# Public Key Cryptography in the Bounded Retrieval Model

Joint work with Joël Alwen, Shabsi Walfish & Daniel Wichs



Speaker: Yevgeniy Dodis (NYU)

# Leakage Attacks

- Standard Crypto Assumption: keys stored secretly.
- Reality: information leaks
  - Timing attacks, Power consumption attacks, Freezing attacks, Hackers, Malware, Viruses…
- Usual Crypto Response: not our problem.
- Better Crypto Response: provably secure primitives that allow leakage.
  - Assume leakage *arbitrary but incomplete.*

# Modeling Incomplete Leakage

- Adversary can learn *any efficiently computable function* f : $\{0,1\}^* \to \{0,1\}^L$ of the secret key.
  L = Leakage Bound.
  - Relative leakage [AGV09, DKS09, NS09, KV09].
    - Key size dependent on security parameter (e.g. 1024 bits). Leakage L is dependent on key size (e.g. 50% of key size).
    - **Goal**: Allow for large percentage of leakage.
    - **Problem**: in reality, leakage may be large in *absolute* terms (e.g. L can be on scale of Kbs, Mbs or even Gbs)
      - For example: hackers/malware/virus attacks.
  - More robust model: Absolute leakage

# Modeling Incomplete Leakage

- Adversary can learn *any efficiently computable function* $f : \{0,1\}^* \rightarrow \{0,1\}^L$ of the secret key.
  L = Leakage Bound. k = Security Parameter
  - Relative leakage [AGV09, DKS09, NS09, KV09].
  - Bounded Retrieval Model (BRM) [Dzie06, CLW06, DP07]:
    - Key size |SK| depends on security parameter k AND leakage bound L. (Note: must be more than L)
    - Other efficiency parameters only depend on k.
      - E.g., public key, communication, computation, read-locality
    - **Goal**: flexibly accommodate ANY leakage bound L ***ONLY*** by increasing |SK| and *without* impacting other parameters.

# Our Results

- Efficient constructions of virtually all *public key* primitives in the BRM:
  - ID, Signatures, Authenticated Key Agreement (AKA) [ADW09].
    - Based on Okamoto ID/Sigs.
  - Encryption, IBE [ADWW09].
    - Based on Gentry IBE.
- Efficiency: Leakage bound $L$. Security parameter $k$.
  - Secret key size: $O(L)$, in some cases $L(1+\varepsilon)$.
  - Public key size: Constant number of group elements.
  - Communication:
    - ID/Sig/AKA: Constant number of group elements.
    - Enc/IBE: $O(k)$ group elements.
  - Data Accessed: $O(k)$ group elements.
  - Computation: $O(k)$ exponentiations.

# What does it mean? For example…

- An efficient Authenticated Key Agreement (AKA) protocol with short public key and 10 GB secret key.
  - All other efficiency parameters "short" as well
- A virus must download at least 5 GB of information to *impersonate* the infected computer
- All sessions completed prior to infection remain secure, even if virus learns the entire 10 GB key.
  - Major advantage over encryption [AGV09,NS09,KV09,ADWW09].
- Almost as efficient as standard protocols.