"Extreme" Threshold Cryptosystems:

Adaptively Secure Non-Interactive Threshold Cryptosystems

Benoît Libert[1] and **Moti Yung**[2]

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

August 16, 2011

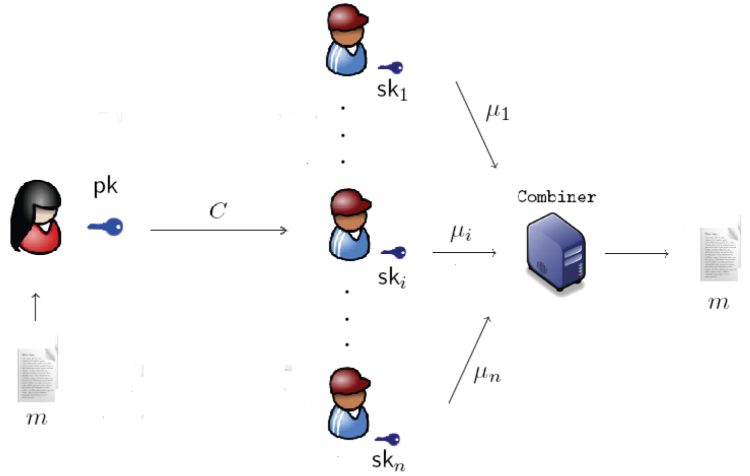
Santa Barbara

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 a honest majority from decrypting/signing.

Threshold Cryptography

The public-key encryption case:



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): requires interaction or storage of many pre-shared secrets; robust and adaptively secure for $t = O(n^{1/2})$.
  - 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): need for erasures.
  - Jarecki-Lysyanskaya (Eurocrypt'00): no need for erasures, but interaction 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

- Despite more than 10 years of research, adaptive security has not been achieved with:
  - CCA2-security for encryption and CMA-security for signatures.
  - 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

Our contribution (ICALP'11):

- An *adaptively* secure *fully non-interactive* threshold cryptosystem providing
  - 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

New Results: An Alternative 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.

Example: using the Linear assumption

- Use Damgaard's Elgamal with $PK = (g,\ g_1,\ g_2,\ X_1 = g_1^{x_1} g^z,\ X_2 = g_2^{x_2} g^z)$.

$$C_0 = M \cdot X_1^r \cdot X_2^s, \qquad C_1 = g_1^r, \qquad C_2 = g_2^s, \qquad C_3 = g^{r+s}$$

- Add a simulation-sound proof that $(C_1, C_2, C_3) = (g_1^r, g_2^s, g^{r+s})$ using a CRS that depends on VK, where $(\mathsf{SK}, \mathsf{VK}) \leftarrow \mathcal{G}(\lambda)$ is a one-time key pair.

- Security proof works:
  - CRS is only WI for the challenge ciphertext and only the challenger can generate *one* fake proof.
  - Adversary can only prove true statements.
  - Simulator knows the decryption keys (as in HPS-based proofs).

Thanks!