Signature-in-Signature: the Last Line of Defence in Case of Signing Key Compromise

Przemysław Błaśkiewicz, Mirosław Kutyłowski, Marcin Słowik

Wrocław University of Science and Technology
Faculty of Computer Science & Telecommunication
Poland

September 21, 2021

Fairy tale of digital signatures

SSCD: legally binding electronic signatures/seals must be created by an SSCD (Secure Signature Creation Device)

Hope:
- SSCD designed so that it prevents key leakage,
- without the owner's consent SSCD will not create a signature.

Invalidation: signature/seal legally binding unless created after key revocation time.

When the worst happens...

security flaws: advances in cryptanalysis, cleptography and other trapdoors, implementation errors . . .

When the worst happens...

security flaws: advances in cryptanalysis, cleptography and other trapdoors, implementation errors . . .

latent attacks: With access to the signing key one can forge many documents *before* the signatory learns about that.

When the worst happens...

security flaws: advances in cryptanalysis, cleptography and other trapdoors, implementation errors ...

latent attacks: With access to the signing key one can forge many documents *before* the signatory learns about that.

useless revocation: (broken) electronic signature is a perfect tool for cybercriminality ...

When the worst happens...

security flaws: advances in cryptanalysis, cleptography and other trapdoors, implementation errors ...

latent attacks: With access to the signing key one can forge many documents *before* the signatory learns about that.

useless revocation: (broken) electronic signature is a perfect tool for cybercriminality ...

unless: ...we find a method to fish out forged signatures.

Our goal

Provide last line of defence

A signature system where one can tell between "legitimate" signatures and the ones created with a duplicate signing key.

Our goal

Provide last line of defence

A signature system where one can tell between "legitimate" signatures and the ones created with a duplicate signing key.

Limit the overhead

To leverage exisiting schemes and not rely on excessive external systems (blockchain, mediator schemes, . . .).

Our solution

Inner

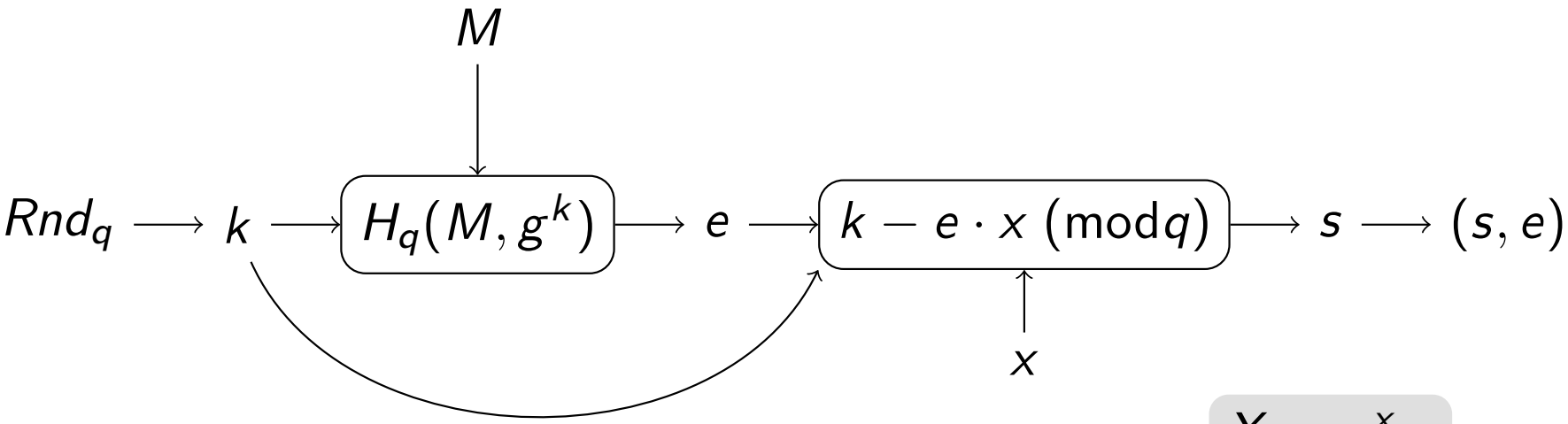
A standard signature contains an **inner signature** that is

cryptographically undetectable – even for a holder of the (leaked) signing key

verifiable in a standard way **once the signing key is revealed together with the inner public key**

Example: inner signature creation on top of Schnorr signature

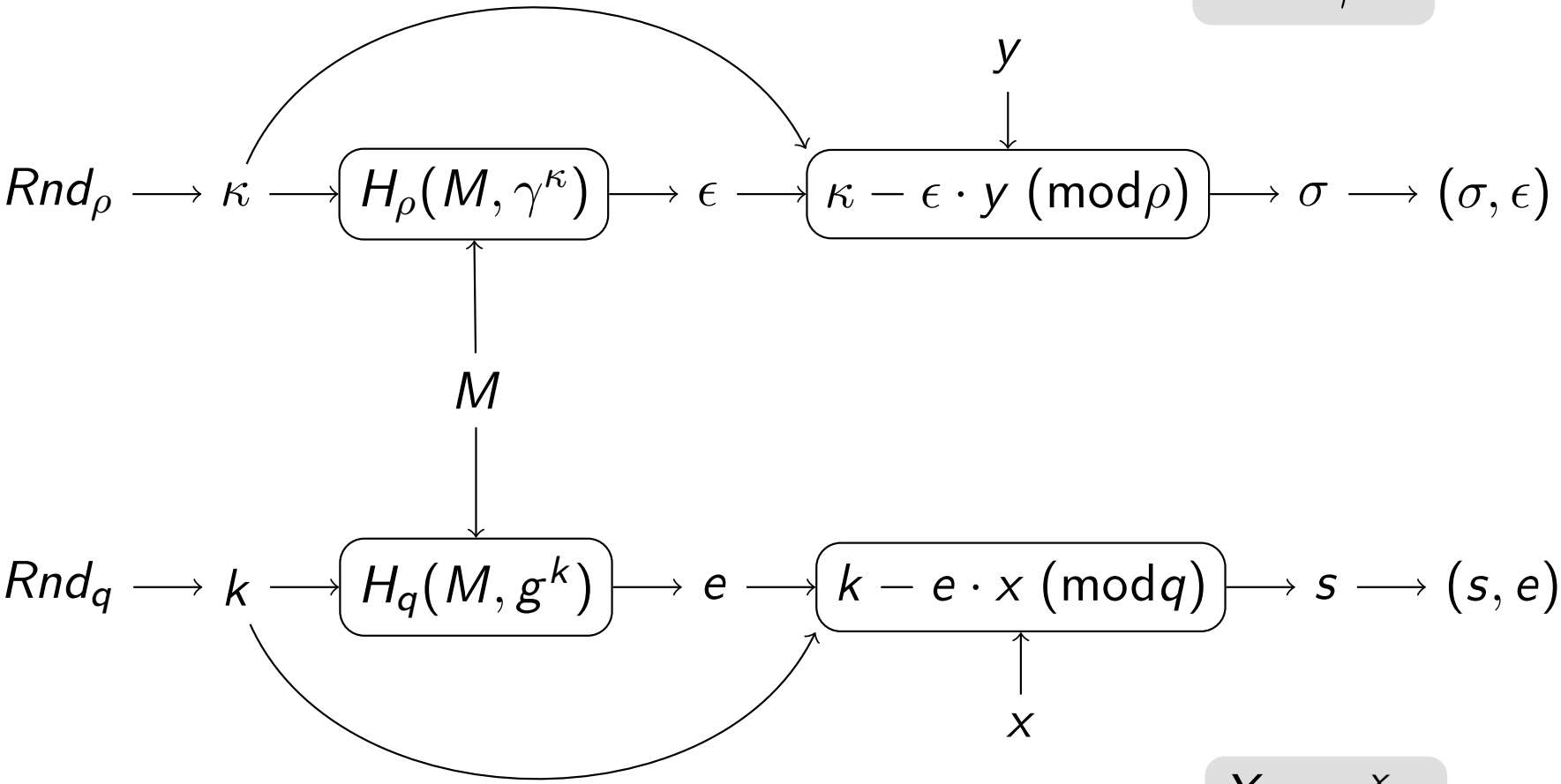
Example: inner signature creation on top of Schnorr signature



Example: inner signature creation on top of Schnorr signature

$\frac{\log q}{\log \rho} \geq 2$

$Y = \gamma^y$

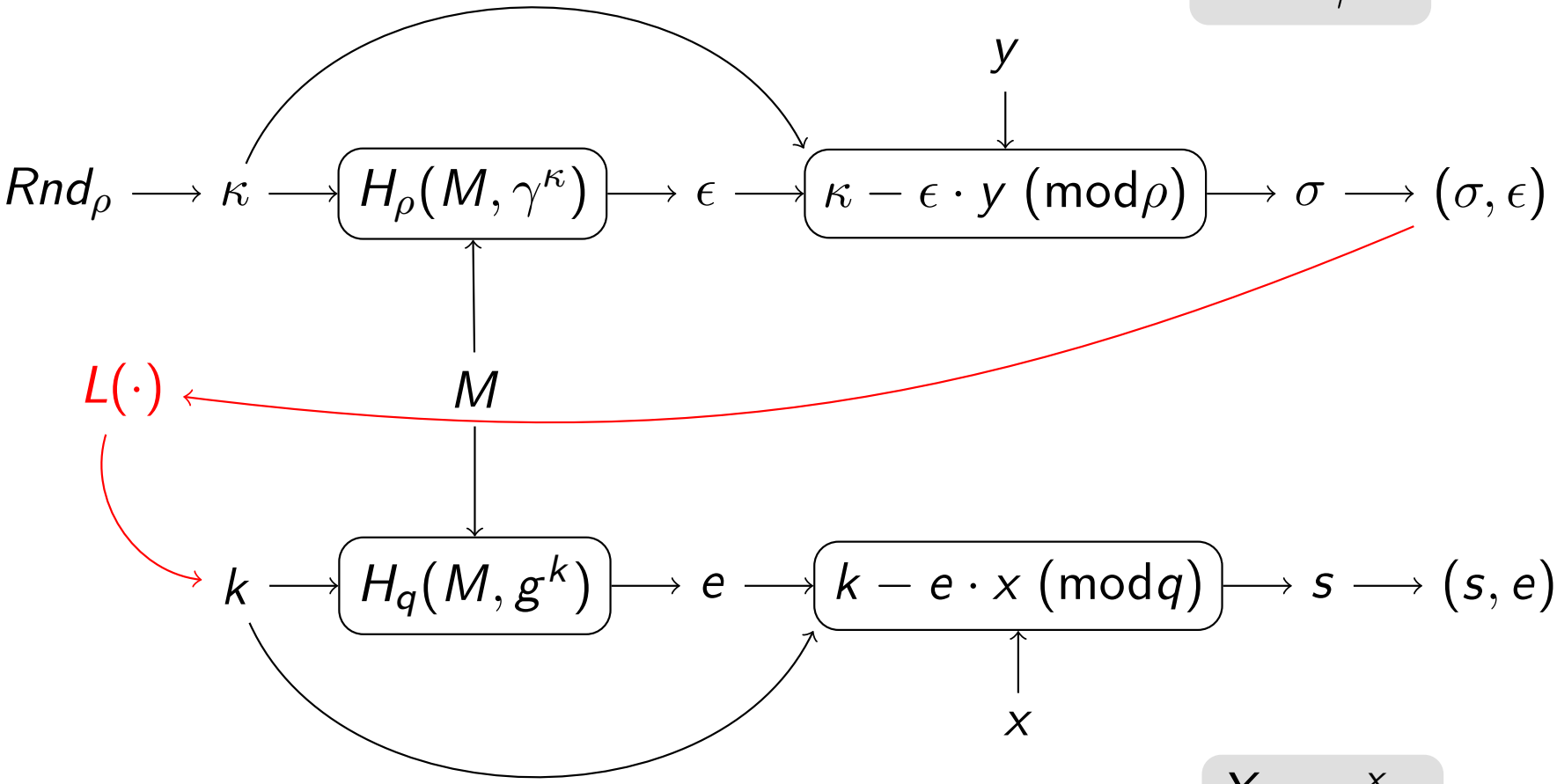


$X = g^x$

Example: inner signature creation on top of Schnorr signature

$\frac{\log q}{\log \rho} \geq 2$

$Y = \gamma^y$



$X = g^x$

Inner signature verification

given: inner public key Y, device's secret (compromised) key x are given.

procedure: for a signature (s, e) under M:

1. recompute ephemeral k as $k := s + e \cdot x \mod q$
2. retrieve inner signature(s): $(\sigma, \epsilon) := L^{-1}(k)$
3. (s, e) is valid if (σ, ϵ) verifies with Y for M.

Inner signature verification

given: inner public key Y, device's secret (compromised) key x are given.

procedure: for a signature (s, e) under M:

1. recompute ephemeral k as $k := s + e \cdot x \mod q$
2. retrieve inner signature(s): $(\sigma, \epsilon) := L^{-1}(k)$
3. (s, e) is valid if (σ, ϵ) verifies with Y for M.

Note: L is an invertible encoding function: given $L(\alpha, \beta)$ it should be possible to derive α, β.

Properties (1)

For the outer signature k is no longer random, but:

Schnorr pseudorandomness

Given message M, secret key x and a number s, it is infeasible to decide whether exists k fulfilling $s = k - x \cdot Hash(M, g^k)$ or s is random.

Properties (1)

For the outer signature k is no longer random, but:

Schnorr pseudorandomness

Given message M, secret key x and a number s, it is infeasible to decide whether exists k fulfilling $s = k - x \cdot Hash(M, g^k)$ or s is random.

Hence: hard to tell if the inner signature is hidden in a single instance of a Schnorr signature.

Properties (2)

Adversary managed to break X (?) and calculated x, so maybe he can do the same with Y?

- recover ephemeral values for outer signagures
- use L^{-1} to create candidate pairs of inner signatures
- break them...

Properties (2)

Adversary managed to break X (?) and calculated x, so maybe he can do the same with Y?

- recover ephemeral values for outer signagures
- use L^{-1} to create candidate pairs of inner signatures
- break them...

... however:

Secrecy of the public key

1. It is infeasible to derive the public key from Schnorr pair (σ, ϵ) or decide that no matching key exists.
2. It is infeasible to decide if two signatures $(s_0, e_0), (s_1, e_1)$ under M_0, M_1, respectively, correspond to the same public key.

Further notes

- Subsequent inner signatures can be linked

Further notes

- Subsequent inner signatures can be linked
- Implementation is ongoing, a few workable options presented in Appendix.

Thank you for your attention!