(1^\lambda)$: Obtains a bilinear group $\mathsf{bgp} = (q, \mathbb{G}, \mathbb{G}_T, g, e)$, and samples $k + T$ elements $R_1, \ldots, R_k, h_1, \ldots, h_T \leftarrow \mathbb{G}$. Additionally it generates a key pair $(\mathsf{sk}', \mathsf{pk}') \leftarrow \mathsf{KeyGen}'(1^\lambda)$ of a regular signature scheme and a key $K \leftarrow \mathcal{K}$ for a pseudorandom function $F : \mathcal{K} \times \{0,1\}^* \rightarrow \mathbb{Z}_q$. Output $\mathsf{sk} = (\mathsf{sk}', K)$ and $\mathsf{pk} = (\mathsf{pk}', \mathsf{bgp}, \{h_j\}_{j=1}^T, \{R_i\}_{i=1}^k)$.

$\mathsf{HSign}(\mathsf{sk}, \Delta, i, m)$: It runs $z \leftarrow F_K(\Delta)$ and computing $Z = g^z$. It binds Z to the dataset identifier Δ by using the regular signature scheme, i.e. it sets $\sigma_\Delta \leftarrow \mathsf{Sign}'(\mathsf{sk}', (Z|\Delta))$. Then, it computes $\Lambda \leftarrow (R_i \cdot \prod_{j=1}^T h_j^{m[j]})^z$ and returns the signature $\sigma = (\sigma_\Delta, Z, \Lambda)$.

Our Solution

TECHNISCHE UNIVERSITÄT DARMSTADT

$\mathsf{HEval}(\mathsf{pk}, f, \boldsymbol{\sigma})$: It computes $\Lambda = \prod_{i=1}^{k} \Lambda_i^{f_i}$, and returns the signature $\sigma = (Z_1, \sigma_{\Delta,1}, \Lambda)$.

$\mathsf{HVerify}(\mathsf{pk}, (f, \tau_1, \ldots, \tau_k, \Delta), m, \sigma)$: It checks if $\mathsf{Verify}'(\mathsf{pk}', (Z|\Delta), \sigma_\Delta) = 1$ and $e\left(R \cdot \prod_{j=1}^{T} h_j^{m[j]}, Z\right) = e(\Lambda, g)$, where $R \leftarrow \prod_{i=1}^{k} R_i^{f_i}$. If both checks pass, it outputs 1, otherwise, it returns 0.

Our Solution

TECHNISCHE UNIVERSITÄT DARMSTADT

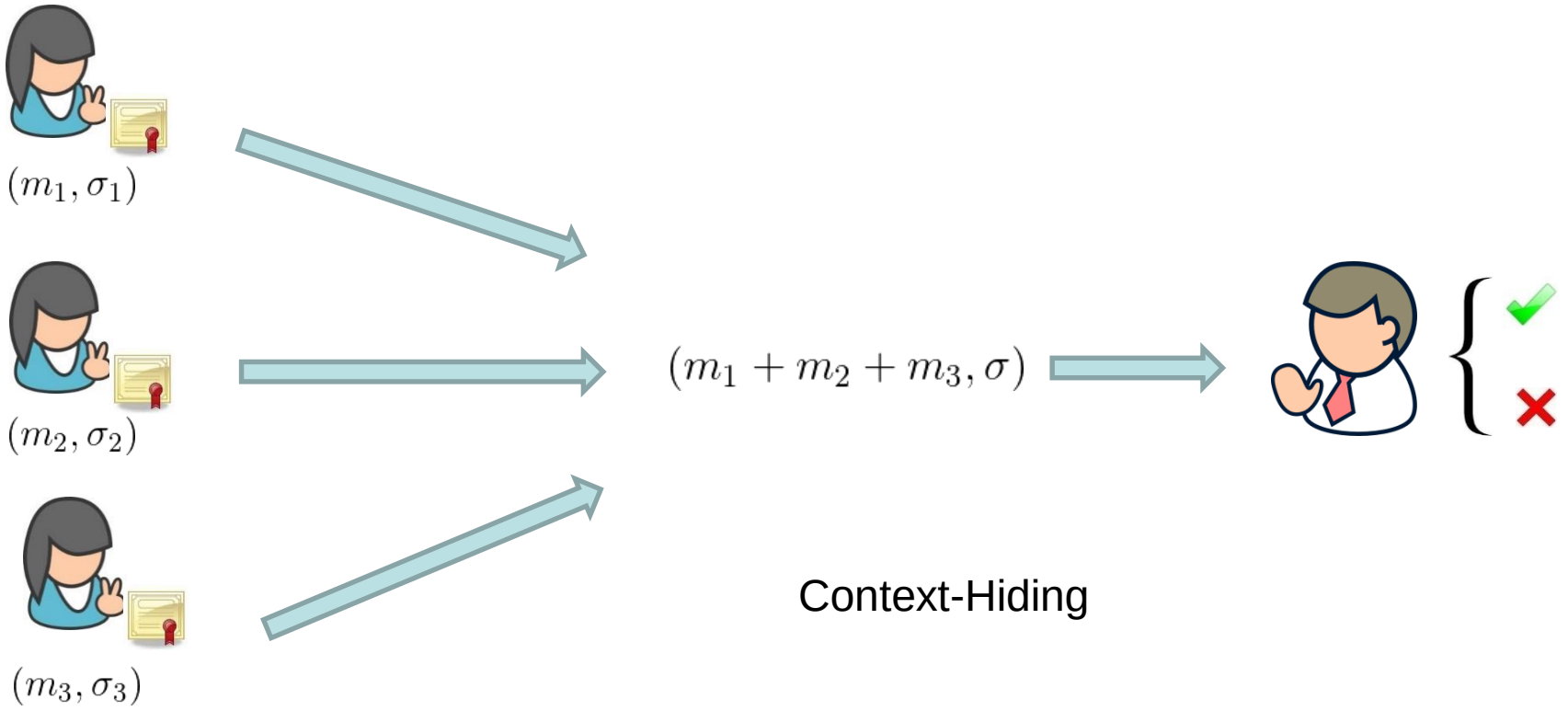
Strongly secure under CDH in $\mathbb{G}$ in the standard model, if *F* is pseudorandom and *Sig* is an unforgeable (conventional) signature scheme.

Constant-time verification, after a one time preprocessing.

Perfectly context hiding.

Can be combined with homomorphic encryption.

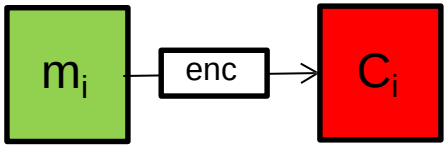
Input Privacy



Linearly Homomorphic Authenticated Encryption with Public Verifiability (LAEPuV)

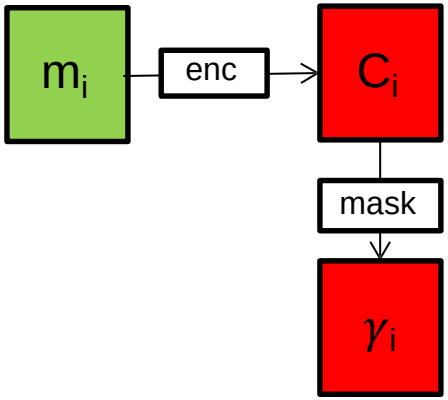
Linearly Homomorphic Authenticated Encryption with Public Verifiability (LAEPuV)

TECHNISCHE UNIVERSITÄT DARMSTADT



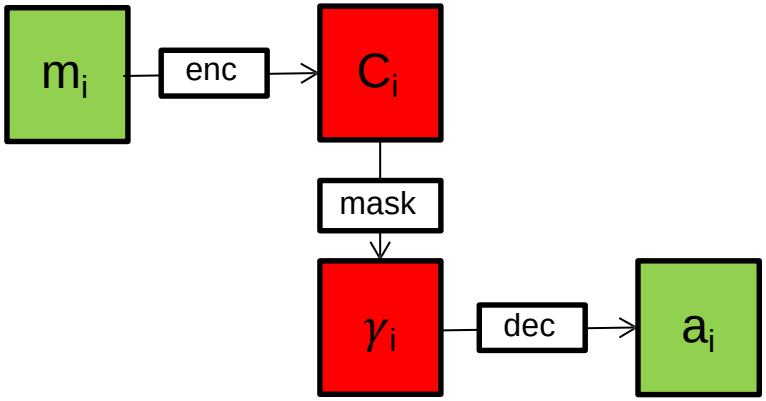
Linearly Homomorphic Authenticated Encryption with Public Verifiability (LAEPuV)

TECHNISCHE UNIVERSITÄT DARMSTADT

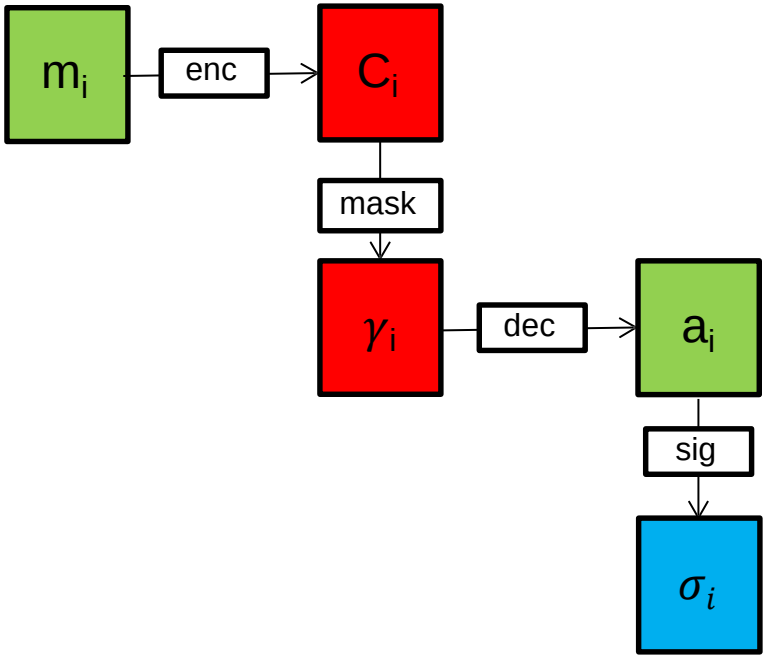


Linearly Homomorphic Authenticated Encryption with Public Verifiability (LAEPuV)

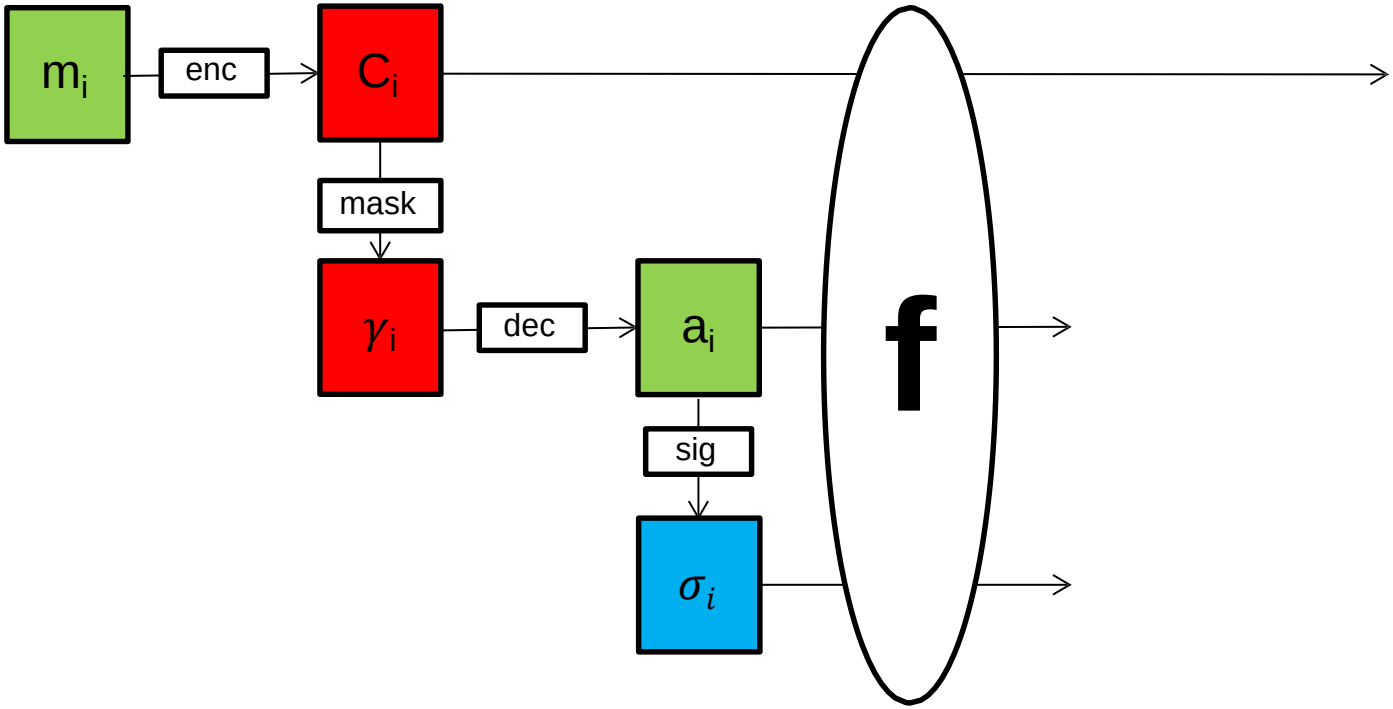
TECHNISCHE UNIVERSITÄT DARMSTADT



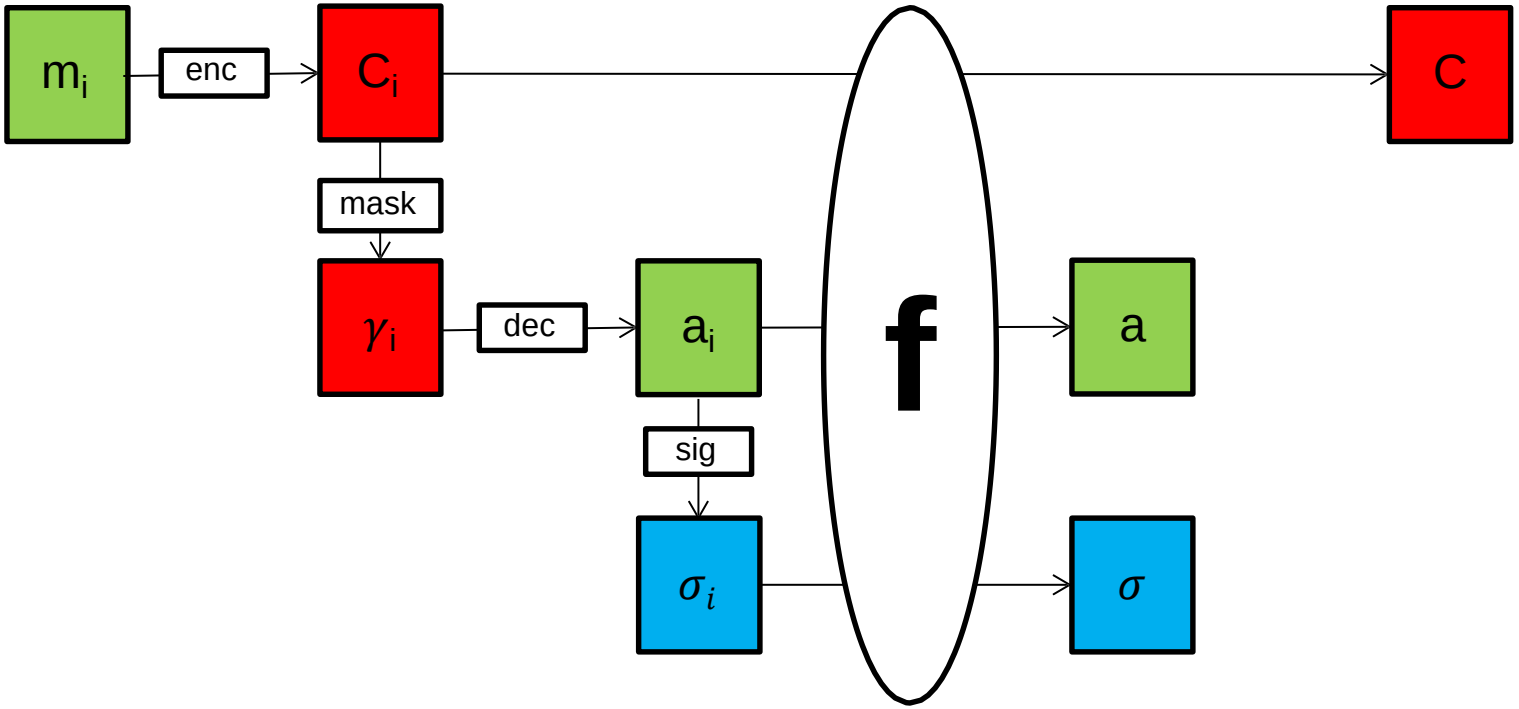
Linearly Homomorphic Authenticated Encryption with Public Verifiability (LAEPuV)



Linearly Homomorphic Authenticated Encryption with Public Verifiability (LAEPuV)

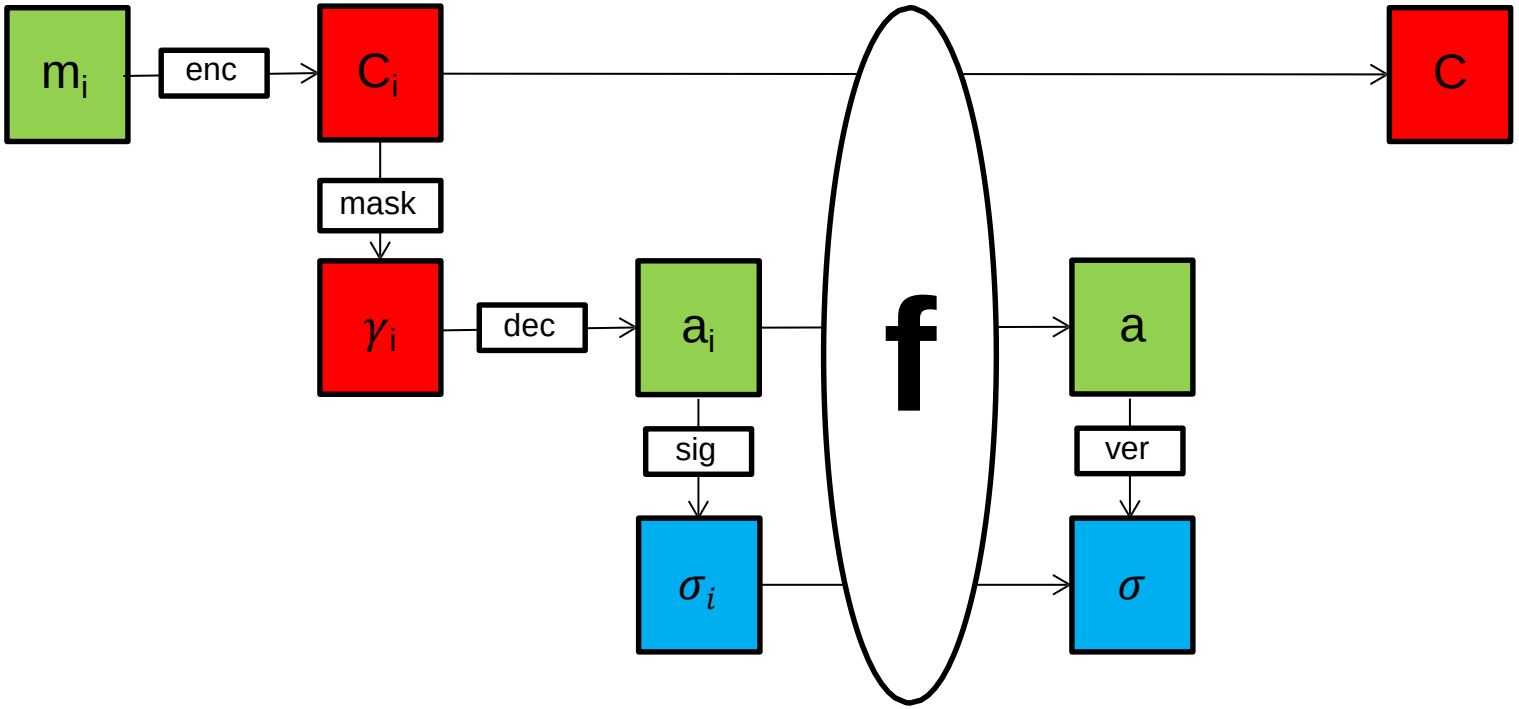


Linearly Homomorphic Authenticated Encryption with Public Verifiability (LAEPuV)

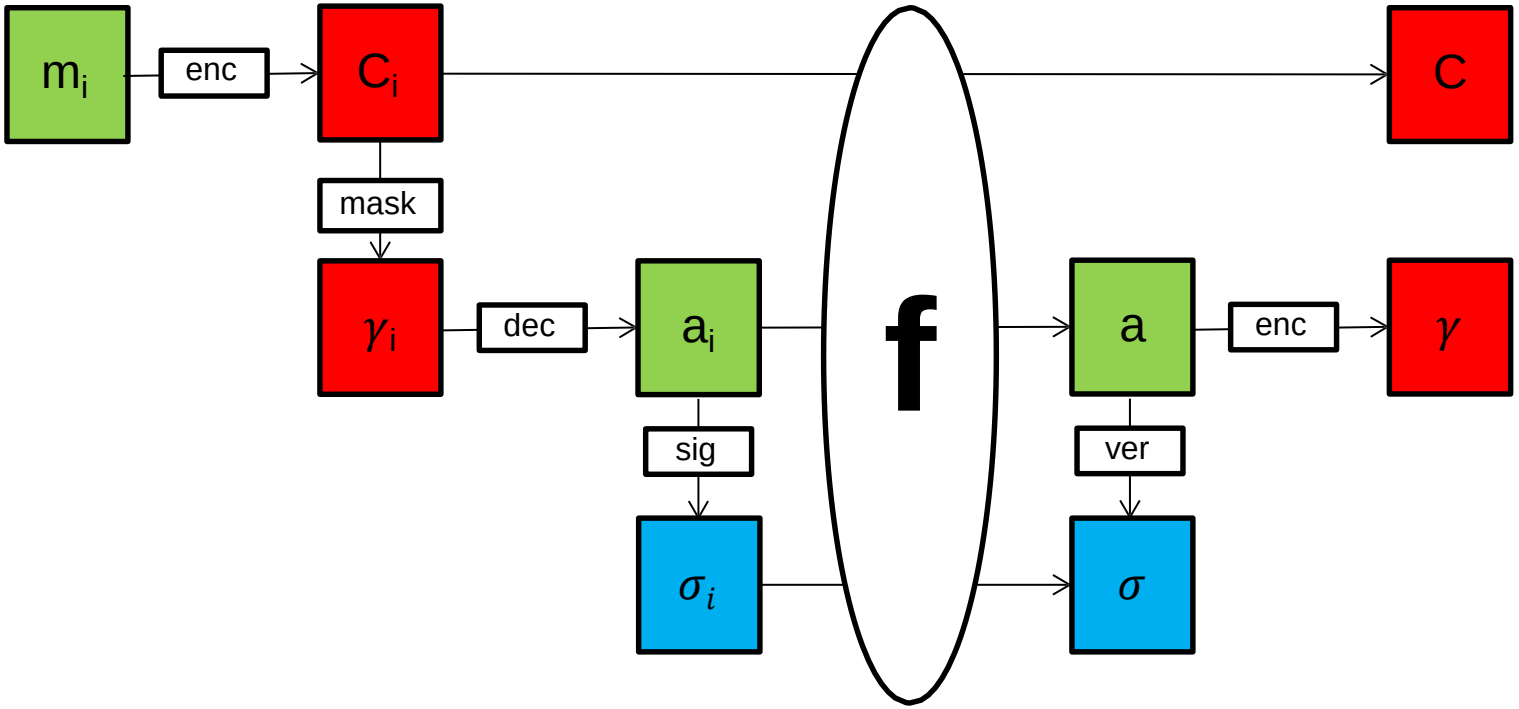


Linearly Homomorphic Authenticated Encryption with Public Verifiability (LAEPuV)

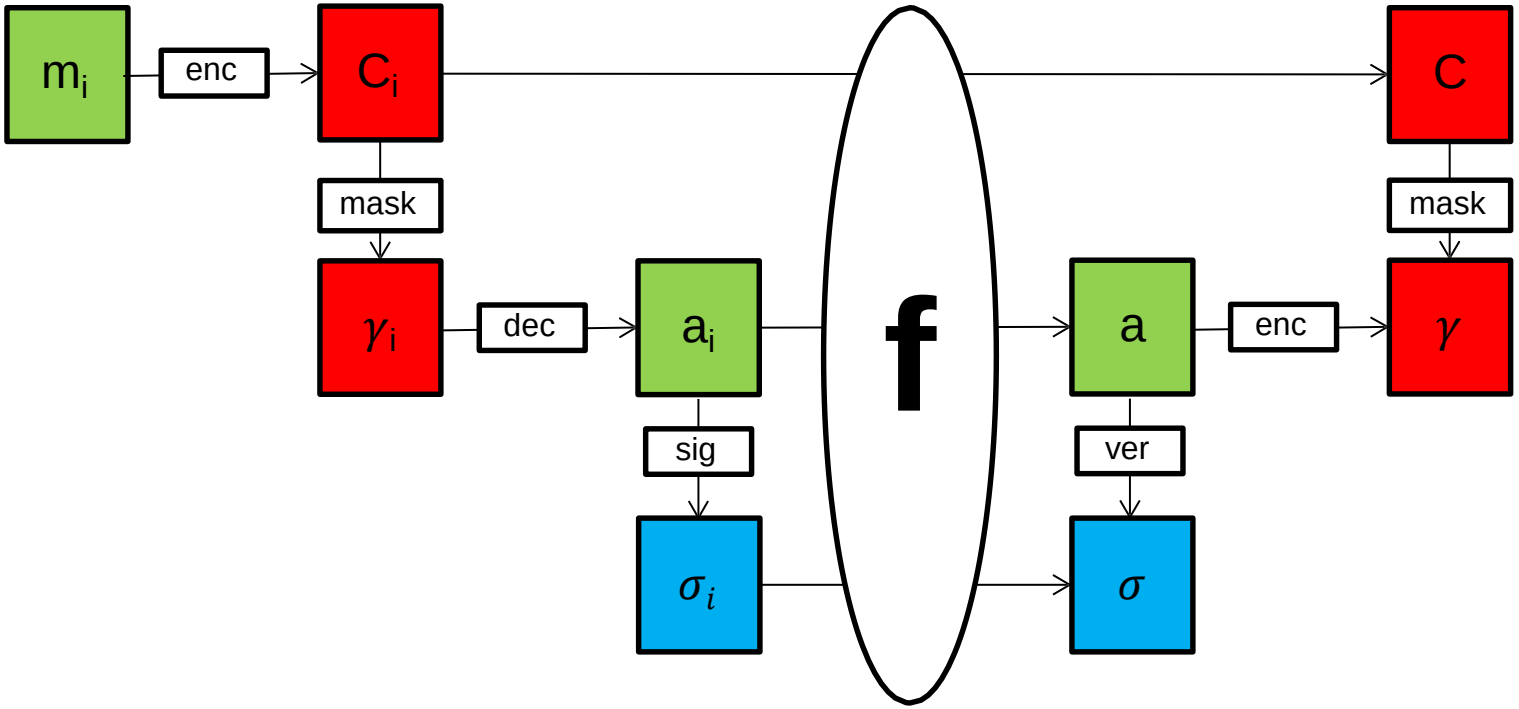
TECHNISCHE UNIVERSITÄT DARMSTADT



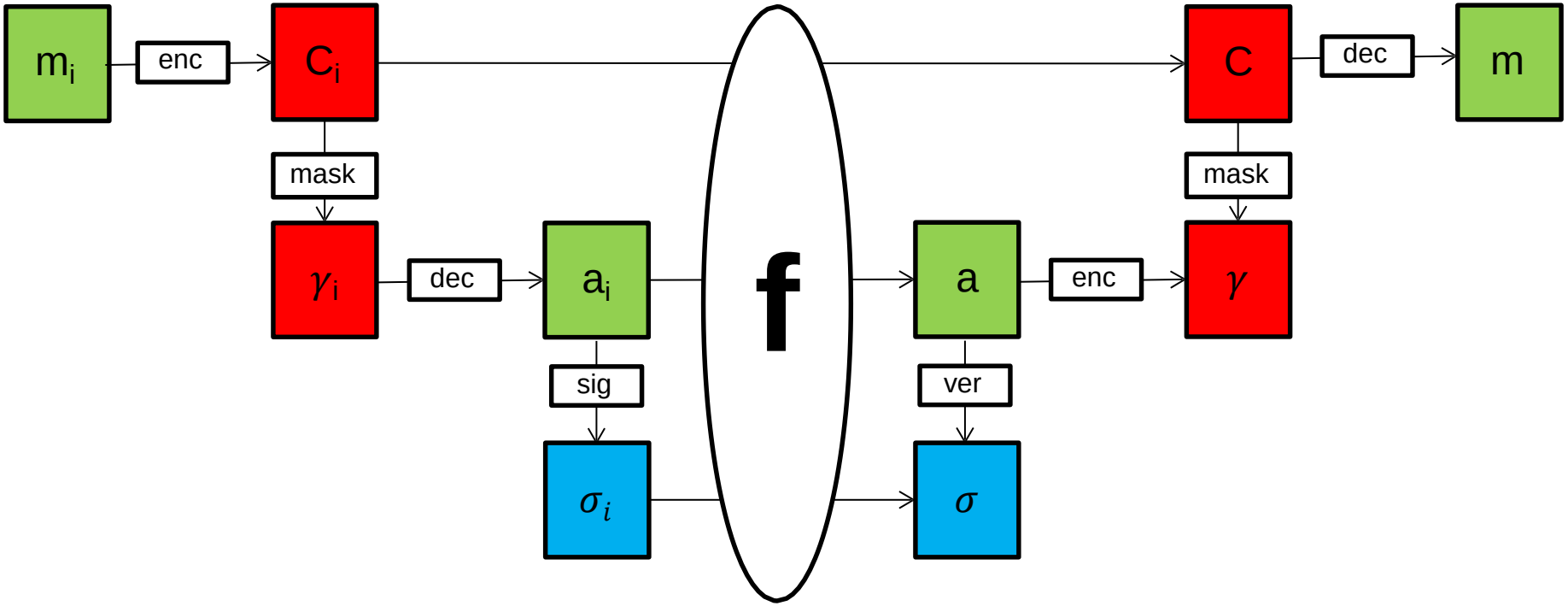
Linearly Homomorphic Authenticated Encryption with Public Verifiability (LAEPuV)



Linearly Homomorphic Authenticated Encryption with Public Verifiability (LAEPuV)



Linearly Homomorphic Authenticated Encryption with Public Verifiability (LAEPuV)



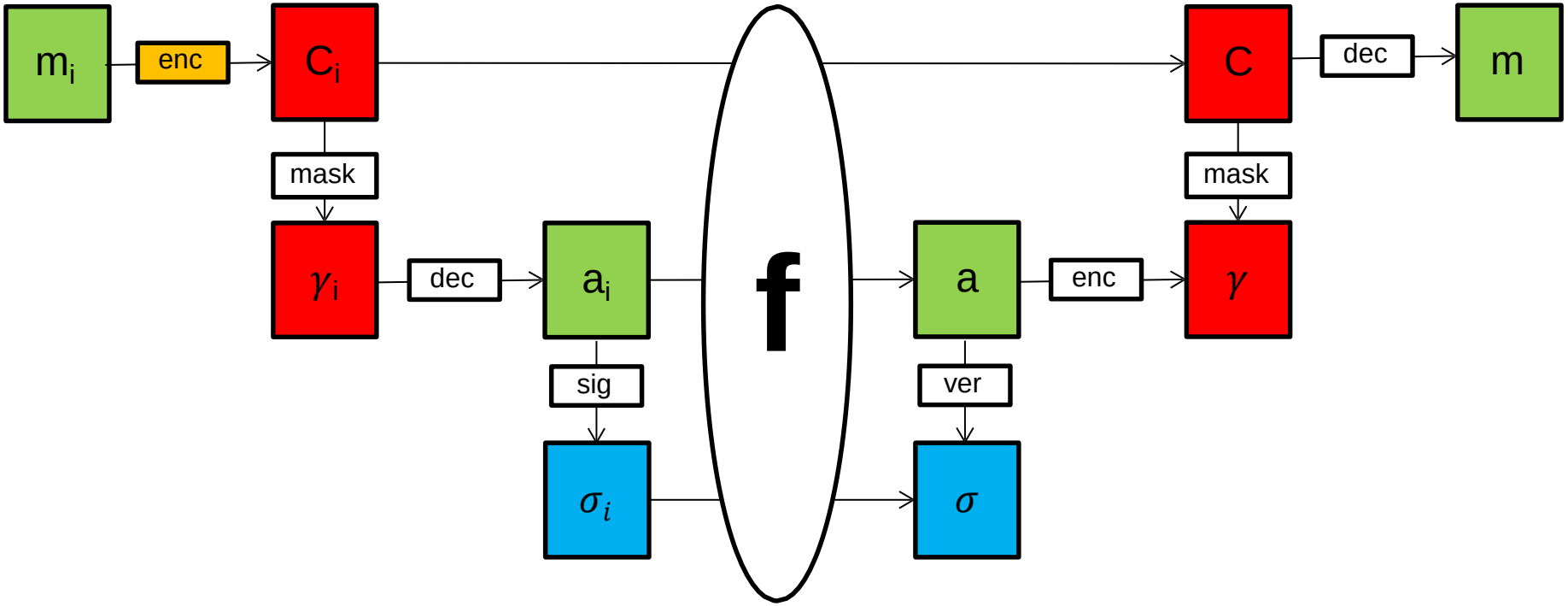
Paillier Encryption

$$C = g^m \cdot r^n \bmod n^2 \text{ with } n = pq$$

Linearly Homomorphic

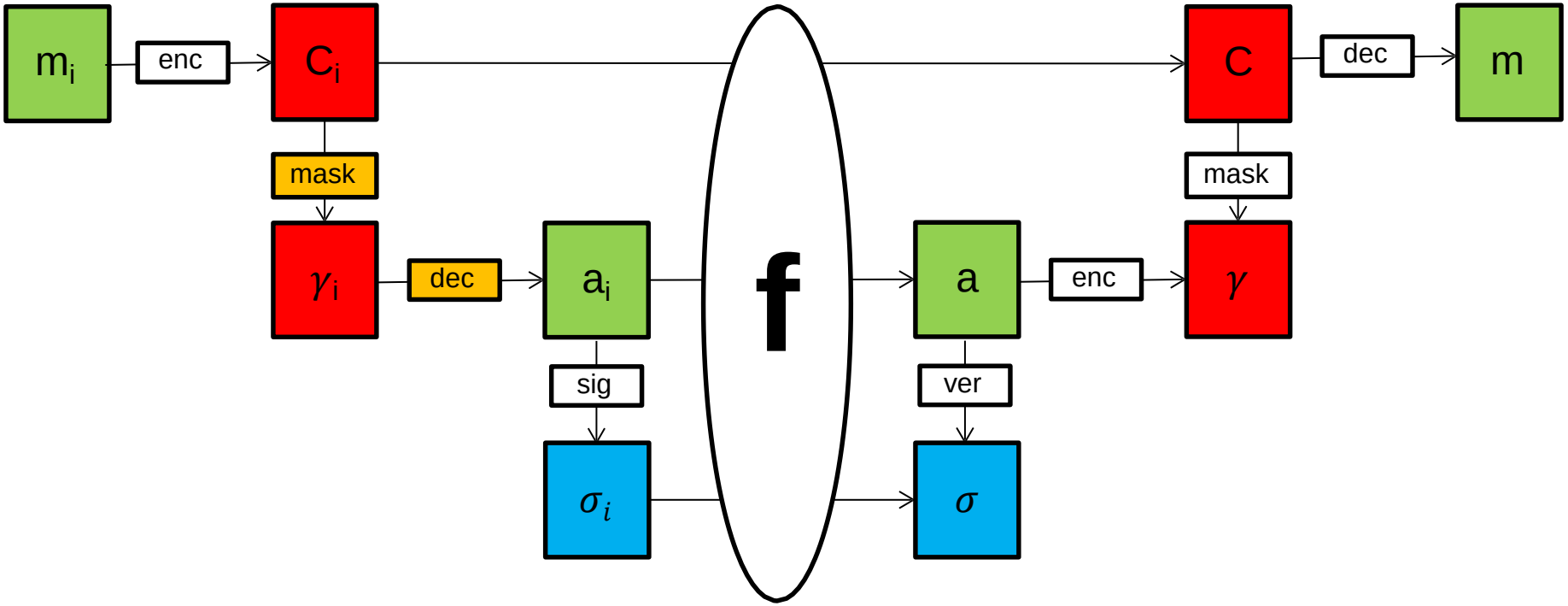
Public Key Encryption

LAEPuV



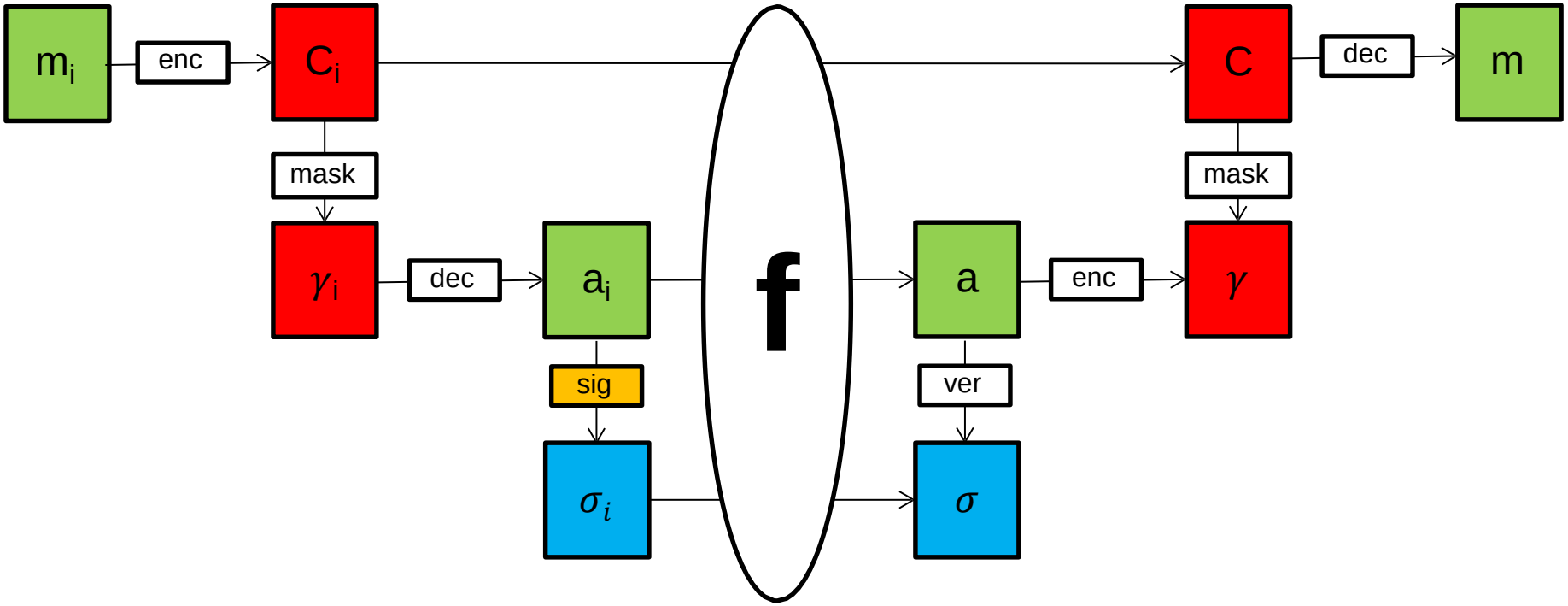
$$C_i = Enc(m_i)$$

LAEPuV



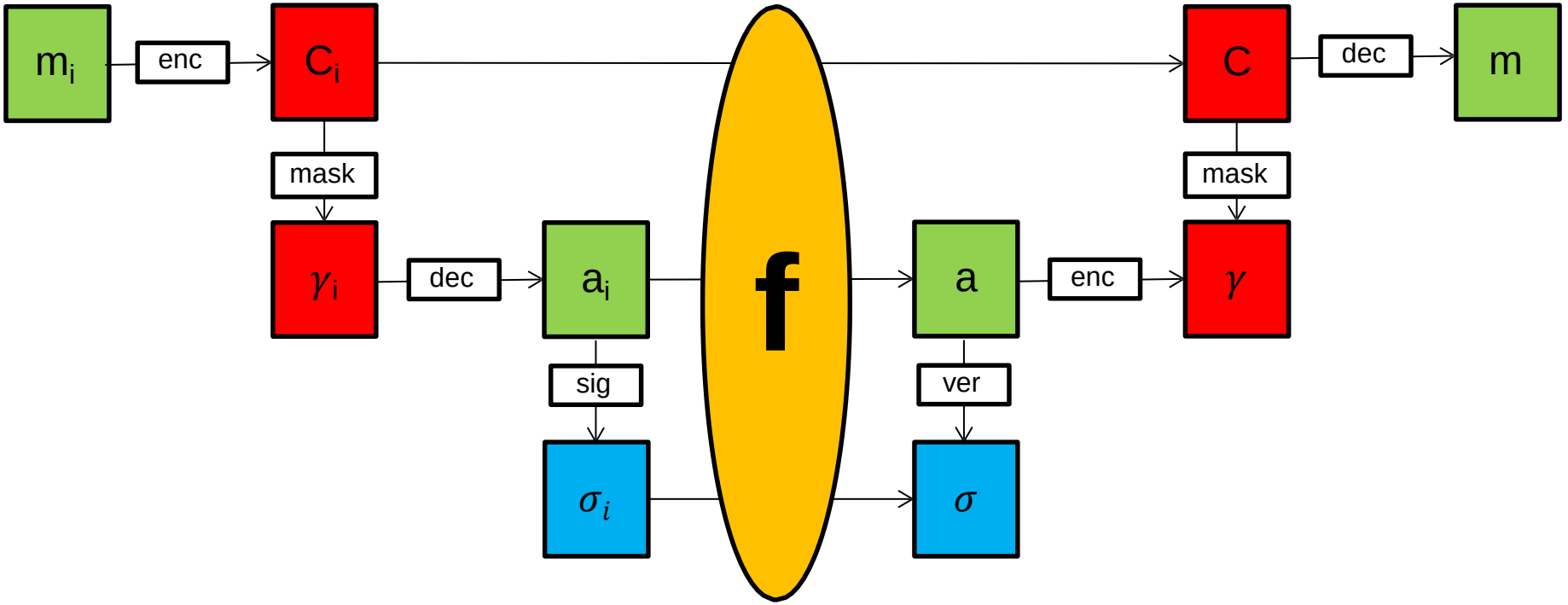
$$S \leftarrow H(\Delta|i) \qquad a_i \leftarrow Dec(C_i S_i)$$

LAEPuV



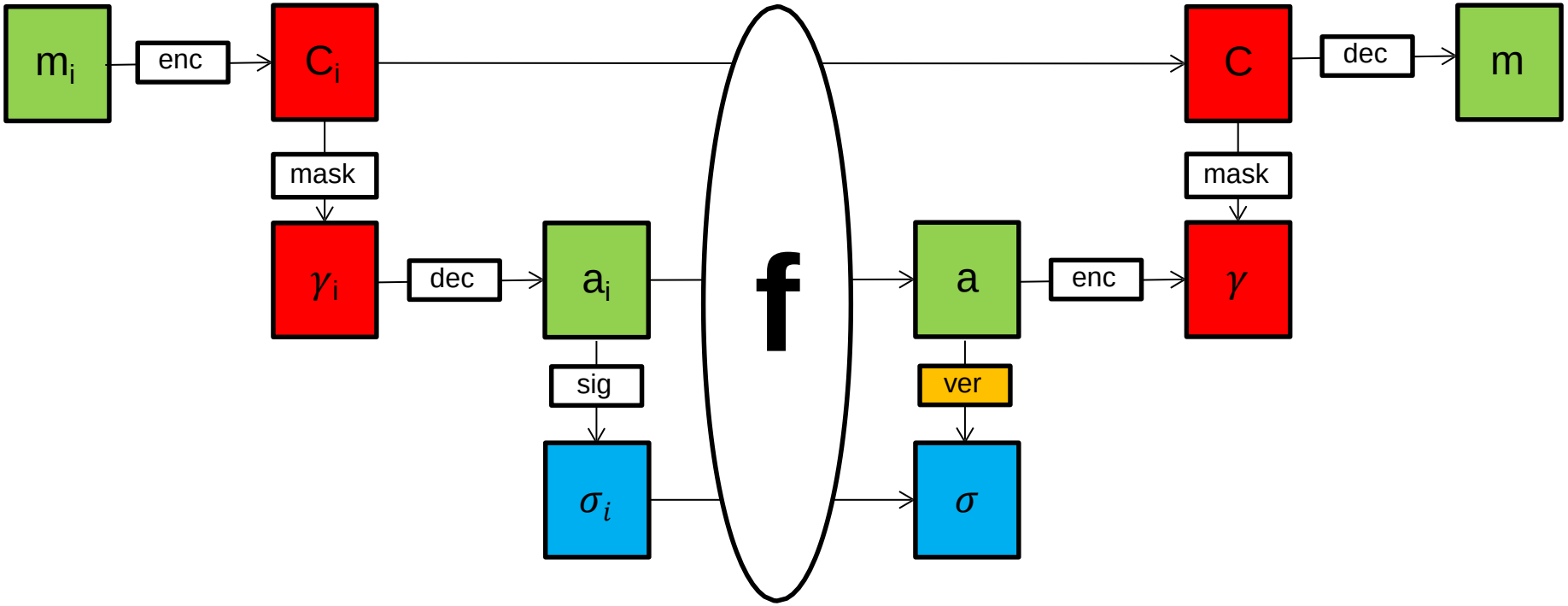
$$\Lambda \leftarrow (R_i \cdot h^{a_i})^z$$

LAEPuV



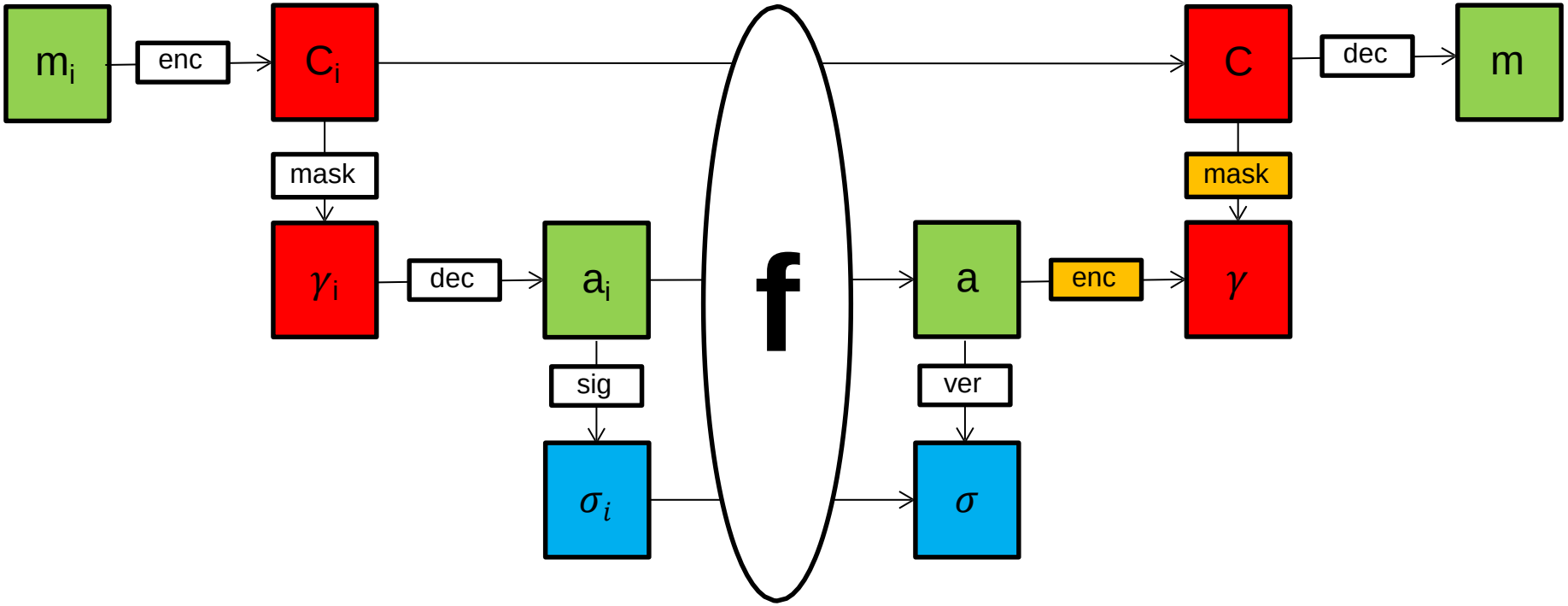
$$m \leftarrow \sum_{i=1}^{n} f_i m_i \qquad a \leftarrow \sum_{i=1}^{n} f_i a_i \qquad \Lambda \leftarrow \prod_{i=1}^{n} \Lambda_i^{f_i} \qquad C \leftarrow \prod_{i=1}^{n} C_i^{f_i}$$

LAEPuV



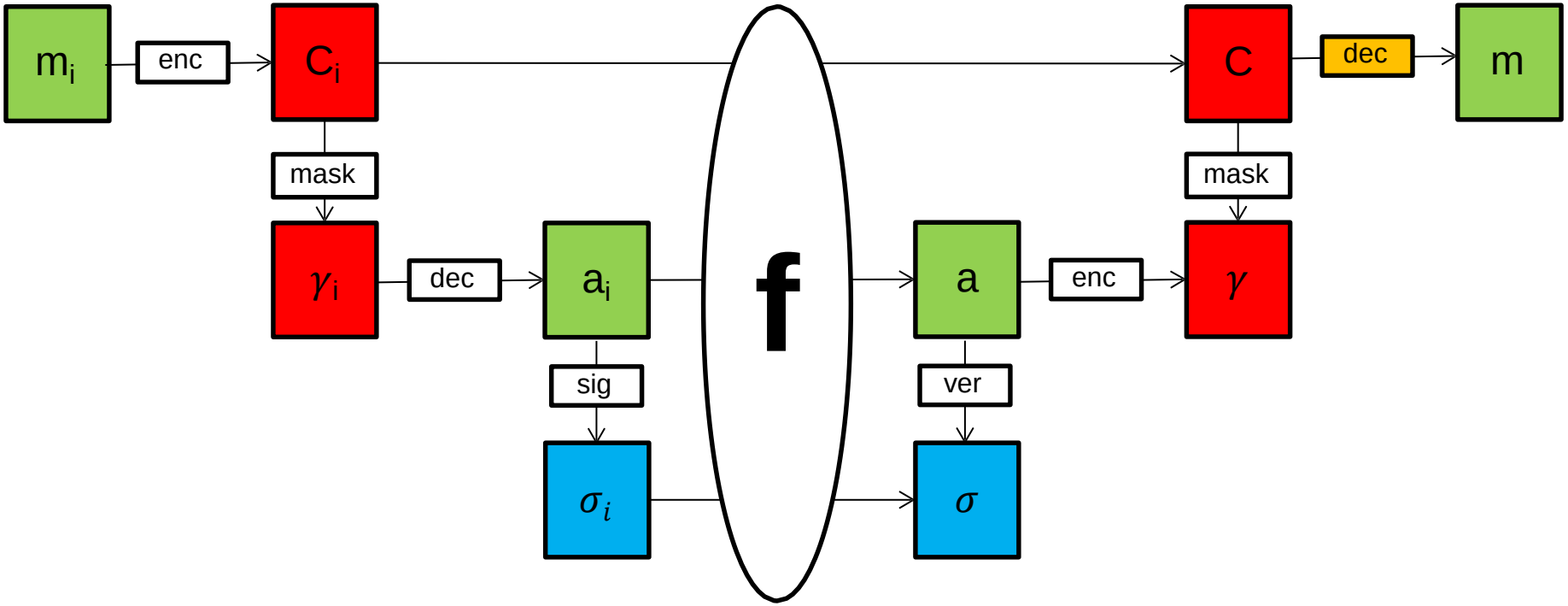
$$e(Rh^a, Z) = e(\Lambda, g)$$

LAEPuV



$$S \leftarrow \prod_{i=1}^{n} H(\Delta|i)^{f_i} \qquad Enc(a) = SC$$

LAEPuV



$$m = Dec(C)$$

Conclusion

TECHNISCHE UNIVERSITÄT DARMSTADT

Strongly secure under CDH in $\mathbb{G}$ in the standard model, if *F* is pseudorandom and *Sig* is an unforgeable (conventional) signature scheme.

Constant-time verification, after a one time preprocessing.

Information-theoretically context hiding (input privacy).

Can be extended to a LAEPuV scheme (output privacy).

Results

TECHNISCHE UNIVERSITÄT DARMSTADT

Thank you for your attention!

Questions?