Public-key encryption with non-interactive opening

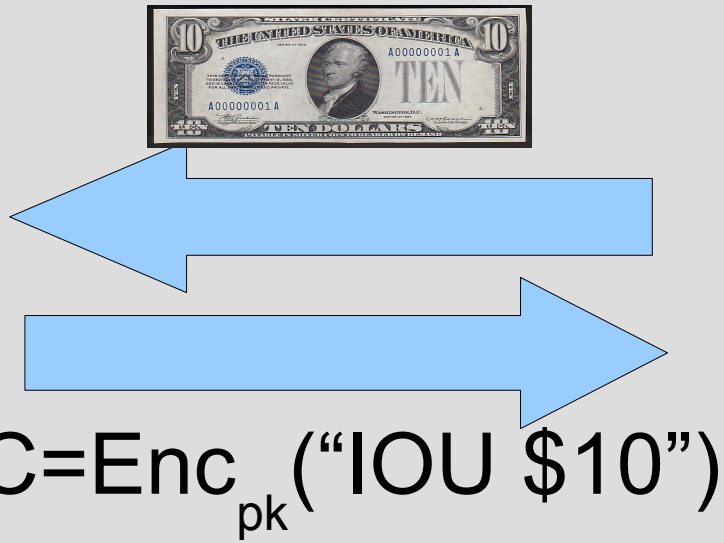
Ivan Damgård (BRICS)
Dennis Hofheinz (CWI)
Eike Kiltz (CWI)
Rune Thorbek (BRICS)

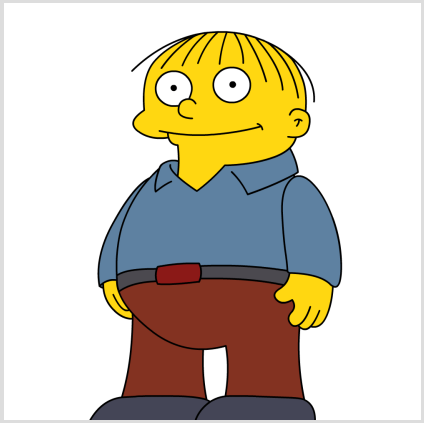
Cryptographic tools

- Cryptography essentially a toolbox
- Here's one tool: public key encryption
  - Everyone can encrypt (public encryption key)
  - Only designated receiver can decrypt (secret key)
- This talk: “trustworthy public key encryption”
  - Ciphertexts can be opened **publicly**
  - Content of ciphertext provable/verifiable

Toy scenario

- Suppose Ralph lends money to Snake



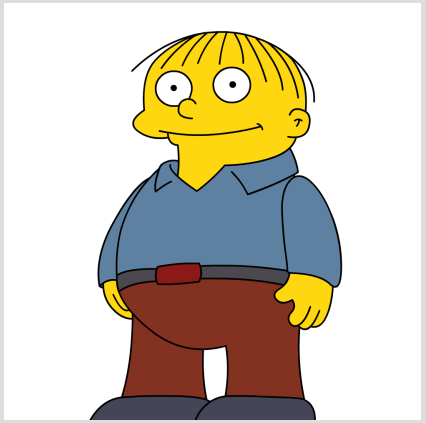


- Suppose Snake doesn’t want to pay later on

Toy scenario

"Make Snake pay!"

$C=Enc_{pk}(\text{“IOU \$10”})$



- How can Ralph convince his daddy/the police?

Toy scenario

- How can Ralph convince his daddy/the police?
  - Solution 1: Let Snake **sign** the IOU
    - Requires public verification key for Snake
    - Bad if Ralph has many clients
  - Solution 2: Ralph publicizes his secret key
    - Works **once** (and maybe not even that)
  - Solution 3: Enable Ralph to **open C publicly**
    - Assume C is broadcasted in the first place
    - Other ciphertexts to be kept secure

Toy scenario

- How can Ralph convince his daddy/the police?
  - Solution 1: Let Snake **sign** the IOU
    - Requires public verification key for Snake
    - Bad if Ralph has many clients
  - Solution 2: Ralph publicizes his secret key
    - Works **once** (and maybe not even that)
  - Solution 3: Enable Ralph to **open C publicly**
    - Assume C is broadcasted in the first place
    - Other ciphertexts to be kept secure

(Public key) encryption with non-interactive opening

- Goal: enable **receiver** to open ciphertext publicly
  - Easy in zero-knowledge, but **inefficient**
  - Detail that makes it hard: **invalid** ciphertexts
  - Remark: easy for **sender** to open ciphertext publicly
    - Release random coins used during encryption
    - Note: decryption does not necessarily retrieve these coins
    - Correctness of scheme guarantees unique message
- Syntax: $\text{Prove}_{sk}(C)=(m,\pi)$ and $\text{Verify}_{pk}(C,\pi,m)$

Almost-solution based on IBE

- Identity-based encryption scheme:
  - Many identities id, encryption is $Enc_{IBE}(pk,id,m)$
  - Decryption key usk_{id} for id derived from master sk
  - Idea: no public key infrastructure needed
    - One master public key, no individual user public keys
    - Only individual user secret keys
- Efficient pairing-based IBE schemes known

Almost-solution based on IBE

- Solution implicit in [Damgård, Thorbek 2006]:
  - Interpret IBE scheme as PKE
  - PKE public/secret key = IBE master public/secret key
  - $Enc_{PKE}(pk,m) = (id, Enc_{IBE}(pk,id,m))$ (id random)
  - $Prove_{sk}((id,C) ,m) = usk_{id}$
- Problem: what if twice-used id opened?
  - Evil party can force publication of usk_{id} for “honest” id

Our new results

- Full solution (generic IBE → PKE-NO transform)
  - One-time signatures → no id collisions [CHK]
- More efficient solution (optimized scheme)
- Equivalence of PKE-NO definitions
  - Game-based = universally composable definition
  - PKE-NO behaves well under composition

Our new results

- Full solution (generic IBE → PKE-NO transform)
  - One-time signatures → no id collisions [CHK]
- More efficient solution (optimized scheme)
- Equivalence of two PKE-NO definitions
  - Game-based = universally composable definition
  - PKE-NO behaves well under composition

Our new results

- Almost-solution (IBE → PKE-NO transform):
  - Interpret IBE scheme as PKE
  - PKE secret key = IBE master secret key
  - $Enc_{PKE}(pk,m) = (id, Enc_{IBE}(pk,id,m))$ (id random)
  - $Prove_{sk}((id,C) ,m) = usk_{id}$

Our new results

- Our solution ([CHK] conversion + Prove/Verify):
  - Interpret IBE scheme as PKE
  - PKE secret key = IBE master secret key
  - $Enc_{PKE}(pk,m) = (vk, Enc_{IBE}(pk,vk,m), \sigma)$
  - $Prove_{sk}((id,C) ,m) = usk_{id}$

Signs ciphertext with signing key that belongs to vk

Remarks

- Required from IBE scheme: usk_{id} verifiable
  - Testing on random encryptions not enough
- For concrete IBE scheme, optimization possible
- Recent, completely different way of encrypting:
  - Decryption retrieves random coins from encryption
  - “Lossy trapdoor functions” [PW07]
  - Not (yet) as efficient as IBE conversion