REACT: Rapid Enhanced-security Asymmetric Cryptosystems Transform

Cryptography Workshop '2001
Monte Verita, Switzerland, March 2001

Tatsuaki Okamoto
NTT
Yokosuka - Japan

David Pointcheval
ENS - CNRS
Paris - France

David.Pointcheval@ens.fr
http://www.di.ens.fr/users/pointche

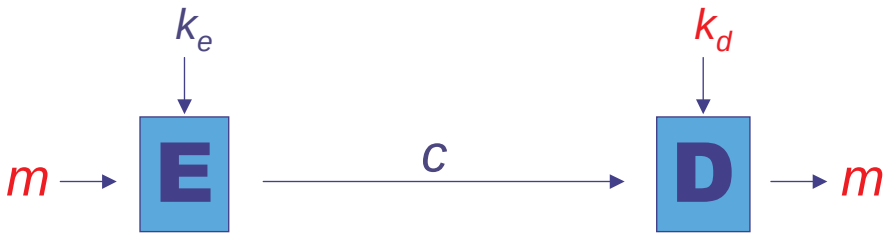
Overview

- Introduction to Encryption
- Previous conversions
- REACT: the new conversion
  - Description
  - Security Result
  - Sketch of the Proof
- Conclusion

Asymmetric Encryption

Encryption Algorithm **E** Encryption key k_e

Decryption Algorithm **D** Decryption key k_d



Security: it is impossible to get back m just from c, k_e, **E** and **D** (without k_d)

Security Notions

Depending on the security concerns, one defines

- the goals that an adversary may would like to reach
- the means/information available to the adversary

Goals of an Adversary

- One-Wayness
- Semantic Security
  (Indistinguishability):

  no polynomial adversary can learn any information about the plaintext from the ciphertext and public data (but the size)

Kinds of Attacks

- Chosen Plaintext: *(basic scenario)*

  in the public-key setting, any adversary can get the encryption of any plaintext of her choice

  Basic security level: OW-CPA

- Chosen Ciphertext (adaptively):

  the adversary has furthermore access to a decryption oracle which decrypts any ciphertext of her choice (excepted the specific challenge!)

  Highest security level: IND-CCA

Example I: RSA Encryption

- $n = pq$, product of large primes
- e, exponent relatively prime to $\varphi(n) = (p-1)(q-1)$
- n, e : **public** key
- $d = e^{-1} \bmod \varphi(n)$: **secret** key

public $\quad \mathbf{E}(m) = m^e \bmod n$

secret $\quad \mathbf{D}(c) = c^d \bmod n$

OW-CPA = RSA problem

Example II: El Gamal Encryption

- **G** = (<g>, $\times$) group of order q
- x : **secret** key
- $y=g^x$: **public** key

public $\quad \mathbf{E}(m) = (g^a, y^a m) \rightarrow (c, d)$

secret $\quad \mathbf{D}(c, d) = d / c^x$

OW-CPA = CDH problem
IND-CPA = DDH problem

Generic Conversions

- Any trapdoor one-way (injective) function leads to a OW-CPA cryptosystem
- But OW-CPA not enough
- How to reach IND-CCA ?

  $\Rightarrow$ generic conversions from OW-CPA to IND-CCA

$(\mathcal{E},\mathcal{D})$ is assumed to be weakly secure and one designs a secure $(\mathbf{E},\mathbf{D})$

Previous Conversions: OAEP

Bellare-Rogaway (at EC '94) proposed the **Optimal Asymmetric Encryption Padding**, a very efficient conversion

It was believed to provide a conversion of any *trapdoor one-way* ***permutation*** into IND-CCA

Actually, it just provides a conversion of any *trapdoor* ***partially*** *one-way* ***permutation***

Anyway, RSA is the sole application
RSA-OAEP: **IND-CCA=RSA** [*FOPS'00*]

Recent Generic Conversions

Fujisaki-Okamoto (PKC '99)
from ***IND-CPA into IND-CCA***

Fujisaki-Okamoto (Crypto '99)
and Pointcheval (PKC '00)
from ***OW-CPA into IND-CCA***

Efficiency:

- possible hybridity
- optimal encryption (just few more hashings)
- **non-optimal decryption** (1 re-encryption)

New Conversion: REACT

PK-Cryptosystem $(\mathcal{E},\mathcal{D})$: $\mathcal{M} \times \mathcal{R} \rightarrow C$
Block-Cipher $\mathbf{E}_k, \mathbf{D}_k$: $\{0,1\}^\lambda \rightarrow \{0,1\}^\lambda$
Hash functions G, H

$\mathbf{E}(m,r||s) =$ $a = \mathcal{E}(r, s)$ with $r \in \mathcal{M}$, $s \in \mathcal{R}$
$b = \mathbf{E}_k(m)$ where $k = \mathrm{G}(r)$
$c = \mathrm{H}(m,r,a,b)$

$\mathbf{D}(a,b,c)$: Compute $r = \mathcal{D}(a)$ and $k = \mathrm{G}(r)$
extract $m = \mathbf{D}_k(b)$
if $c = \mathrm{H}(m,r,a,b)$ and $r \in \mathcal{M}$ then output m

New Conversion: REACT

Efficiency:

- optimal encryption (just 2 more hashings)
- **optimal decryption** (just 2 more hashings)

Security: conversion

- in the random oracle model
- of any ***OW-PCA cryptosystem*** into an IND-CCA cryptosystem

A New Attack: PCA

- Plaintext Checking Attack: the adversary
  - can get the encryption of any plaintext of her choice (by encrypting it by herself)
  - has furthermore access to an oracle which, on input a pair (m,c), answers whether c encrypts m, or not

RSA function: OW-PCA = RSA

El Gamal: OW-PCA = GDH

Symmetric Encryption Scheme

One just needs a symmetric encryption $(\mathbf{E}_k, \mathbf{D}_k)$ semantically secure against passive attacks:

- One-Time Pad: perfectly secure ($\mathrm{Adv}^{\mathbf{E}} = 0$)
- Any classical scheme (DES, IDEA, AES,…)

  $\mathrm{Adv}^{\mathbf{E}} = \nu$ (very small)

Security Result

$$G : \mathcal{M} \rightarrow \{0,1\}^{\ell_G} \qquad H : \{0,1\}^* \rightarrow \{0,1\}^{\ell_H}$$

If an adversary $\mathcal{A}$ against IND-CCA reaches an advantage $\mathrm{Adv}^{\mathcal{A}} > \mathrm{Adv}^{\mathbf{E}}$ after q_G, q_H and $q_{\mathbf{D}}$ queries to G, H and **D** resp. one can break the OW-PCA of $(\mathcal{E}, \mathcal{D})$ with probability greater than

$$\frac{\mathrm{Adv}^{\mathcal{A}} - \mathrm{Adv}^{\mathbf{E}}}{2} - \frac{q_{\mathbf{D}}}{2^{\ell_H}}$$

Semantic Security (OTP)

Given (a,b,c) such that
$a = \mathcal{E}(r,s)$,
$k = \mathrm{G}(r)$, $b = k \oplus m$,
$c = \mathrm{H}(m,r,a,b)$

$\mathbf{E}(m,r||s) = a = \mathcal{E}(r,s)$ with $r \in \mathcal{M}$, $s \in \mathcal{R}$
$b = \mathbf{E}_k(m)$ where $k = \mathrm{G}(r)$
$c = \mathrm{H}(m,r,a,b)$
$\mathbf{D}(a,b,c)$ = Compute $r = \mathcal{D}(a)$ and $k = \mathrm{G}(r)$
extract $m = \mathbf{D}_k(b)$
if $c = \mathrm{H}(m,r,a,b)$ and $r \in \mathcal{M}$ then output m

In order to guess the bit d such that $m = m_d$ an adversary has to ask either

- r to G to get k (and check b)
- (m_0,r,a,b) or (m_1,r,a,b) to H (and check c)

because of the randomness of G and H

Semantic Security (OTP Cont'd)

Probability that r $(= \mathcal{D}(a))$ has been asked to G or H greater than $\mathrm{Adv}^{\mathcal{A}}/2$

Simply find the good one with the PC-oracle, into all the G queries and the H queries
$\Rightarrow q_{\mathrm{G}} + q_{\mathrm{H}}$ queries to the PC-oracle

Plaintext Extractor

$\mathbf{E}(m,r||s) = a = \mathcal{E}(r, s)$ with $r \in \mathcal{M}$, $s \in \mathcal{R}$
$b = \mathbf{E}_k(m)$ where $k = \mathrm{G}(r)$
$c = \mathrm{H}(m,r,a,b)$
$\mathbf{D}(a,b,c)$ = Compute $r = \mathcal{D}(a)$ and $k = \mathrm{G}(r)$
extract $m = \mathbf{D}_k(b)$
if $c = \mathrm{H}(m,r,a,b)$ and $r \in \mathcal{M}$ then output m

$C' = (a',b',c')$
valid ciphertext

$\Rightarrow$ one has asked for (m',r',a',b') to H
to get a valid c' or has guessed it,
(but with probability less than $1/2^{\ell_H}$)

$\Rightarrow$ simply looks into the H queries

Correct extraction with probability greater than $1 - 1/2^{\ell_H}$

Applications

- RSA: IND-CCA=RSA
  alternative to RSA-OAEP
- El Gamal: IND-CCA=GDH
  Rk: On Elliptic Curves = PSEC-3
- REACT-El Gamal is the most efficient El Gamal variant:
  - 1 exp./Enc + 2 hashings
  - 2 exp./Dec + 2 hashings

Conclusion on REACT

- OW-PCA into IND-CCA
  - $\Rightarrow$ the best security level
- The cost is just:
  - 2 more hashings in encryption/decryption
  - $\Rightarrow$ almost optimal
- Improved Efficiency:
  - Can integrate symmetric encryption
  - To encrypt many messages:
    $a = \mathcal{E}(r, s)$ and $k = \mathrm{G}(r)$
    $b_i = \mathbf{E}_k(m_i)$ and $c_i = \mathrm{H}(m_i, r, a, b_i)$