Adaptively Securre Non-Interactive Threshold Cryptosystems: New Framework and Constructions

Benoît Libert[1] and **Moti Yung**[2]

[1]Université catholique de Louvain, Crypto Group – F.N.R.S.
[2] Google Inc. and Columbia University

November 21, 2011

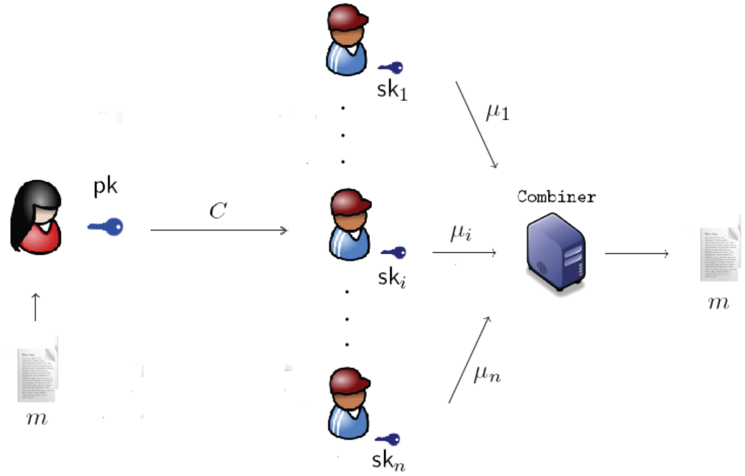
Darmstadt

Threshold Cryptography

- Introduced by Desmedt-Frankel (Crypto'89) and Boyd (IMA'89)
- Split private keys into n shares $SK_1, \ldots, SK_n$ so that knowing strictly less than $t \leq n$ shares is useless to the adversary.
- At least $t \leq n$ shareholders must contribute to private key operations.
  - Decryption requires the cooperation of t decryption servers.
  - Signing requires at least t servers to run a joint signing protocol.
- *Robustness*: up to $t - 1 \leq n$ malicious servers cannot prevent an honest majority from decrypting/signing.

Threshold Cryptography

The public-key encryption case:



Outline

1. Threshold Cryptography
   - Static vs Adaptive corruptions
   - Security notions: CCA2 security and consistency

2. A New Framework for Non-Interactive Threshold CCA2 Encryption
   - All-But-One Perfectly Sound Hash Proof Systems
   - General Construction
   - Instantiations based on Simple Assumptions
   - Efficiency comparisons

Static vs Adaptive corruptions

- Static corruptions: adversary corrupts servers *before* seeing the public key.

  Robust threshold cryptosystems with IND-CCA2 security:

  - Shoup-Gennaro (Eurocrypt'98): in the ROM.
  - Canetti-Goldwasser (Eurocrypt'99): interactive decryption or storage of many pre-shared secrets; non-optimal resilience $t \approx n/3$.
  - Abe (Crypto'99): optimal-resilience $t = (n-1)/2$ in [CG'99].
  - Dodis-Katz (TCC'05): generic constructions; ciphertexts of size $O(n)$.
  - Boneh-Boyen-Halevi (CT-RSA'06): no interaction needed for robustness.
  - Wee (Eurocrypt'11): generic constructions from (threshold) extractable hash proof systems.

Static vs Adaptive corruptions

- Adaptive corruptions: adversary corrupts up to $t-1$ servers *at any time*.
  - Canetti *et al.* (Crypto'99) and Frankel-MacKenzie-Yung (ESA'99, Asiacrypt'99): reliance on erasures.
  - Jarecki-Lysyanskaya (Eurocrypt'00): no need for erasures, but interaction required at decryption with Cramer-Shoup.
  - Lysyanskaya-Peikert (Asiacrypt'01): adaptively secure signatures with interaction.
  - Abe-Fehr (Crypto'04): adaptively secure UC-secure threshold signatures and encryption with interaction.
  - Almansa-Damgaard-Nielsen (Eurocrypt'06): adaptively secure proactive RSA signatures.

Threshold Cryptosystems: Our Goal

- Until recently (and despite more than 10 years of research), adaptive security has not been achieved in threshold encryption schemes with:
  - CCA2-security
  - Non-interactive schemes
  - Robustness against malicious adversaries
  - Optimal resilience ($t = (n-1)/2$)
  - No erasures for shareholders
  - Share size independent of t, n
  - Proof in the standard model

CCA2-Secure Non-interactive Threshold Encryption

Recently (ICALP'11), we described:

- The first *adaptively* secure *fully non-interactive* threshold cryptosystem with
  - CCA2 security and robustness w/o random oracles
  - Short (*i.e.*, $O(1)$-size) private key shares
- The construction
  - Builds on the dual system encryption approach (Waters, Crypto'09) and the Lewko-Waters techniques (TCC'10).
  - Handles adaptive corruptions by instantiating Boneh-Boyen-Halevi (CT-RSA'06) in bilinear groups of order $N = p_1 p_2 p_3$.

    $\Rightarrow$ Ciphertexts live in the subgroup $\mathbb{G}_{p_1}$, private keys in $\mathbb{G}_{p_1 p_3}$
- Gives adaptively secure non-interactive threshold signatures; also yields non-interactive *forward-secure* threshold encryption.

CCA2-Secure Non-interactive Threshold Encryption

New results: a new approach from hash proof systems with *public* verifiability

- Combines universal hash proofs with simulation-sound proofs of ciphertext validity ($\Rightarrow$ publicly verifiable ciphertexts).
- Proofs of validity associated with tags and perfectly sound on *all but one* tag.
- New constructions in groups of order $N = p_1 p_2$ and *prime*-order groups
  - Better efficiency
  - Tighter security (no gap $O(q)$ in the reduction) under a *single* assumption
  - Easier to combine with a DKG protocol

Security of Non-interactive Threshold Encryption

- Chosen-ciphertext (IND-CCA) security:

  1. Challenger generates PK, $\mathbf{SK} = (SK_1, \ldots, SK_n)$ and gives PK to $\mathcal{A}$.
  2. $\mathcal{A}$ makes adaptive queries
     - Corruption $i \in \{1, \ldots, n\}$: $\mathcal{A}$ receives SK_i (up to $t-1$ queries allowed).
     - Decryption (i, C): $\mathcal{A}$ receives $\mu_i = \mathit{Share\text{-}Decrypt}(PK, i, SK_i, C)$
  3. $\mathcal{A}$ chooses M_0, M_1 and gets $C^\star = \mathit{Encrypt}(PK, M_\beta)$ for some $\beta \stackrel{R}{\leftarrow} \{0, 1\}$.
  4. $\mathcal{A}$ makes further queries with restrictions.
  5. $\mathcal{A}$ outputs $\beta' \in \{0, 1\}$ and wins if $\beta' = \beta$

Security of Non-interactive Threshold Encryption

- Chosen-ciphertext (IND-CCA) security:

 1. Challenger generates PK, $\mathbf{SK} = (SK_1, \ldots, SK_n)$ and gives PK to $\mathcal{A}$.
 2. $\mathcal{A}$ makes adaptive queries
    - Corruption $i \in \{1, \ldots, n\}$: $\mathcal{A}$ receives SK_i (up to $t-1$ queries allowed).
    - Decryption (i, C): $\mathcal{A}$ receives $\mu_i = \mathit{Share\text{-}Decrypt}(PK, i, SK_i, C)$
 3. $\mathcal{A}$ chooses M_0, M_1 and gets $C^\star = \mathit{Encrypt}(PK, M_\beta)$ for some $\beta \stackrel{R}{\leftarrow} \{0,1\}$.
 4. $\mathcal{A}$ makes further queries with restrictions.
 5. $\mathcal{A}$ outputs $\beta' \in \{0,1\}$ and wins if $\beta' = \beta$

Security of Non-interactive Threshold Encryption

- Chosen-ciphertext (IND-CCA) security:

1. Challenger generates PK, $\mathbf{SK} = (SK_1, \ldots, SK_n)$ and gives PK to $\mathcal{A}$.
2. $\mathcal{A}$ makes adaptive queries
   - Corruption $i \in \{1, \ldots, n\}$: $\mathcal{A}$ receives SK_i (up to $t-1$ queries allowed).
   - Decryption (i, C): $\mathcal{A}$ receives $\mu_i = \mathit{Share\text{-}Decrypt}(PK, i, SK_i, C)$
3. $\mathcal{A}$ chooses M_0, M_1 and gets $C^\star = \mathit{Encrypt}(PK, M_\beta)$ for some $\beta \stackrel{R}{\leftarrow} \{0, 1\}$.
4. $\mathcal{A}$ makes further queries with restrictions.
5. $\mathcal{A}$ outputs $\beta' \in \{0, 1\}$ and wins if $\beta' = \beta$

Security of Non-interactive Threshold Encryption

- Chosen-ciphertext (IND-CCA) security:

1. Challenger generates PK, $\mathbf{SK} = (SK_1, \ldots, SK_n)$ and gives PK to $\mathcal{A}$.
2. $\mathcal{A}$ makes adaptive queries
   - Corruption $i \in \{1, \ldots, n\}$: $\mathcal{A}$ receives SK_i (up to $t-1$ queries allowed).
   - Decryption (i, C): $\mathcal{A}$ receives $\mu_i = \mathit{Share\text{-}Decrypt}(PK, i, SK_i, C)$
3. $\mathcal{A}$ chooses M_0, M_1 and gets $C^\star = \mathit{Encrypt}(PK, M_\beta)$ for some $\beta \stackrel{R}{\leftarrow} \{0,1\}$.
4. $\mathcal{A}$ makes further queries with restrictions.
5. $\mathcal{A}$ outputs $\beta' \in \{0,1\}$ and wins if $\beta' = \beta$

Security of Non-interactive Threshold Encryption

- Chosen-ciphertext (IND-CCA) security:

1. Challenger generates PK, $\mathbf{SK} = (SK_1, \ldots, SK_n)$ and gives PK to $\mathcal{A}$.
2. $\mathcal{A}$ makes adaptive queries
   - Corruption $i \in \{1, \ldots, n\}$: $\mathcal{A}$ receives SK_i (up to $t-1$ queries allowed).
   - Decryption (i, C): $\mathcal{A}$ receives $\mu_i = \textit{Share-Decrypt}(PK, i, SK_i, C)$
3. $\mathcal{A}$ chooses M_0, M_1 and gets $C^\star = \textit{Encrypt}(PK, M_\beta)$ for some $\beta \stackrel{R}{\leftarrow} \{0, 1\}$.
4. $\mathcal{A}$ makes further queries with restrictions.
5. $\mathcal{A}$ outputs $\beta' \in \{0, 1\}$ and wins if $\beta' = \beta$

Security of Non-interactive Threshold Encryption

- Consistency:

  1. Challenger generates PK, $\mathbf{SK} = (SK_1, \ldots, SK_n)$ and gives PK to $\mathcal{A}$.
  2. $\mathcal{A}$ makes adaptive queries
     - Corruption query $i \in \{1, \ldots, n\}$: $\mathcal{A}$ receives SK_i.
     - Decryption query (i, C): $\mathcal{A}$ receives $\mu_i = \mathit{Share\text{-}Decrypt}(PK, i, SK_i, C)$
  3. $\mathcal{A}$ outputs a ciphertext C and sets $S = \{\mu_1, \ldots, \mu_t\}$, $S' = \{\mu'_1, \ldots, \mu'_t\}$ of shares such that
     - C is a valid ciphertext.
     - S and S' are sets of valid shares.
     - $\mathit{Combine}(PK, C, S) \neq \mathit{Combine}(PK, C, S')$.

A New Framework for Adaptive Security

Based on Hash Proof Systems:

- Let $\mathcal{C}$ be a set and $\mathcal{V} \subset \mathcal{C}$ be a subset; let (pk, sk) be a key pair such that
  - If $\Phi \in \mathcal{V}$, $\mathsf{PrivEval}(sk, \Phi)$ is completely fixed by Φ and pk (and computable as $\mathsf{PubEval}(pk, \Phi, r)$ using a witness r that $\Phi \in \mathcal{V}$).
  - If $\Phi \in \mathcal{C} \backslash \mathcal{V}$, $\mathsf{PrivEval}(sk, \Phi)$ is information-theoretically hidden.
- $D_1 = \{\Phi \mid \Phi \stackrel{R}{\leftarrow} \mathcal{V}\}$ is indistinguishable from $D_0 = \{\Phi \mid \Phi \stackrel{R}{\leftarrow} \mathcal{C} \backslash \mathcal{V}\}$.
- Message M can be encrypted as $(C_0, C_1) = \big(M \cdot \mathsf{PubEval}(pk, \Phi, r), \Phi\big)$ and decrypted as $M = C_0 \cdot \mathsf{PrivEval}(sk, C_1)^{-1}$.
- In the security proof, to decide if $\Phi^\star \in \mathcal{V}$ or $\Phi^\star \in \mathcal{C} \backslash \mathcal{V}$, set

$$(C_0^\star, C_1^\star) = \big(M_\beta \cdot \mathsf{PrivEval}(sk, \Phi^\star), \Phi^\star\big).$$

A New Framework for Adaptive Security

- In the security proof, to decide if $\Phi^\star \in \mathcal{V}$, set

$$(C_0^\star, C_1^\star) = \big(M_\beta \cdot \mathsf{PrivEval}(sk, \Phi^\star), \Phi^\star\big).$$

- Private key sk is available to the reduction.

- For CCA2-security, the reduction should reject $(C_0, C_1 = \Phi)$ if $\Phi \notin \mathcal{V}$.

  $\Rightarrow$ Cramer-Shoup uses non-interactive designated-verifier proofs that $\Phi \in \mathcal{V}$

- In the threshold setting, $\Phi \in \mathcal{V}$ cannot be checked from partial decryptions.

  $\Rightarrow$ Existing solutions [CG99,JL00,AF04] require interaction to render ciphertexts with $\Phi \notin \mathcal{V}$ harmless.

A New Framework for Adaptive Security

Our approach: *All-But-One Perfectly Sound* Hash Proof Systems

- Combination between
  - Universal hash proofs (simulator knows private keys in reduction).
  - Simulation-sound proofs of ciphertext validity (publicly verifiable ciphertexts).
- Proofs of validity associated with tags and perfectly sound on *all but one* tag.
- Gives new constructions
  - Based on the Subgroup Decision assumption in composite order groups with two primes $N = p_1 p_2$.
  - Or Groth-Sahai proofs (D-Linear/SXDH assumptions) in *prime-order* groups:

    $\Rightarrow$ Better efficiency; easier to combine with a DKG protocol.

All-But-One Perfectly Sound Hash Proof Systems

Non-interactive proofs that $\Phi \in \mathcal{V}$ are associated with tags

- Two distinct setup procedures
  - $\mathsf{SetupSound}(\lambda, t, n)$: gives $\big(pk, \{sk_i\}_{i=1}^{n}\big)$ where pk yields sound proofs.
  - $\mathsf{SetupABO}(\lambda, t, n, \mathsf{tag}^\star)$: gives $\big(pk, \{sk_i\}_{i=1}^{n}\big)$ and a trapdoor τ such that proofs are perfectly sound on all tags but $tag^\star$.
- Two distinct proving algorithms
  - $\mathsf{Prove}(pk, \mathsf{tag}, r, \Phi)$: returns real proofs using the witness r that $\Phi \in \mathcal{V}$.
  - $\mathsf{SimProve}(pk, \tau, \mathsf{tag}^\star, \Phi)$: returns a simulated proof for any $\Phi \in \mathcal{C}$.

All-But-One Perfectly Sound Hash Proof Systems

Main properties:

- SETUP INDISTINGUISHABILITY: $\mathsf{SetupSound}(\lambda, t, n)$ and $\mathsf{SetupABO}(\lambda, t, n, \mathsf{tag}^\star)$ have indistinguishable public outputs.
- ALL-BUT-ONE SOUNDNESS:
  a. For any $(pk, (sk_1, \ldots, sk_n), \tau) \leftarrow \mathsf{SetupABO}(\lambda, t, n, \mathsf{tag}^\star)$ and any $\mathsf{tag} \neq \mathsf{tag}^\star$, if $\pi_{\mathcal{V}}$ is a valid proof w.r.t. tag, then $\Phi \in \mathcal{V}$.
  b. For any $(pk, (sk_1, \ldots, sk_n), \tau) \leftarrow \mathsf{SetupABO}(\lambda, t, n, \mathsf{tag}^\star)$, $\mathsf{SimProve}(pk, \tau, \mathsf{tag}^\star, \Phi)$ gives a NIZK proof that $\Phi \in \mathcal{V}$ for any $\Phi \in \mathcal{C}$.

General Construction of Threshold CCA2 Cryptosystem

- **Keygen**(λ, t, n): runs $\mathsf{SetupSound}(\lambda, t, n)$ to obtain $\big(pk, \{sk_i\}_{i=1}^n\big)$.
- **Encrypt**(pk, M): generate a one-time signature key pair $(\mathsf{SK}, \mathsf{VK}) \leftarrow \mathcal{G}(\lambda)$,
  1. Sample $\Phi \stackrel{R}{\leftarrow} \mathcal{V}$ using random coins r.
  2. Compute $C_0 = M \cdot \mathsf{PubEval}(pk, r, \Phi)$.
  3. Compute a proof $\pi_{\mathcal{V}} \leftarrow \mathsf{Prove}(pk, \mathsf{VK}, \Phi)$ that $\Phi \in \mathcal{V}$.

  Return $C = (\mathsf{VK}, C_0, \Phi, \pi_{\mathcal{V}}, \sigma)$, where $\sigma = \mathcal{S}(\mathsf{SK}, (C_0, \Phi, \pi_{\mathcal{V}}))$.
- **Share-Decrypt**(sk_i, pk, C):
  1. Return $\perp$ if $\mathcal{V}(\mathsf{VK}, \sigma, (C_0, \Phi, \pi_{\mathcal{V}})) = 0$ or $\pi_{\mathcal{V}}$ is an invalid proof w.r.t. VK.
  2. Otherwise, compute a share $\mathsf{PrivEval}(sk_i, \Phi)$ with a proof of validity.
- **Combine**: verifies all decryption shares and combines them.

Security

Theorem

The scheme is consistent and **IND-CCA2** *under adaptive corruptions if*

- Σ *is a strong one-time signature.*
- *The ABO-PS-HPS is secure*

Idea of the proof of IND-CCA security:

- CRS only allows NIZK proofs in the challenge ciphertext and only the challenger can generate *one* fake proof.
- Adversary can only prove true statements (cf. one time simulation-soundness).
- Simulator knows the decryption keys (as in HPS-based proofs).

Instantiation in groups of order $N = p_1 p_2$

Subgroup Decision Problem: in a group $\mathbb{G}$ of order $N = p_1 p_2$, given $\big(g \in \mathbb{G}_{p_1},\ h \in \mathbb{G}\big)$ and η, decide if $\eta \in_R \mathbb{G}_{p_1}$ or $\eta \in_R \mathbb{G}$.

An ordinary Hash Proof System: let $\mathcal{C} = \mathbb{G}$ and $\mathcal{V} = \mathbb{G}_{p_1}$.

- Setup(λ):
  1. Choose a group $\mathbb{G}$ of order $N = p_1 p_2$ with $g \stackrel{R}{\leftarrow} \mathbb{G}_{p_1}$.
  2. Set $X = g^x$ with $x \stackrel{R}{\leftarrow} \mathbb{Z}_N$.
  3. Let $H : \mathbb{G} \rightarrow \{0,1\}^{\ell}$ be a pairwise independent hash function for some ℓ.

  Output $pk = \big(\mathbb{G},\ N,\ g,\ X,\ H\big)$ and $sk = x$.

- PubEval(pk, r, Φ): given $r \in \mathbb{Z}_N$ such that $\Phi = g^r$, output $H(X^r)$.
- PrivEval(sk, Φ): given $\Phi \in \mathbb{G}_{p_1}$, output $H(\Phi^x)$.

Instantiation in groups of order $N = p_1 p_2$

Define $\mathcal{C} = \mathbb{G}$ and $\mathcal{V} = \mathbb{G}_{p_1}$.

- SetupSound(λ, t, n): chooses $g \stackrel{R}{\leftarrow} \mathbb{G}_{p_1}$, $u, v \stackrel{R}{\leftarrow} \mathbb{G}$.
- SetupABO$(\lambda, t, n, \mathsf{tag}^\star)$: is like SetupSound but chooses $v = u^{-\mathsf{tag}^\star} \cdot g^\alpha$ where $\alpha \stackrel{R}{\leftarrow} \mathbb{Z}_N$ is the trapdoor $\tau := \alpha$.
- Prove$(pk, \mathsf{tag}, r, \Phi)$: given $\Phi = g^r \in \mathbb{G}_{p_1}$, output $\pi_{\mathcal{V}} = (u^{\mathsf{tag}} \cdot v)^r$ such that
$$e(g, \pi_{\mathcal{V}}) = e(\Phi, u^{\mathsf{tag}} \cdot v),$$
which guarantees $\Phi \in \mathbb{G}_{p_1}$.
- SimProve$(pk, \tau, \mathsf{tag}^\star, \Phi)$: given $\tau = \alpha \in \mathbb{Z}_N$, output $\pi_{\mathcal{V}} = \Phi^\alpha$, which satisfies
$$e(g, \pi_{\mathcal{V}}) = e(\Phi, u^{\mathsf{tag}^\star} \cdot v)$$
since $u^{\mathsf{tag}^\star} \cdot v = g^\alpha$.

Instantiation in prime order groups

Instantiation based on Groth-Sahai proofs and the D-Linear assumption:

- **Linear Problem**: given $(g, g_1, g_2, g_1^a, g_2^b, Z)$, decide if $Z \stackrel{?}{=} g^{a+b}$.
- Equivalently, given

$$\vec{g_1} = (g_1, 1, g), \quad \vec{g_2} = (1, g_2, g), \quad \vec{\varphi} = (g_1^a, g_2^b, Z),$$

  decide whether $\vec{g_1}, \vec{g_2}, \vec{\varphi}$ are linearly dependent (*i.e.*, $\vec{\varphi} \stackrel{?}{=} \vec{g_1}^{a} \cdot \vec{g_2}^{b}$).
- To commit to $x \in \mathbb{Z}_p$, set $\vec{C} = \vec{\varphi}^x \cdot \vec{g_1}^{t_1} \cdot \vec{g_2}^{t_2}$.
- Dual mode commitments:
  - Perfect binding commitments and perfectly sound proofs if $\vec{\varphi} \notin \mathrm{span}(\vec{g_1}, \vec{g_2})$.
  - Perfectly hiding commitments and WI proofs if $\vec{\varphi} \in \mathrm{span}(\vec{g_1}, \vec{g_2})$.

Instantiation in prime order groups

Linear Problem: given $(g, g_1, g_2, g_1^a, g_2^b, Z)$, decide if $Z \stackrel{?}{=} g^{a+b}$.

An ordinary HPS: given $g_1, g_2, g \in \mathbb{G}$, let $\mathcal{C} = \mathbb{G}^3$ and $\mathcal{V} = (g_1^a, g_2^b, g^{a+b})$.

- Setup(λ): choose a group $\mathbb{G}$ of order p with $g \stackrel{R}{\leftarrow} \mathbb{G}$ and set

$$pk = (\mathbb{G},\ g,\ g_1,\ g_2,\ X_1 = g_1^{x_1} \cdot g^z,\ X_2 = g_2^{x_2} \cdot g^z)$$

  where $sk = (x_1, x_2, z) \stackrel{R}{\leftarrow} \mathbb{Z}_p^3$.

- PubEval(pk, r, Φ): given $(r, s) \in \mathbb{Z}_p^2$ s.t. $(\Phi_1, \Phi_2, \Phi_3) = (g_1^r, g_2^s, g^{r+s})$, output

$$X_1^r \cdot X_2^s.$$

- PrivEval(sk, Φ): given $\Phi = (\Phi_1, \Phi_2, \Phi_3) \in \mathbb{G}^3$, output $\Phi_1^{x_1} \cdot \Phi_2^{x_2} \cdot \Phi_3^z$.

Instantiation in groups of prime order

Define $\mathcal{C} = \mathbb{G}^3$ and $\mathcal{V} = (g_1^a, g_2^b, g^{a+b})$.

- SetupSound(λ, t, n): set

$$\vec{g_1} = (g_1, 1, g), \qquad \vec{g_2} = (1, g_2, g), \qquad \vec{\varphi} = \vec{g_1}^{\,a} \cdot \vec{g_2}^{\,b}.$$

- SetupABO$(\lambda, t, n, \mathsf{tag}^\star)$: is like SetupSound$(\lambda, t, n)$ but

$$\vec{\varphi} = \vec{g_1}^{\,a} \cdot \vec{g_2}^{\,b} \cdot (1, 1, g)^{-\mathsf{tag}^\star}$$

  and the trapdoor is $\tau := (a, b) \in \mathbb{Z}_p^2$.

- Prove$(pk, \mathsf{tag}, r, \Phi)$: given $\Phi = (\Phi_1, \Phi_2, \Phi_3) = (g_1^r, g_2^s, g^{r+s})$ and (r, s), generate a proof that $\Phi \in \mathcal{V}$ w.r.t. the CRS $(\vec{g_1}, \vec{g_2}, \vec{\varphi} \cdot (1, 1, g)^{\mathsf{tag}})$.

- SimProve$(pk, \tau, \mathsf{tag}^\star, \Phi)$: simulate a NIZK proof using $\tau = (a, b) \in \mathbb{Z}_p^2$ on the "fake" CRS $(\vec{g_1}, \vec{g_2}, \vec{\varphi} \cdot (1, 1, g)^{\mathsf{tag}^\star})$.

Efficiency comparisons

- Estimations at the 128-bit security level

Approaches	Group order	Assumptions	Ciphertext overhead (# of bits)
Dual system	$N = p_1 p_2 p_3 > 2^{3072}$	Subgroup Decision Assumptions	6144
ABO-PS-HPS	$p > 2^{512}$	D-Linear	10240
ABO-PS-HPS	$p > 2^{256}$	SXDH	3328

Figure: Comparisons in terms of ciphertext overhead

- Under D-Linear: 12 pairings to check ciphertexts (using batch-verification); sender computes 19 exponentiations.
- Under SXDH: only 6 pairings to check ciphertexts (with batch-verification); sender computes 7 exponentiations.

Conclusion

- We described
  - A framework for CCA2-secure *robust* and *non-interactive* threshold cryptosystems secure against *adaptive* corruptions
  - Constructions in prime order groups using simple assumptions
    - Better efficiency
    - Compatibility with adaptively secure DKG protocols
  - ...with tight security proofs using fewer assumptions
- Open problems:
  - Are there instantiations without pairings?
  - Can we do the same for threshold signatures?