On the (im)possibility of ElGamal blind signatures

Liliya Akhmetzyanova, Evgeny Alekseev,
Alexandra Babueva and Stanislav Smyshlyaev

CryptoPro LLC

CRYPTOPRO

CTCrypt'2022

Outline

1. Motivation
2. Blind signatures
3. ElGamal blind signatures
4. Impossibility results

Outline

1. Motivation
2. Blind signatures
3. ElGamal blind signatures
4. Impossibility results

Motivation

November 2020

starting the research of perspective blind signature schemes

May 2022

1st draft of Methodic Recommendations «Blind signature schemes»

CRYPTOPRO

Motivation

Blind signature scheme based on ElGamal (preferably, GOST) equation is desirable:

- well-studied construction
- verification algorithm is the same as GOST signature verification algorithm
- standard basic mechanisms (elliptic curves, hash function)

Motivation

Known ElGamal blind signature schemes:

- Camenisch, J. L., Piveteau, J. M., Stadler, M. A., “Blind signatures based on the discrete logarithm problem”, 1994
- Rostovtsev, A. G., “Blind signature on elliptic curve for e-cash”, 2000
- Jena D., Panigrahy S. K., Acharya B., Jena S. K., “A Novel ECDLP-Based Blind Signature Scheme”, 2008
- Moldovyan, N. A. “Blind Signature Protocols from Digital Signature Standards”, 2011
- Shen, V. R., Chung, Y. F., Chen, T. S., Lin, Y. A., “A blind signature based on discrete logarithm problem”, 2011
- Gorbenko I., Yesina M., Ponomar V., “Anonymous electronic signature method”, 2016
- Tan D. N., Nam H. N., Van H. N., Thi, L. T., Hieu M. N., “New blind mutisignature schemes based on signature standards”, 2017
- Khater, M. M., Al-Ahwal, A., Selim, M. M., Zayed, H. H., “New Blind Signature Scheme Based on Modified ElGamal Signature for Secure Electronic Voting”, 2018
- Tan, D. N., Nam, H. N., Hieu, M. N., Van, H. N., “New Blind Muti-signature Schemes based on ECDLP”, 2018

Motivation

Known ElGamal blind signature schemes:

- Camenisch, J. L., Piveteau, J. M., Stadler, M. A., “Blind signatures based on the discrete logarithm problem”, 1994
- Rostovtsev, A. G., “Blind signature on elliptic curve for e-cash”, 2000
- Jena D., Panigrahy S. K., A arya B., Jena S. K Novel ECDLP-Based Blind Signature Scheme”, 2008
- Moldovyan, N. A. “Blind Signa e Pro ols from Digital Signature Standards”, 2011
- Shen, V. R., Chung, Y. F., Chen, T. , Y. A., “A blind signature based on discrete logarithm problem”, 2011
- Gorbenko I., Yesina M., Pon r V., “Ano ous electronic signature method”, 2016
- Tan D. N., Nam H. N., Van H. N., Thi, L. T., Hieu M. N., “New blind mutisignature schemes based on signature standards”, 2017
- Khater, M. M., Al-Ahwal, A., Selim, M. M., Zayed, H. H., “New Blind Signature Scheme Based on Modified ElGamal Signature for Secure Electronic Voting”, 2018
- Tan, D. N., Nam, H. N., Hieu, M. N., Van, H. N., “New Blind Muti-signature Schemes based on ECDLP”, 2018

Problem

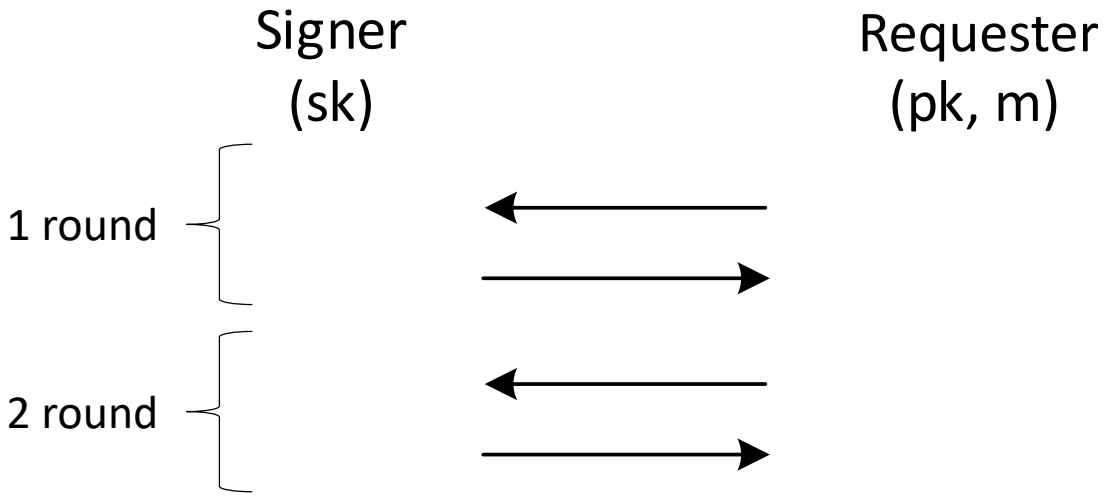
Is it possible to construct secure blind signature scheme based on ElGamal signature equation?

Outline

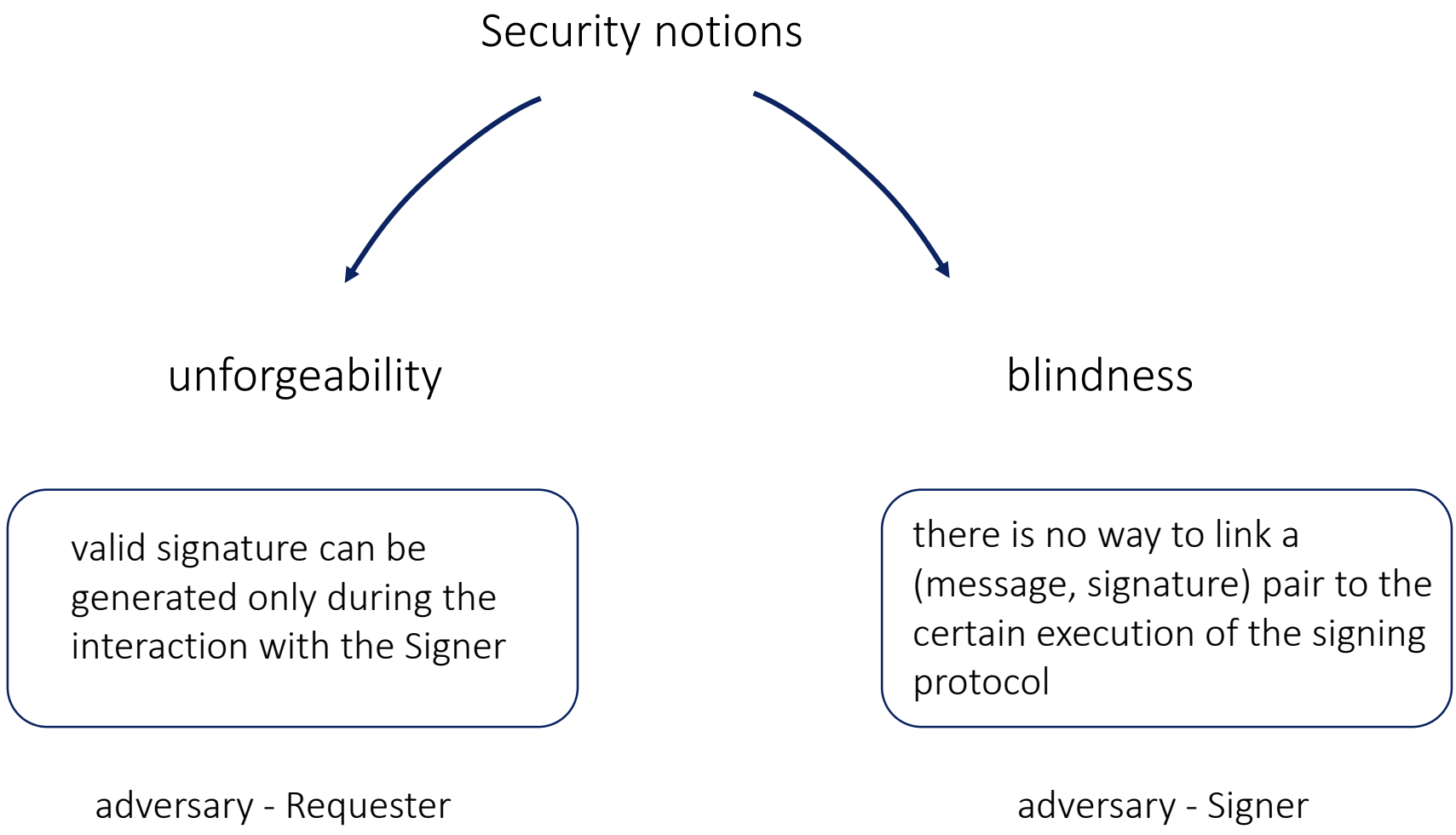
1. Motivation
2. Blind signatures
3. ElGamal blind signatures
4. Impossibility results

Blind signatures

- $(sk, pk) \leftarrow KeyGen(\,)$: key generation algorithm
- $(b, \sigma) \leftarrow \langle Signer(sk), Requester(pk, m)\rangle$: interactive signing protocol that is run between a Signer and a Requester
- $b \leftarrow Verify(pk, m, \sigma)$: verification algorithm



Blind signatures



Blind signatures

unforgeability

valid signature can be generated only during the interaction with the Signer

adversary - Requester

attacks with parallel sessions!

blindness

there is no way to link a (message, signature) pair to the certain execution of the signing protocol

adversary - Signer

Outline

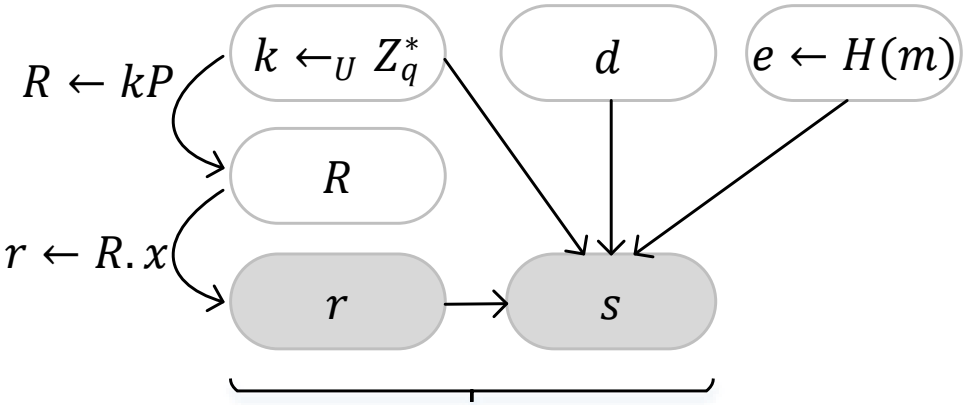
1. Motivation
2. Blind signatures
3. ElGamal blind signatures
4. Impossibility results

ElGamal signature

$KeyGen()$:

$d \leftarrow_U \mathbb{Z}_q^*$
$Q \leftarrow dP$
return (d, Q)

$Sign(d, m)$:

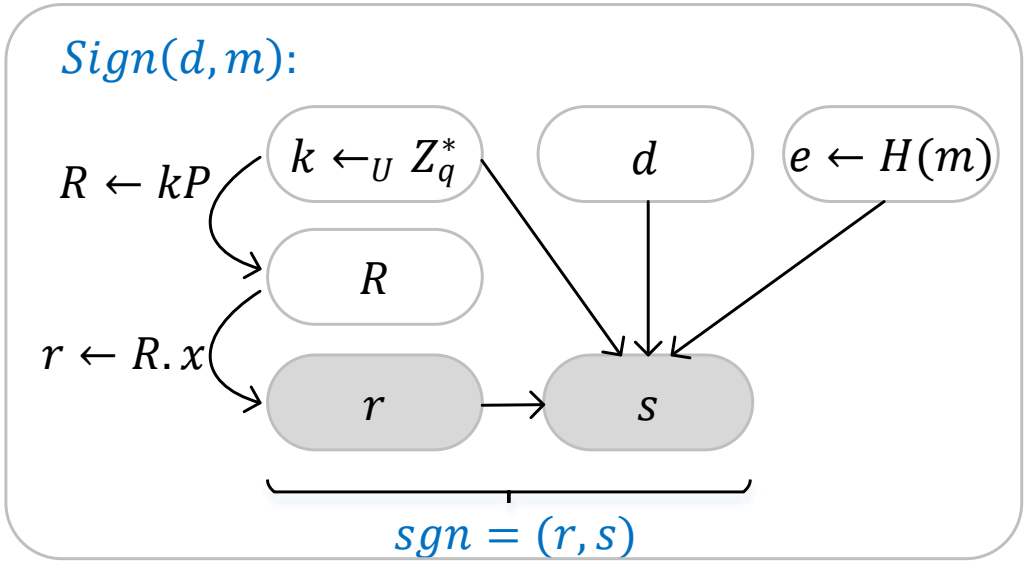


General ElGamal signature equation:

$$\underbrace{G_d(r,e,s) \cdot d + G_k(r,e,s) \cdot k + G_0(r,e,s)}_{EG(d,k,r,e,s)} = 0$$

q – order of a cyclic subgroup of the elliptic curve points group
P – the generator of the subgroup of order q

ElGamal signature



General ElGamal signature equation:

$$\underbrace{G_d(r,e,s)\cdot d + G_k(r,e,s)\cdot k + G_0(r,e,s)}_{EG(d,k,r,e,s)} = 0$$

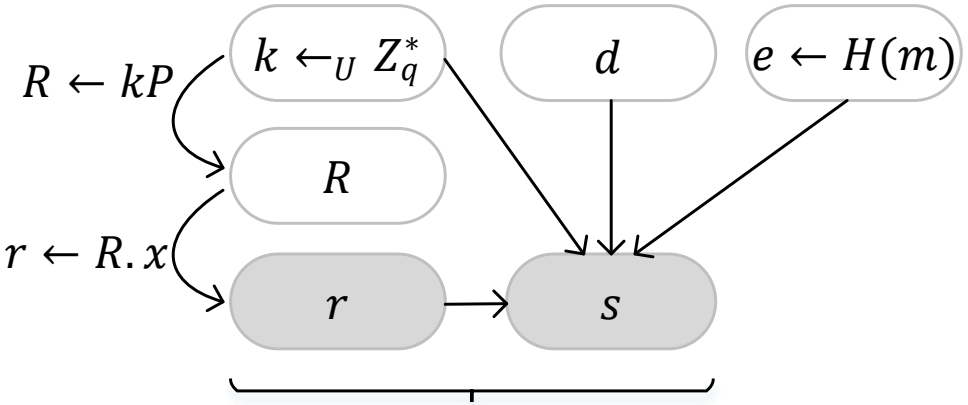
q – order of a cyclic subgroup of the elliptic curve points group
P – the generator of the subgroup of order q

ElGamal signature

$KeyGen()$:

$d \leftarrow_U \mathbb{Z}_q^*$
$Q \leftarrow dP$
return (d, Q)

$Sign(d, m)$:



Harn, Xu "Design of generalised ElGamal type digital signature schemes based on discrete logarithm", 1994

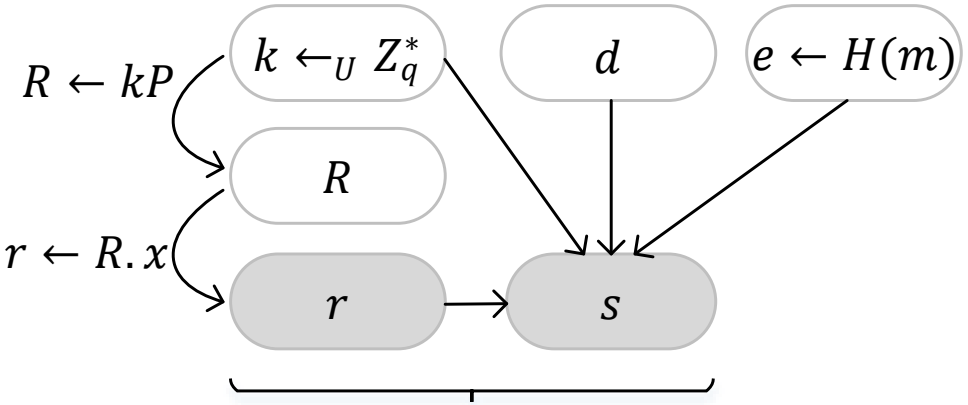
1 :	$ed = rk + s$	7 :	$red = k + s$	13 :	$(r + e)d = k + s$
2 :	$ed = sk + r$	8 :	$d = rek + s$	14 :	$d = (r + e)k + s$
3 :	$rd = ek + s$	9 :	$sd = k + re$	15 :	$sd = k + (r + e)$
4 :	$rd = sk + e$	10 :	$d = sk + re$	16 :	$d = sk + (r + e)$
5 :	$sd = rk + e$	11 :	$red = sk + 1$	17 :	$(r + e)d = sk + 1$
6 :	$sd = ek + r$	12 :	$sd = rek + 1$	18 :	$sd = (r + e)k + 1$

ElGamal signature

$KeyGen()$:

$d \leftarrow_U \mathbb{Z}_q^*$
$Q \leftarrow dP$
return (d, Q)

$Sign(d, m)$:



Harn, Xu "Design of generalised ElGamal type digital signature schemes based on discrete logarithm", 1994

GOST

1 :	$ed = rk + s$	7 :	$red = k + s$	13 :	$(r + e)d = k + s$
2 :	$ed = sk + r$	8 :	$d = rek + s$	14 :	$d = (r + e)k + s$
3 :	$rd = ek + s$	9 :	$sd = k + re$	15 :	$sd = k + (r + e)$
4 :	$rd = sk + e$	10 :	$d = sk + re$	16 :	$d = sk + (r + e)$
5 :	$sd = rk + e$	11 :	$red = sk + 1$	17 :	$(r + e)d = sk + 1$
6 :	$sd = ek + r$	12 :	$sd = rek + 1$	18 :	$sd = (r + e)k + 1$

Our contribution 1: GenEG-BS scheme

Signer d		Requester Q, m
$k \leftarrow_U \mathbb{Z}_q^*$ $R \leftarrow kP$	R →	
	e ←	…
$r \leftarrow R.x \bmod q$		
find s: $EG(d, k, r, e, s) = 0$	s →	
		…
		return (r', s'): $Verify(Q, m, (r', s')) = 1$

ElGamal signature generation algorithm for e

Our contribution 1: GenEG-BS scheme

Signer d		Requester Q, m
$k \leftarrow_U \mathbb{Z}_q^*$ $R \leftarrow kP$	R →	
	← e	…
$r \leftarrow R.x \bmod q$		
find s: $EG(d, k, r, e, s) = 0$	s →	
		…
		return (r', s'): $Verify\big(Q, m, (r', s')\big) = 1$

ElGamal signature generation algorithm for e

All known schemes satisfy this construction

Outline

1. Motivation
2. Blind signatures
3. ElGamal blind signatures
4. Impossibility results

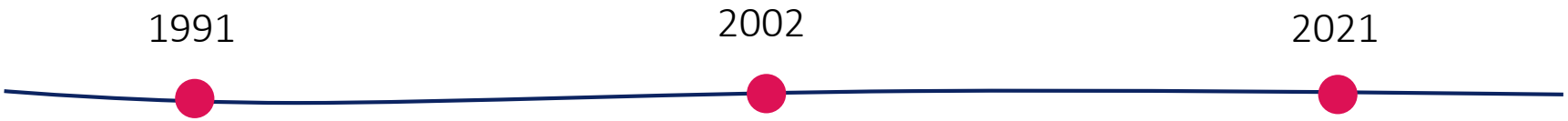
Our contribution 2: security results

We obtain several negative results on the possibility of constructing secure GenEG-BS scheme.

As the consequence we:

- show that all known GenEG-BS schemes are not secure
- provide the necessary conditions for GenEG-BS schemes that can potentially be secure

ROS problem



Schnorr
«Security of blind discrete log signatures against interactive attacks»

introduce ROS problem

Wagner
«A generalized birthday problem»

subexponential algorithm

BLLOR
«On the (in)security of ROS»

polynomial algorithm for $\ell \geq \lceil \log q \rceil$

attack on unforgeability of the Schnorr blind signature when the number of parallel sessions $\ell \geq \lceil \log q \rceil$

GenEG-BS schemes

schemes of Type I

?

generic ROS-style attack on unforgeability when the number of parallel sessions $\ell \geq \lceil \log q \rceil$

ROS-style attack: necessary condition

Signature equation:

$$G_d(r,e,s)\cdot d + G_k(r,e,s)\cdot k + G_0(r,e,s) = 0$$

Condition 1: at least one of the functions $\frac{G_d(r,e,s)}{G_k(r,e,s)}$ or $\frac{G_0(r,e,s)}{G_k(r,e,s)}$ does not significantly depend on s.

does not depend on s — depends on s

Signature equation:

$$k + Y_1(r,e)\cdot G_1(d) + Y_2(r,e,s)\cdot G_2(d) = 0,$$

where $G_1(d), G_2(d) \in \{1, d\}, G_1(d)\cdot G_2(d) = d$.

ROS-style attack: overview

Signer

(breaks one-more unforgeability)

$R_1, R_2, \ldots, R_\ell$ →

← $e_1, e_2, \ldots, e_\ell$

$s_1, s_2, \ldots, s_\ell$ →

ℓ successful sessions

(ℓ+1) valid signatures

$(m_1, \sigma_1), \ldots, (m_{\ell+1}, \sigma_{\ell+1})$

ROS-style attack: overview

Signer

Combining ℓ sessions linearly:

$R_1, R_2, \dots, R_\ell$ →

- Pick $\alpha_1, \alpha_2, \dots, \alpha_\ell, e = H(m)$:

$$R \leftarrow \alpha_1 R_1 + \alpha_2 R_2 + \cdots + \alpha_\ell R_\ell$$
$$r \leftarrow R.x \bmod q, \quad r_i \leftarrow R_i.x \bmod q$$

- Pick $e_i = H(m_i)$ such that:

← $e_1, e_2, \dots, e_\ell$

$$Y_1(r, e) = \sum_i \alpha_i Y_1(r_i, e_i)$$

$s_1, s_2, \dots, s_\ell$ →

- Define s from:

$$Y_2(r, e, s) = \sum_i \alpha_i Y_2(r_i, e_i, s_i)$$

Extend to an attack:

$m, (r, s)$ – (ℓ+1)-th valid pair

ROS-style attack: overview

Signer

ROS problem solvable when $\ell > \log q$

Combining ℓ sessions linearly:

$R_1, R_2, \dots, R_\ell$ →

- Pick $\alpha_1, \alpha_2, \dots, \alpha_\ell, e = H(m)$:

$$R \leftarrow \alpha_1 R_1 + \alpha_2 R_2 + \cdots + \alpha_\ell R_\ell$$
$$r \leftarrow R.x \bmod q, \quad r_i \leftarrow R_i.x \bmod q$$

- Pick $e_i = H(m_i)$ such that:

$$Y_1(r, e) = \sum_i \alpha_i Y_1(r_i, e_i)$$

← $e_1, e_2, \dots, e_\ell$

$s_1, s_2, \dots, s_\ell$ →

- Define s from:

$$Y_2(r, e, s) = \sum_i \alpha_i Y_2(r_i, e_i, s_i)$$

Extend to an attack:

$m, (r, s)$ – (ℓ+1)-th valid pair

ROS-style attack: overview

Signer

$R_1, R_2, \dots, R_\ell$ →

Combining ℓ sessions linearly:

- Pick $\alpha_1, \alpha_2, \dots, \alpha_\ell, e = H(m)$:

$$R \leftarrow \alpha_1 R_1 + \alpha_2 R_2 + \cdots + \alpha_\ell R_\ell$$
$$r \leftarrow R.x \bmod q, \quad r_i \leftarrow R_i.x \bmod q$$

- Pick $e_i = H(m_i)$ such that:

$$Y_1(r, e) = \sum_i \alpha_i Y_1(r_i, e_i)$$

← $e_1, e_2, \dots, e_\ell$

$s_1, s_2, \dots, s_\ell$ →

- Define s from:

$$Y_2(r, e, s) = \sum_i \alpha_i Y_2(r_i, e_i, s_i)$$

Condition 1 allows to pick $e_1, \dots, e_\ell$ before receiving $s_1, \dots, s_\ell$

Extend to an attack:

$m, (r, s)$ – (ℓ+1)-th valid pair

GenEG-BS schemes

schemes of Type I

?

generic ROS-style attack on unforgeability when the number of parallel sessions $\ell \geq \lceil \log q \rceil$

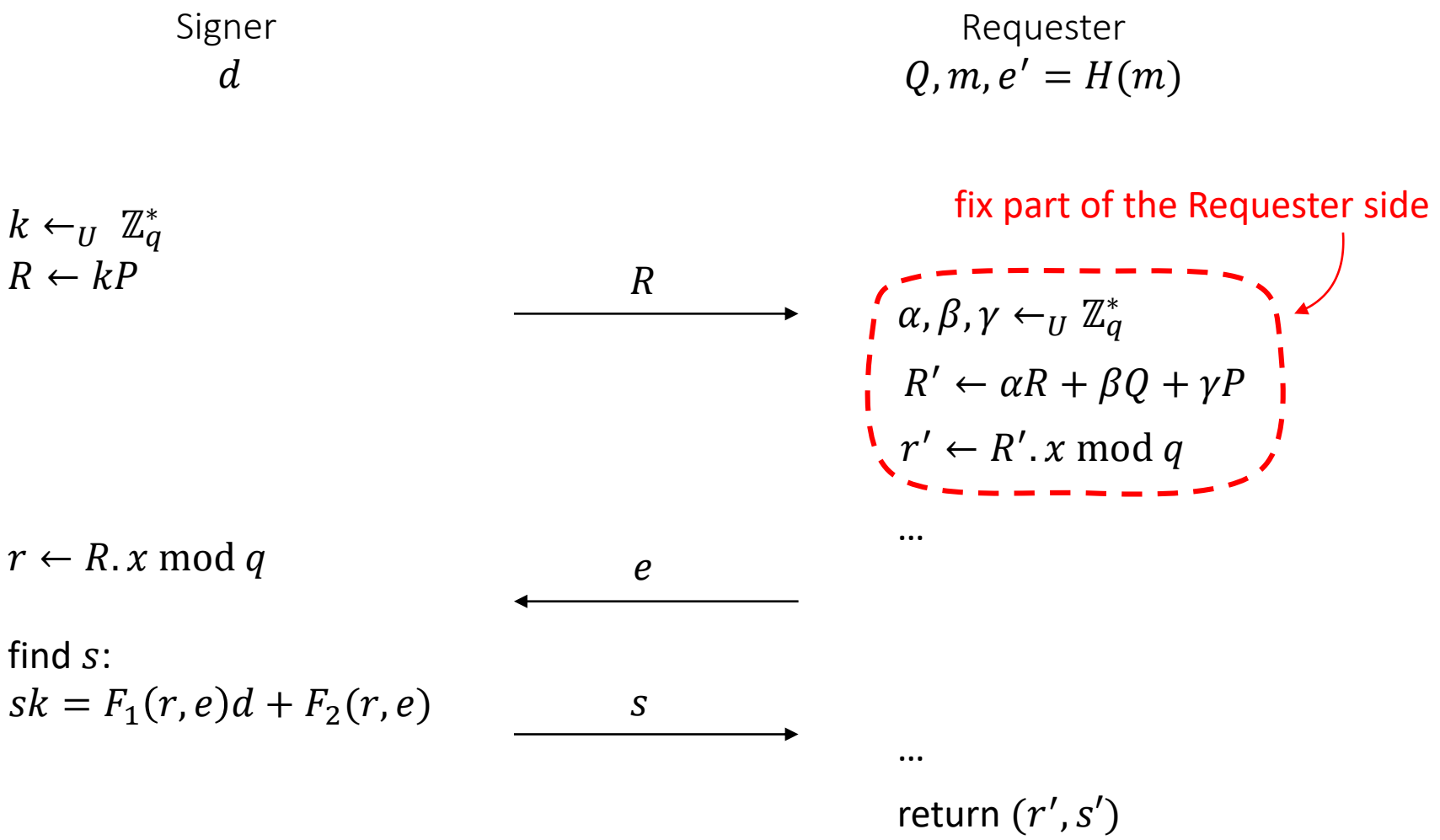
Schemes that do not satisfy Condition 1

1 :	$ed = rk + s$	7 :	$red = k + s$	13 :	$(r + e)d = k + s$
2 :	$ed = sk + r$	8 :	$d = rek + s$	14 :	$d = (r + e)k + s$
3 :	$rd = ek + s$	9 :	$sd = k + re$	15 :	$sd = k + (r + e)$
4 :	$rd = sk + e$	10 :	$d = sk + re$	16 :	$d = sk + (r + e)$
5 :	$sd = rk + e$	11 :	$red = sk + 1$	17 :	$(r + e)d = sk + 1$
6 :	$sd = ek + r$	12 :	$sd = rek + 1$	18 :	$sd = (r + e)k + 1$

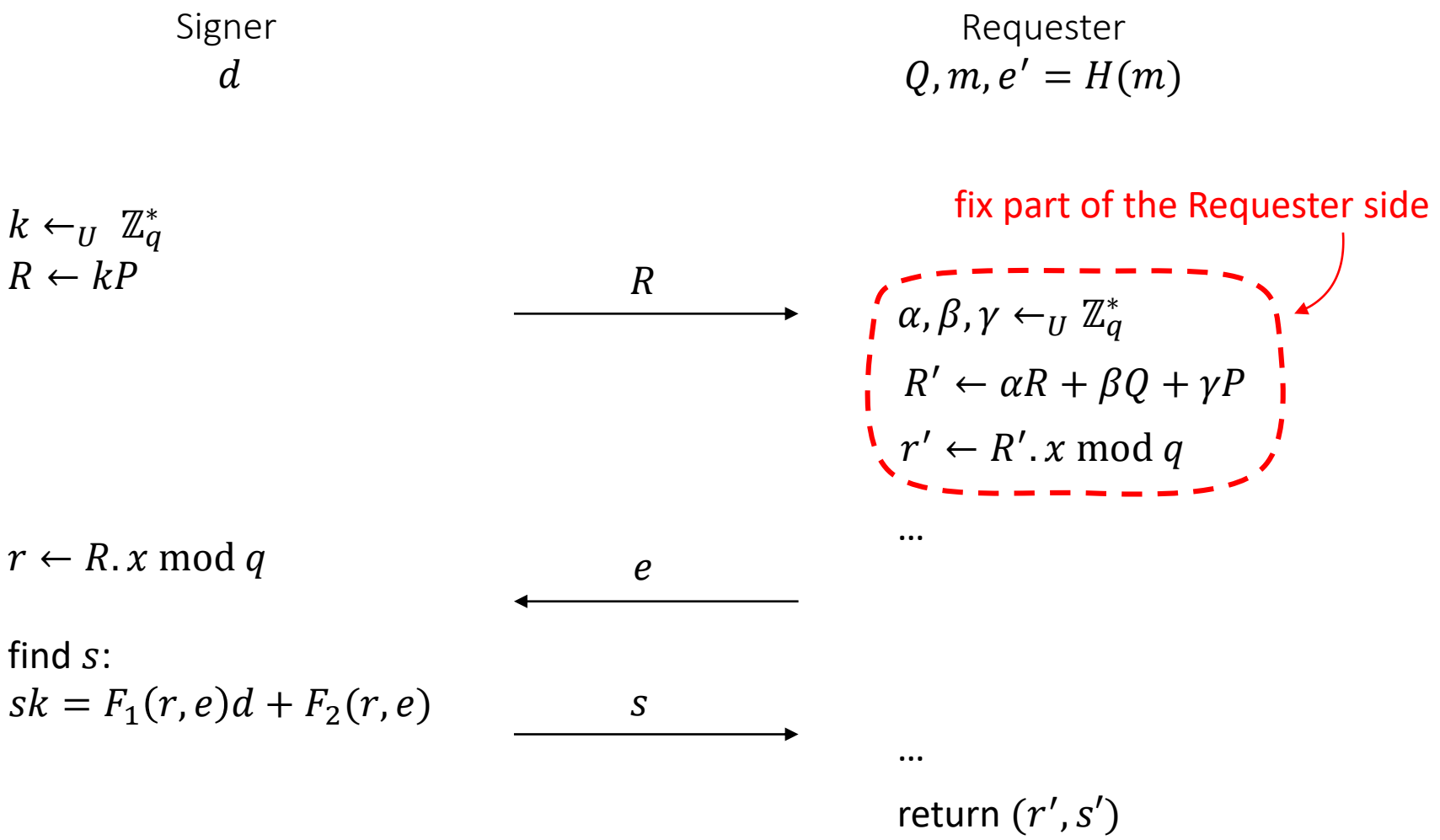
Simplified signature equation for schemes that do not satisfy Condition 1:

$$sk = F_1(r, e)d + F_2(r, e)$$

Schemes of Type II



Schemes of Type II



All known schemes satisfy such r' generation

GenEG-BS schemes

Condition 1

not Condition 1

schemes of Type I

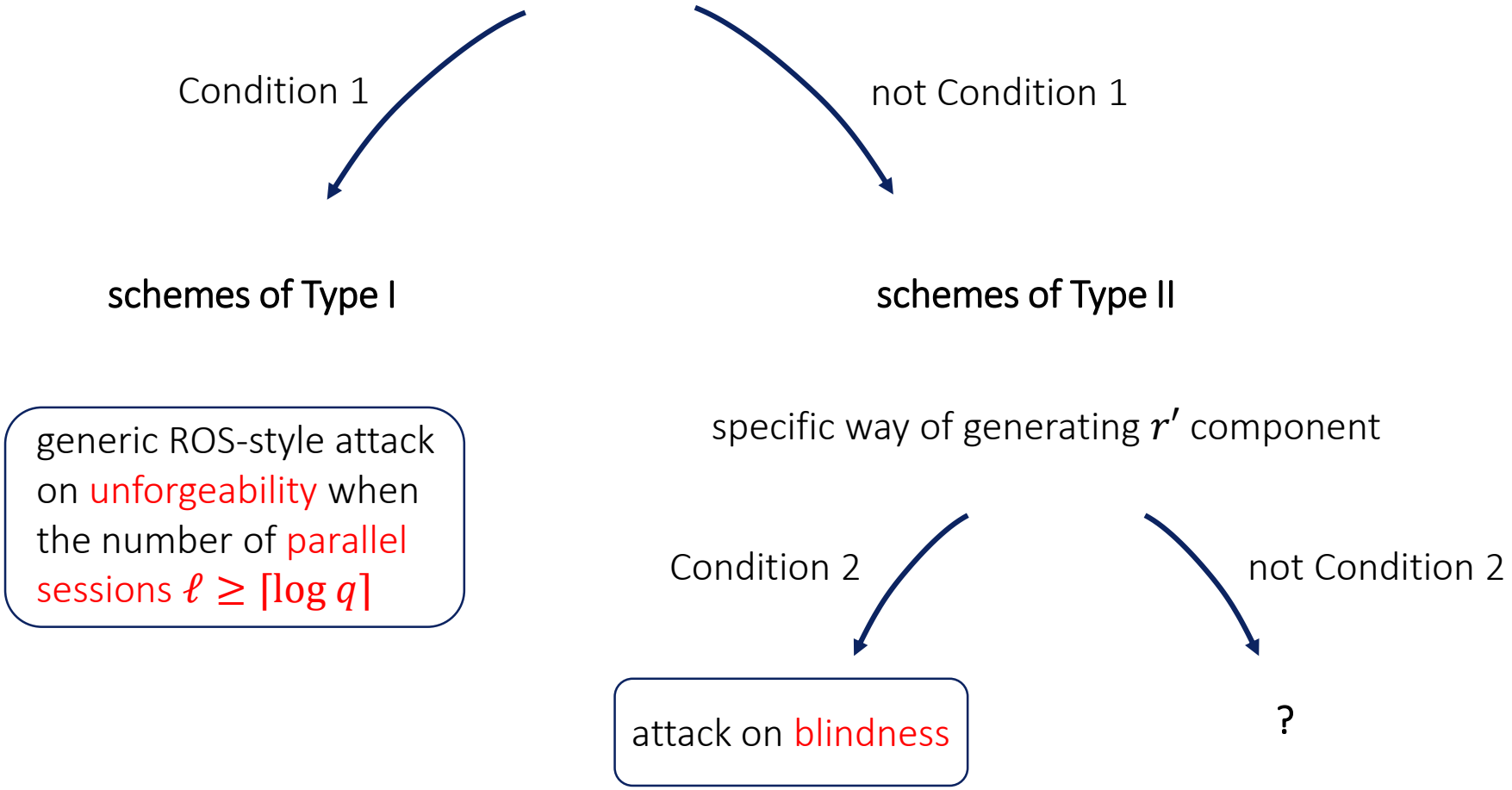
generic ROS-style attack on unforgeability when the number of parallel sessions $\ell \geq \lceil \log q \rceil$

schemes of Type II

specific way of generating r' component

Gorbenko, Yesina, Ponomar, 2016
"Anonymous electronic signature method" – does not provide blindness

GenEG-BS schemes



Blindness attack

Requester

R_1, R_2 →

$r_1 \leftarrow R_1.x \bmod q$
$r_2 \leftarrow R_2.x \bmod q$

← e_1, e_2

s_1, s_2 →

$m_1, (r'_1, s'_1)$ $\quad e'_1 = H(m_1)$

$m_2, (r'_2, s'_2)$ $\quad e'_2 = H(m_2)$

Link transcription (R_i, e_i, s_i) to the signature $m_j, (r'_j, s'_j)$?

Condition 2: for all possible key pairs (d, Q) and messages m the equation

$$F_1(r, e) \cdot F_2(r', e') = F_1(r', e') \cdot F_2(r, e)$$

holds with the overwhelming probability

Blindness attack

Requester

R_1, R_2 →

$r_1 \leftarrow R_1.x \bmod q$
$r_2 \leftarrow R_2.x \bmod q$

← e_1, e_2

s_1, s_2 →

$m_1, (r'_1, s'_1)$ $\quad e'_1 = H(m_1)$
$m_2, (r'_2, s'_2)$ $\quad e'_2 = H(m_2)$

Criteria: (r, e) values from protocol execution match (r', e') values from signature if

$$F_1(r, e) \cdot F_2(r', e') = F_1(r', e') \cdot F_2(r, e)$$

Why this attack works?

- for fixed message $m, e' = H(m)$, and transcription (R, r, e, s) the value r' is defined unambiguously from Condition 2
- negligible probability to choose α, β, γ in several executions such that

$$r' = (\alpha R + \beta Q + \gamma P).x \bmod q$$

GenEG-BS schemes

Condition 1

not Condition 1

schemes of Type I

generic ROS-style attack on unforgeability when the number of parallel sessions $\ell \geq \lceil \log q \rceil$

schemes of Type II

specific way of generating r' component

Condition 2

not Condition 2

attack on blindness

?

there exist no such schemes

GenEG-BS schemes

Condition 1

not Condition 1

schemes of Type I

generic ROS-style attack on unforgeability when the number of parallel sessions $\ell \geq \lceil \log q \rceil$

schemes of Type II

specific way of generating r' component

Condition 2

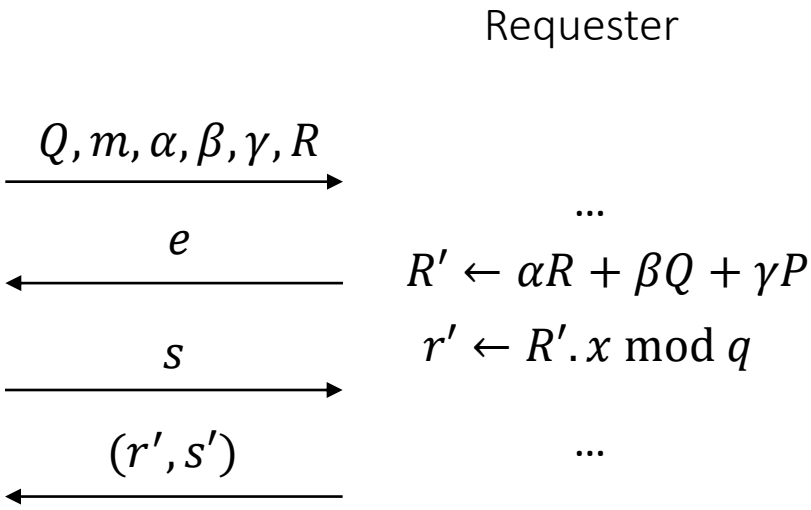
not Condition 2

attack on blindness

if there exists such scheme, then it does not provide unforgeability even with sequential sessions

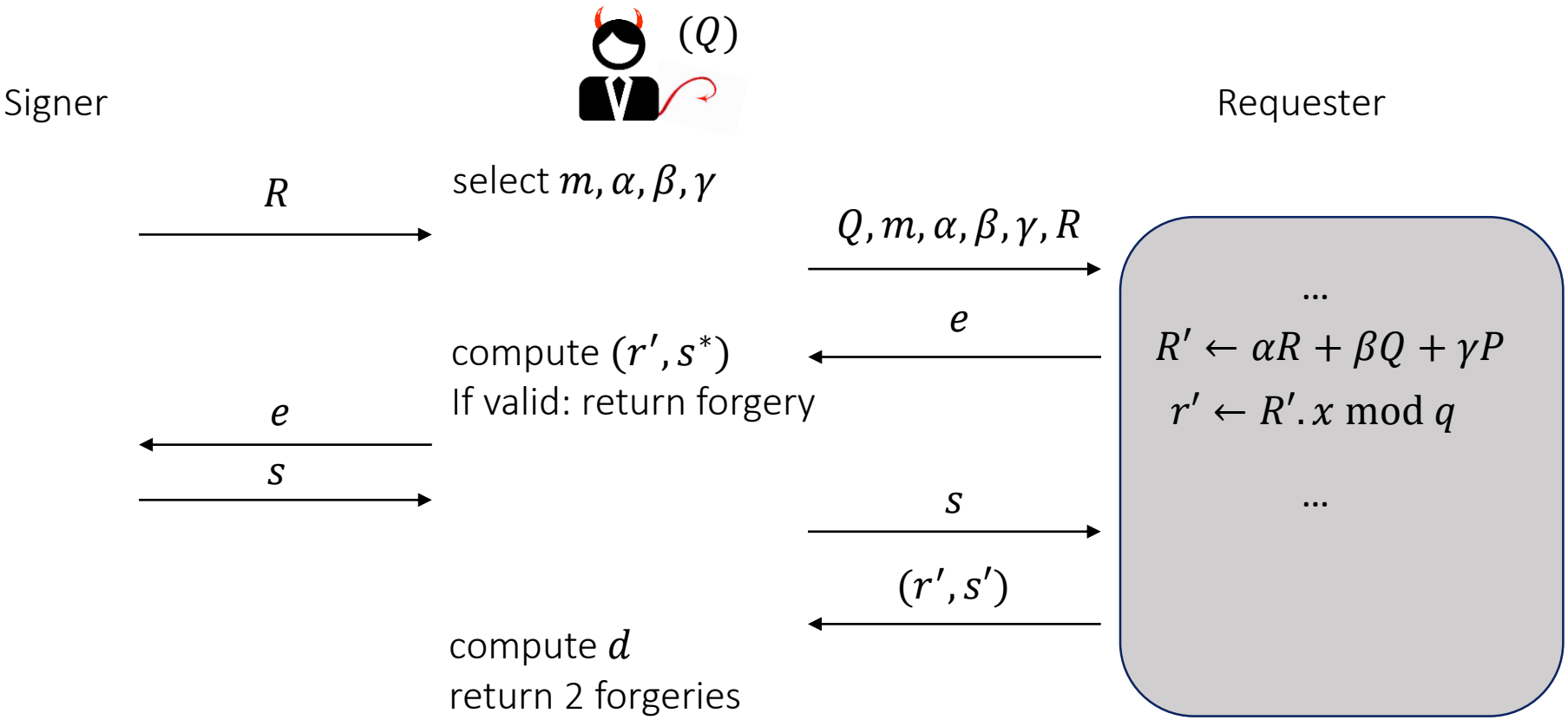
Impossibility result

There exists scheme of Type II ⇒ there exists algorithm Requester:

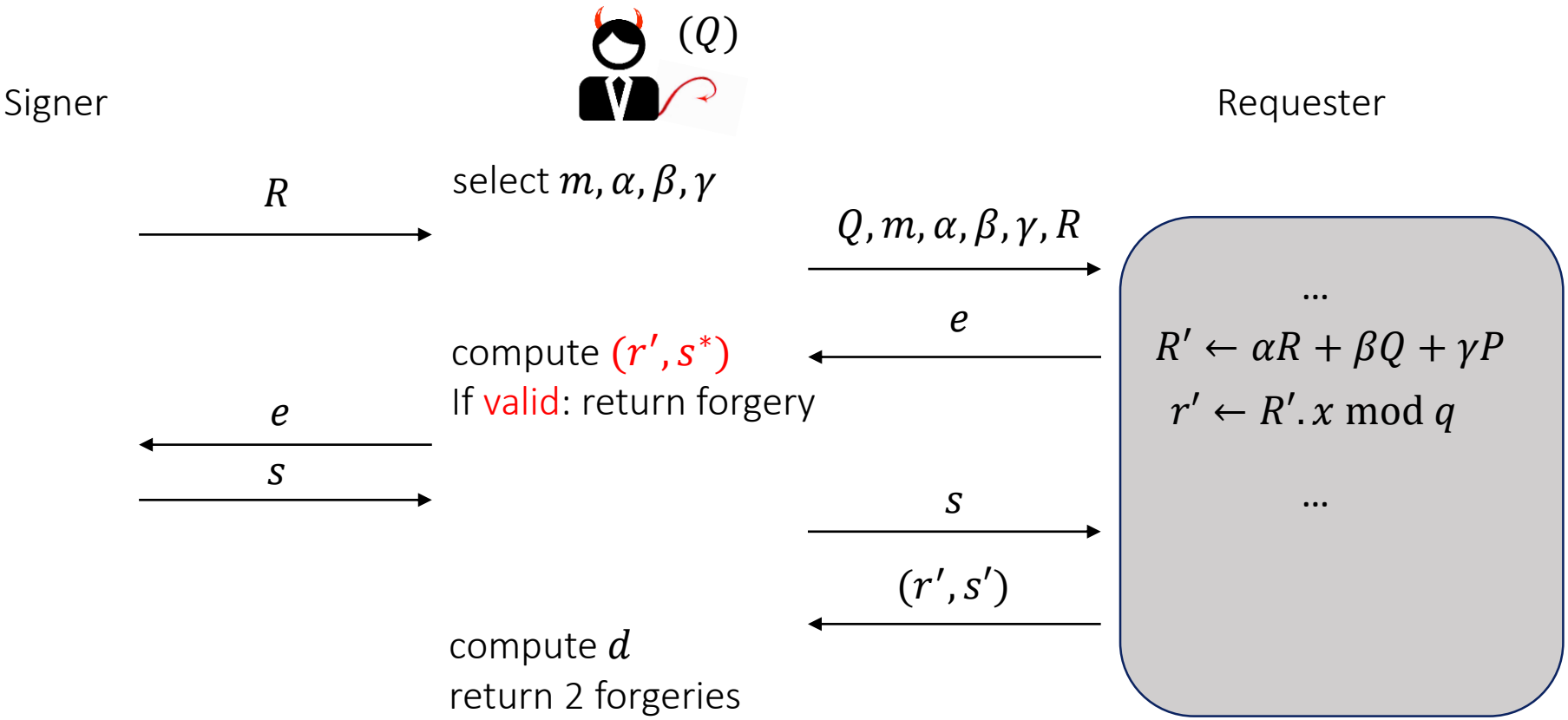


(r', s') – valid signature for m

Impossibility result

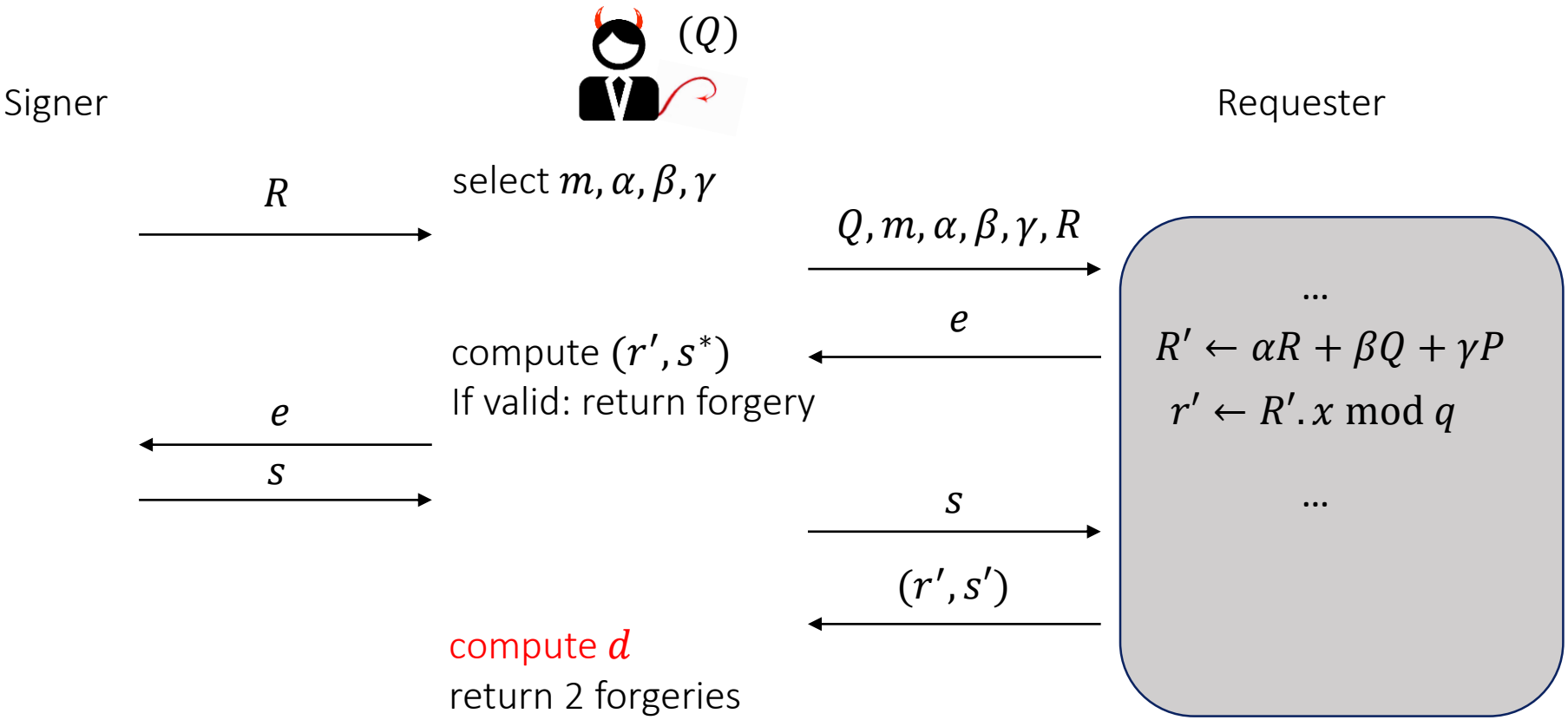


Impossibility result



0 successful interactions, 1 forgery

Impossibility result



1 successful interaction, 2 forgeries

GenEG-BS schemes

Condition 1

not Condition 1

schemes of Type I

generic ROS-style attack on unforgeability when the number of parallel sessions $\ell \geq \lceil \log q \rceil$

schemes of Type II

specific way of generating r' component

Condition 2

not Condition 2

attack on blindness

if there exists such scheme, then it does not provide unforgeability even with sequential sessions

Secure ElGamal blind signature?

Server side?

not plain ElGamal signature generation algorithm

Yi, Lam, 2019
"A new blind ECDSA scheme for bitcoin transaction anonymity"

(use homomorphic encryption and NIZK proof)

plain ElGamal signature generation algorithm (GenEG-BS schemes)

✓ signature equations:

$$sk = F_1(r, e)d + F_2(r, e)$$

✓ radically new way of generating the r' component

Thank you for your attention!

Questions?

babueva@cryptopro.ru