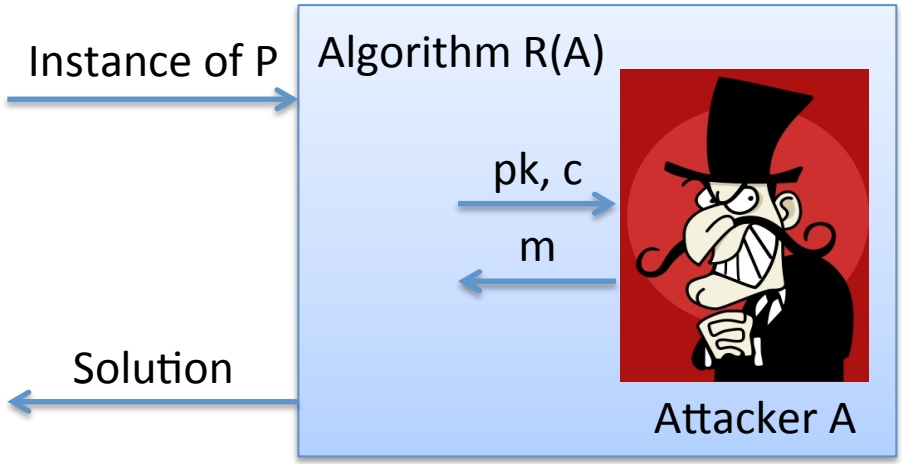
Tightly Secure Signatures and Public-Key Encryption

Dennis Hofheinz and Tibor Jager
Karlsruhe Institute of Technology

CRYPTO 2012

“Tight” Security

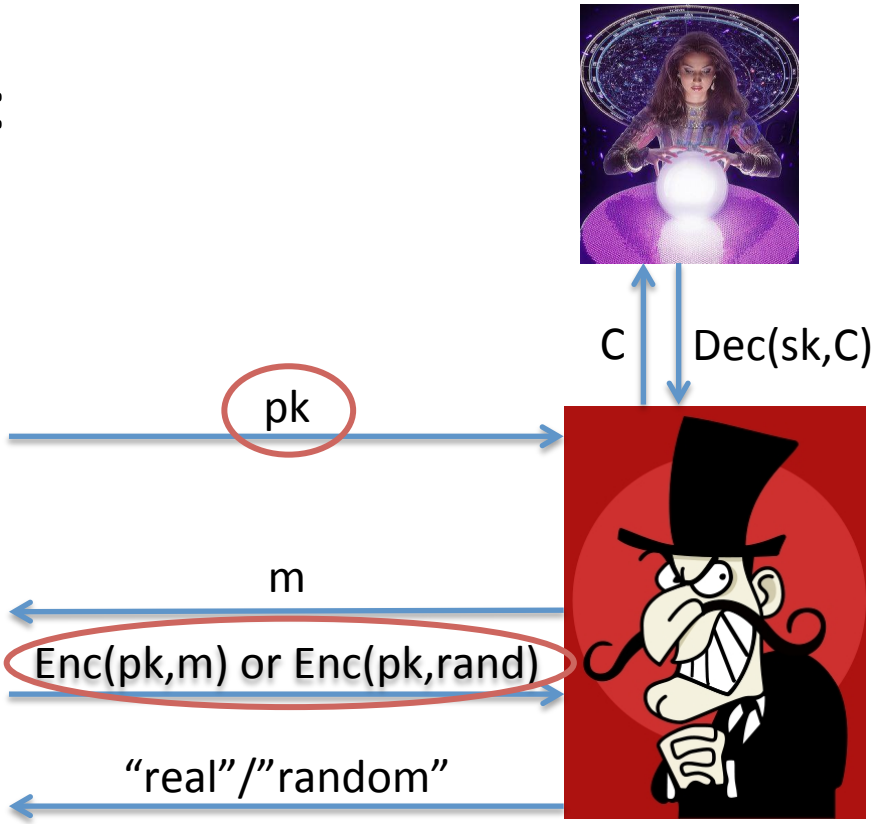
- New cryptosystems usually come with proof that
  - under certain **complexity assumptions**
  - the scheme has certain security properties (“**security proof**”)
- Standard outline of a security proof:
  1. Define a **security model**
  2. Show that an efficient **attacker A** implies an efficient algorithm **R(A)** solving some **hard problem P**



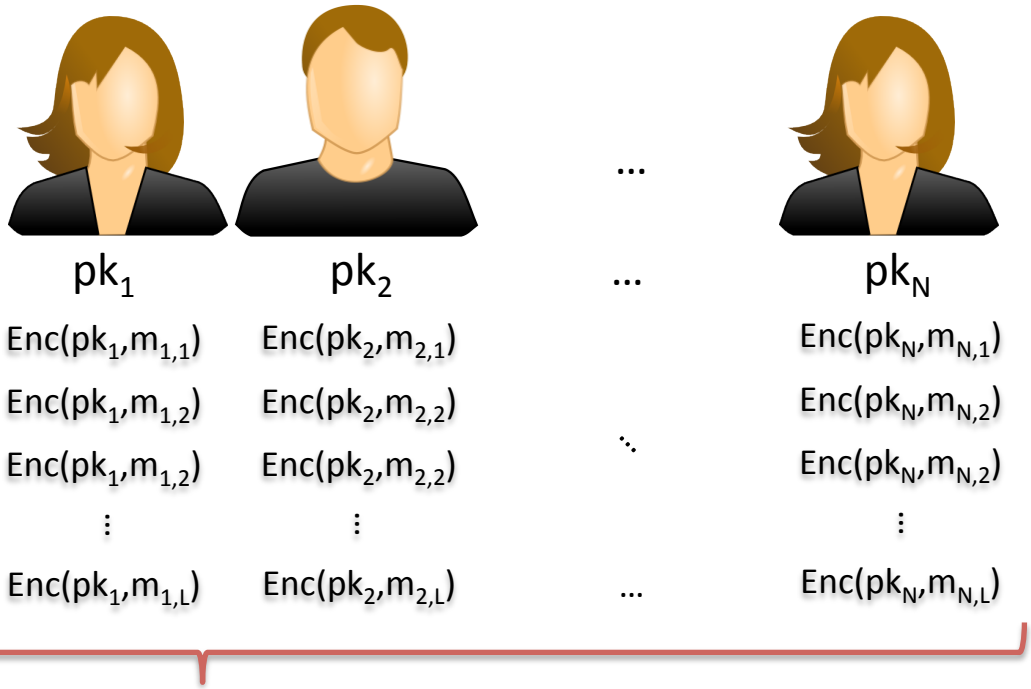
Security proof is **“tight”**, if R(A) has (about) the same **running time** and **success probability** as A

Secure Public-Key Encryption

- Classical security models for public-key encryption:
  - **IND-CPA** security
  - **IND-CCA** security
- Many schemes with tight security proof
- Note that these models consider
  - only **one pk**, and
  - only **one ciphertext**
  - **"(1,1)-security"**



Public-Key Encryption in the Multi-User Setting



In reality:

- **N** public keys
- **L** ciphertexts per public key
- **"(N,L)-security"**

(1,1)-security ⇒ (N,L)-security
[Bellare, Boldyreva, Micali '00]

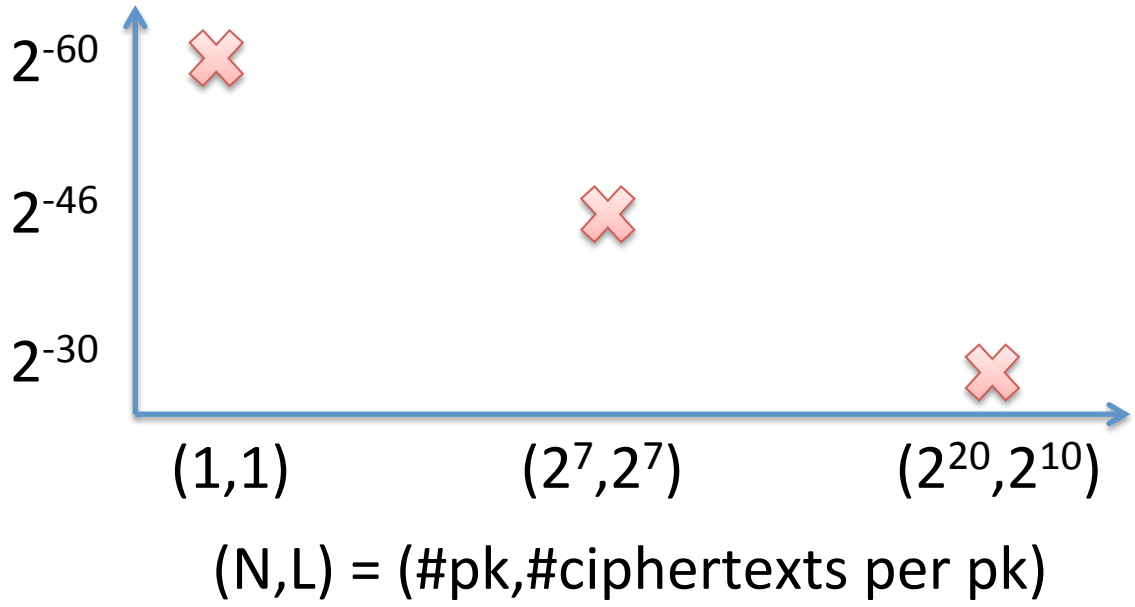
But the reduction is **not tight**
(loses a factor of **N·L** of success probability)

An Example

(Following [Bellare, Boldyreva, Micali`00])

- Assume an encryption scheme with (1,1)-security proof

Bound on **attacker's success probability** implied by choice of parameters and (1,1)-security proof



Proven **security level decreases** with increasing number of users and ciphertexts

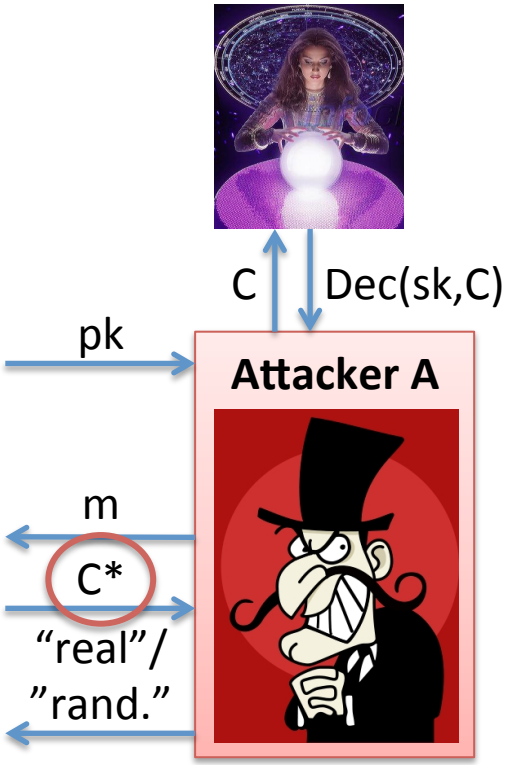
Can we avoid this security loss?

- Trivial solutions:
  - based on **non-standard / parametrized** complexity assumptions
  - in the **Random Oracle Model**
- Bellare, Boldyreva, Micali (Eurocrypt '00):
  - ElGamal is **tightly** (N,L)-IND-CPA secure
  - Cramer-Shoup is **tightly** (N,1)-IND-CCA secure

Our goal: Construct a public-key encryption scheme with
- **tight (N,L)-IND-CCA** security proof
- in the **standard model**
- based on a **standard assumption**

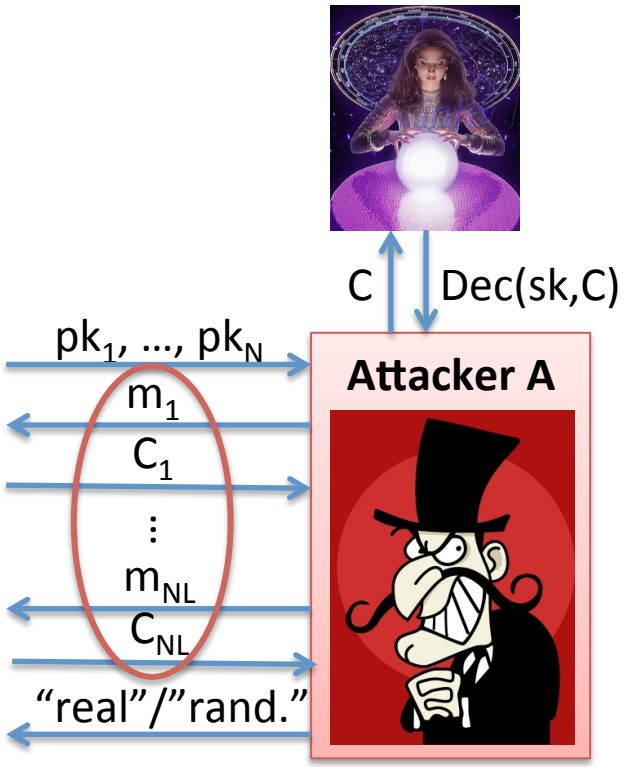
The Difficulty of Tight IND-CCA Security in the Multi-User Setting

- Known techniques exploit that there is **only one** challenge-ciphertext, for instance:
  - "Naor-Yung paradigm" [NY'90] with **one-time** simulation-sound NIZK
  - All-but-**one** simulations (e.g. ABO lossy TDFs [PW'08])
  - …

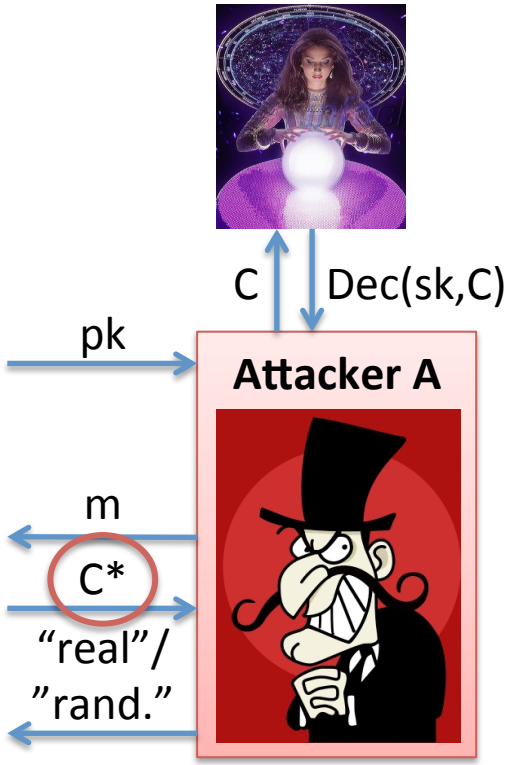


(1,1)-IND-CCA Security Experiment

The Difficulty of Tight IND-CCA Security in the Multi-User Setting

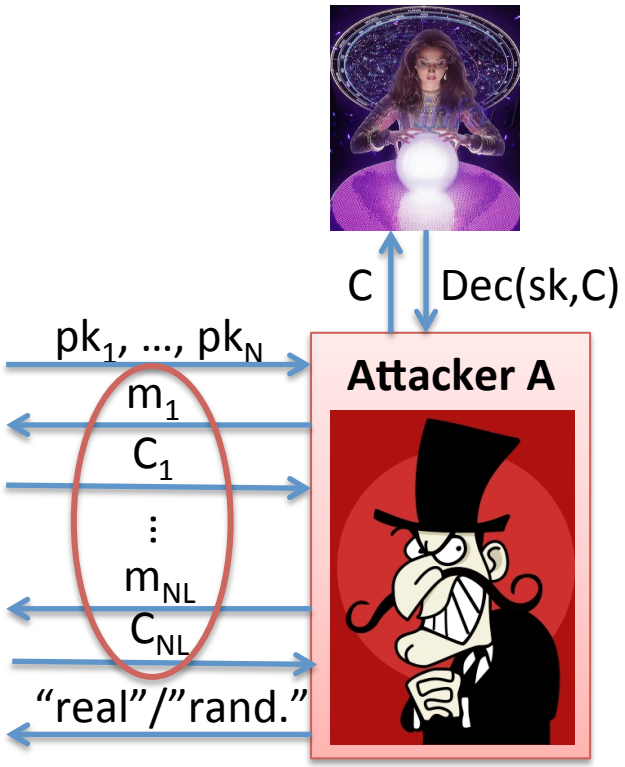


(N,L)-IND-CCA Security Experiment



(1,1)-IND-CCA Security Experiment

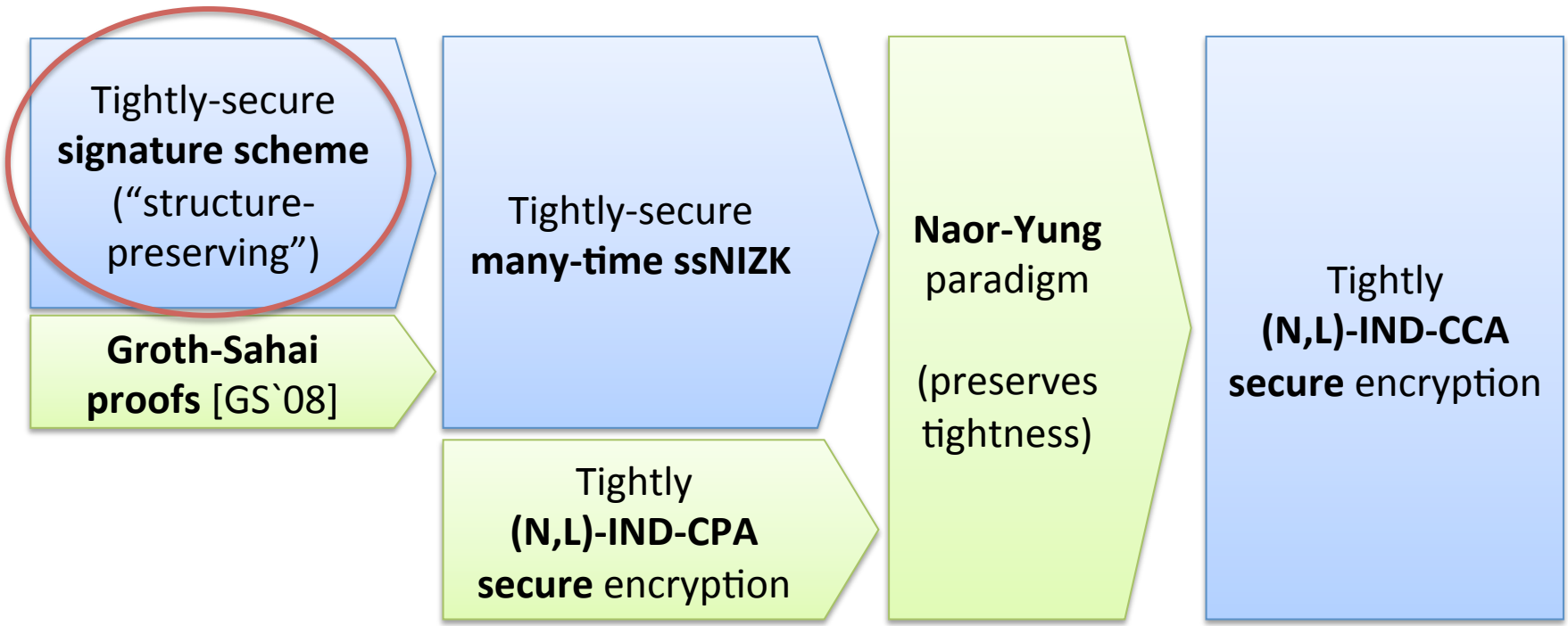
The Difficulty of Tight IND-CCA Security in the Multi-User Setting



(N,L)-IND-CCA Security Experiment

- Known techniques **not immediately** applicable
- Can we adopt them to the multi-user setting?
  - "Naor-Yung paradigm" uses **one-time** ssNIZKs
    - Do **many-time** ssNIZKs help?
    - Can we construct them, with **tight** security proof?

Our Approach



- New constructions
- Known concepts
- All building blocks based on **DLIN** in groups with symmetric pairing

Structure-Preserving Signatures (SPS)

- "Structure-preserving":
  - Public-keys, messages, and signatures are **group elements** (in bilinear group setting)
  - Signature verification checks **conjunction of pairing product equations (PPE)**
- Blend nicely with **Groth-Sahai proofs** [GS'08]
  - Useful tool for **efficient cryptographic constructions**
- **No** known SPS with tight reduction to standard assumption

One-time SPS Scheme

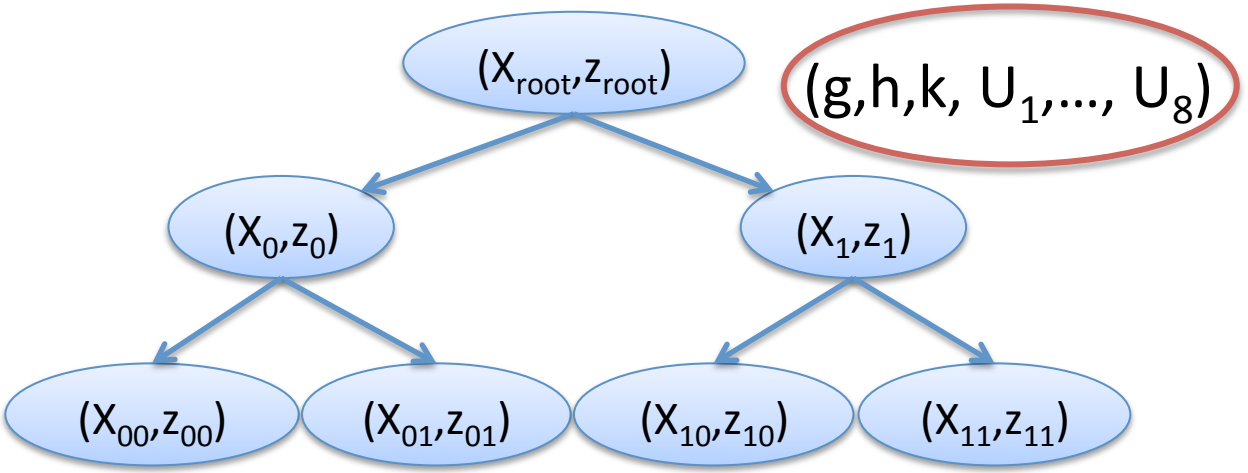
- Let G be a group with pairing $e : G \times G \rightarrow G_T$
- Let $E((a,b,c),d) := (e(a,d),e(b,d),e(c,d))$

- Signature scheme with message space G^n
- $pk = (g,h,k, U_1,\ldots, U_n, X, z)$ where
  - $g,h,k,z \leftarrow G$
  - $U_i = (g^{ui},h^{vi},k^{ui+vi}) \in G^3$ and $X = (g^x,h^y,k^{x+y}) \in G^3$
- To sign $(m_1,\ldots,m_n) \in G^n$, compute $\sigma = (s,t)$ with

$$\prod_{i\in[n]} E(U_i,m_i) \cdot E((g,1,k),s) \cdot E((1,h,k),t) = E(X,z)$$

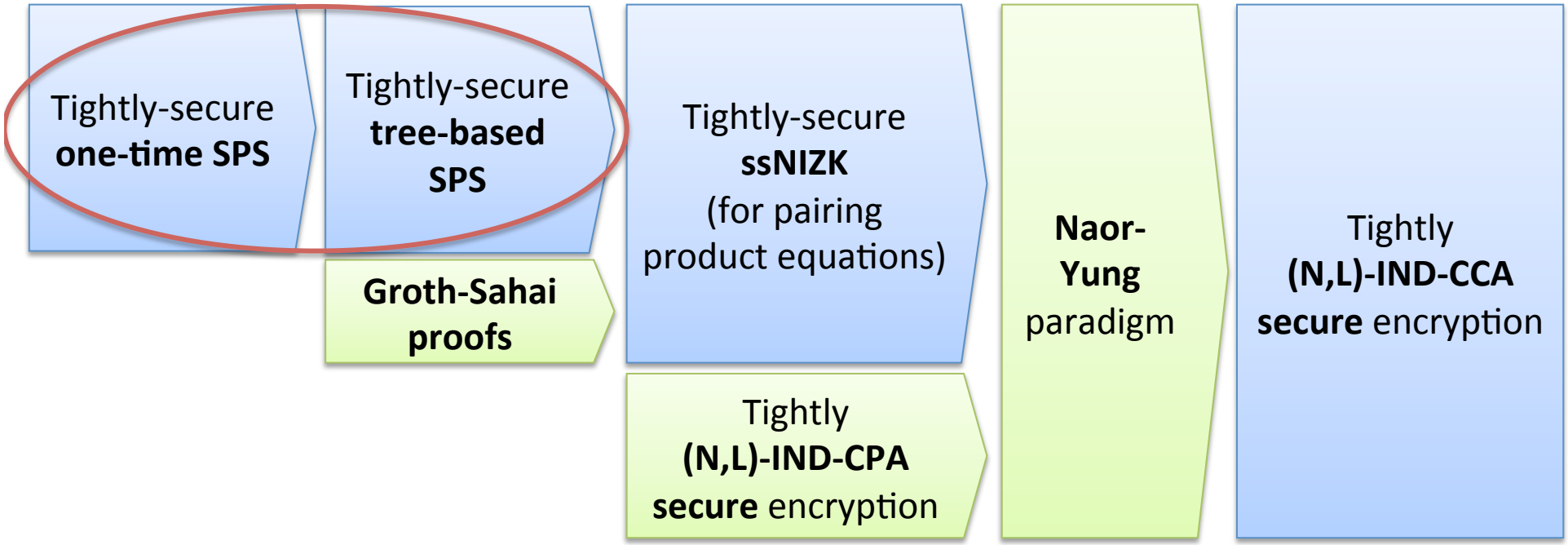
Tightly secure (EUF-1-naCMA) under DLIN

Tightly Secure Tree-based Signature



- Assign **fresh** $(X,z) \in G^3 \times G$ to each node
- Fix $(g,h,k, U_1,\ldots, U_8)$ for **whole tree**
- Intuition: each node assigned with **pk of one-time sig**
  - E.g., node (X_0,z_0) with $pk_0 = (g,h,k, U_1,\ldots, U_8,X_0,z_0)$
- Gives rise to "Merkle tree" scheme [Mer'79]

Summary



- New constructions
- Known concepts
- All building blocks based on **DLIN** in groups with symmetric pairing

Open Problems

- **Further applications** to tightly-secure constructions?
  - ssNIZK + [Camenisch, Chandran, Shoup'09] = **tight** KDM-CCA-secure encryption
- **Shorter tree-based SPS?**
  - Abe et al. (Asiacrypt `12): more efficient **one-time** SPS ⇒more efficient tree-based SPS
- SPS with
  - **Short** signatures and public keys
  - **tight security** from simple complexity assumption?