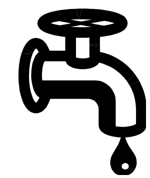
Public-Key Encryption in the Bounded-Retrieval Model

Joël Alwen, Yevgeniy Dodis, Moni Naor,
Gil Segev, Shabsi Walfish, **Daniel Wichs**

Leakage Resilience and the BRM

- Leakage Resilience: [AGV09, NS09,…]
  Cryptographic schemes that remain secure even if adversary learns **partial information** about sk.
  - Goal: High relative leakage.

sk



f(sk)

- Bounded Retrieval Model: [Dzi06, CLW06,…]
  Absolute size of leakage can be arbitrarily large (bits, Mb, Gb…).
  - Accommodate any leakage threshold by increasing key size flexibly.
  - No other loss of efficiency!
    - Small Public Key and Ciphertext.
    - Efficient Encryption/Decryption
    - Independent of leakage.

leak

90% of |sk|

Why have schemes in the BRM?

- Security against viruses:
  - Virus downloads arbitrary information from local storage and sends it to a remote attacker.
  - In practice, virus cannot download too much (< 10 GB).
    - Bandwidth too low, Cost too high, System security may detect.

- Security against side-channel attacks:
  - Adversary gets some "physical output" of computation.
  - May be unreasonable to learn "too much" info, even after many physical readings.
  - How much is "too much" depends on physical implementation (few Kb - few Mb).

Prior Work

- Leakage Resilience (No BRM):
  - Symmetric-Key Authenticated Encryption [DKL09]
  - Public-Key Encryption [AGV09, NS09, KV09]
  - Signatures [ADW09, KV09]

- Bounded Retrieval Model:
  - Secret Sharing [DP07]
  - Symmetric-Key Identification and Authenticated Key Agreement [Dzi06, CDD+07]
  - Public-Key ID schemes, Signatures, Authenticated Key Agreement [ADW09]

Public-Key Encryption in the BRM

- Now: Public-Key Encryption in the BRM.

- Result: PKE parameterized by security parameter s (e.g. 1024 bits) and leakage bound L (e.g. 1024 bits - 10GB).
  - Secret Key size is flexible: $|sk| = (1 + \varepsilon)L$.
  - Public Keys and Ciphertexts are short, only depend on s.
  - Decryption is local. Number of bits accessed is proportional to s.

PKE in the BRM via IBE

- Idea: Use Leakage-Resilient IBE to construct PKE in BRM.
  - Generate a master-key pair (MPK, MSK) for an IBE.
    - Use MSK to generate keys $sk_1,\dots, sk_n$ for identities $1,\dots,n$.
    - Set $PK = MPK$, $SK = (sk_1,\dots, sk_n)$. Delete MSK.
  - To encrypt m:
    - Choose t random identities $ID_i \in [n]$.
    - Compute shares $(s_1,\dots, s_t)$ such that $m = s_1 + \dots + s_t$.
    - Set $c_1=Enc(ID_1, s_1),\dots, c_n=Enc(ID_t, s_t)$.
    - Ciphertext is $C = (ID_1 ,\dots, ID_t , c_1 ,\dots, c_t)$.

- Good news: Ciphertext, Public-Key, Locality is proportional to security parameter.
- Need leakage resilient IBE. (Of independent interest)
- Is the construction secure? How much leakage?

Security of IBE-based Construction

- Does IBE-based construction amplify leakage resilience?
- Hope: If IBE is secure for leakage of L bits of the per-identity secret keys, is the BRM scheme secure for nL bits?

- Answers:
  - Bad News: Not in general. Have artificial counterexample.
  - Good news: Works for PKE/IBE of special form.

Construction

- New notion: "Identity Based Hash-Proof System" (IB-HPS).
  - Hash Proof Systems were shown to give LR PKE in [NS09]
  - Extend to "Identity-Based" setting.
    - Master PK. Secret key for each identity.

- Result 1: IB-HPS gives us Leakage-Resilient IBE.
- Result 2: IB-HPS gives us efficient PKE in BRM.

- Construction based on the [Gentry06] IBE .
  - Bilinear assumption (q-ABDHA).
- Construction based on [GPV08] IBE.
  - Lattice assumption (LWE) + RO model.