Randomness-Extraction Key-Derivation Approach to CCA2-Secure Hybrid Encryption

(Crypto ’08– rump session)

Moti Yung, Google Inc.

Joint work with **Aike Kiltz, Krzyzstof Pietrzak** and **Martin Stam**

The Olympic Spirit ←→ Cryptography

(1) Modern Era Olympics has classical and modern traditions (like cryptography)

(2) Olympic results are improved constantly (well… as in crypto)

(3) Results have participation/ artistic value (the spirit, the techniques, the elegance)

(4) Results have athletic values: improved performance, etc.

The problem: chosen ciphertext security

Definitely a classical problem

Was open for a while

Feasibility in PKC [NY89] CCA1, CCA2: [RS90, DDN90,S,L,…]

Practical systems breakthrough: CS98, DDH based and hash-proof system (HPS) concept.

CS: also a Hybrid Encryption (perhaps the most used in PK encryption)

Pairing-based [CHK]

Hybrid: [KD04]

Obsession

Hybrid:

– KEM- public-key encapsulation

–Then: Symmetric key data encryption (DEM)

Obsession:

Hybrid Encryption in the efficient case uses “integrity check” on the symmetric and asymmetric levels (to some extent). Can we have only one? …(one world one dream…..and in the process have some achievements in the CCA-Olympiad).

What we have:

Tool one: Hash Proof Systems [CS]:

- 1-universal-HPS → CCA1
- 2-universal-HPS → CCA2

Tool two: (one time) Authenticated Symmetric Encryption

Tool Three: Randomness extractor: 4-wise independent based randomness extractor (4 deg polynomial over the field, hashing enough).

Then Design:

Take a CCA1 scheme based on 1-u-HPS

Add 4-wise hash randomness extractor to the public key

Use extracted key as the one to the Authenticated Symmetric Enc.

Why and What?

Theorem: Applying our randomness extractor as/ on top of the key derivation over the derived key:

transforms 1-u-HPS → 2-u-HPS

(giving a mechanism to design CCA2)

Basic idea: We have CCA1 system that is secure when no after challenge probing,

Then: Modified KEM and/or modified DEM at the after challenge stage will fail, since extractor "throws the symmetric key to a random location." The system will check only the symmetric auth. Enc. For integrity!

→ New Systems Based on QR, Paillier, DLOG where we have CCA1 easily we can have CCA2

Example

Take Damgaard-ElGamal system from 90.

Use it in the KEM/DEM paradign

Key Gen

- X=g1^{x1} * g2^{x2}
- ADD: k: key for the hash (extractor)
- X and k are public key, where xa and x2 are secret.

System (cont.)

Encryption (Hybrid)

- Choose r at random c1=g1^{r}, c2=g2^{r}
- K=Hk(X^{r}), c3= Auth-Enc(k,m)
- Send c1, c2, c3

Decryption:

J=Hk(c1^(x1)* c2^{x2})

Auth-decr(C3) check and if fails return nothing

otherwise return authenticated message.

→

Theorem: This is CCA2 secure under DDH.

Note: other (even recent) variants of this system required much stronger assumption just to get CCA1

What we get

A new approach and proof methodology to derive CCA2-secure PK system efficiently (we do not have too many methodologies of this kind).

New Systems: Viewing various system (even old one) as Hybrid makes them CCA1 and then with our methodology transforms to CCA2 (new PK systems with strong security assurance).

It is a new way to design CCA2: the resulting systems are even efficient when starting from an efficient system (always save UOWHF computations and the KEM check, and even at times also, say, an exponentiation).

Citius, Altius, Fortius,