Public Key Cryptography

PKC ‘ 2000

18-20 january 2000 - Melbourne - Australia

Chosen-Ciphertext Security
for any One-Way Cryptosystem

David Pointcheval
Département d ’Informatique
ENS - CNRS

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

Overview

- Introduction
- Previous Conversions
- The New Conversion
- Security Properties
- Some Applications
- Conclusion

Introduction

Public-Key Cryptosystems = Confidentiality

- One-wayness:

  nobody can recover the whole plaintext from the ciphertext and public data

  - RSA: OW = e-th root modulo $n=pq$
  - Rabin: OW = SQRT = factorization
  - El Gamal: OW = Diffie-Hellman problem

Better security?

- Perfect Security:

  the ciphertext and public data do not reveal any information about the plaintext (but maybe the size)

- Semantic Security (Indistinguishability):

  no polynomial adversary can learn any information about the plaintext from the ciphertext and public data

Kinds of Attacks

- Chosen Plaintext: *(basic scenario)*

  in the public-key setting, any adversary can get the encryption of a plaintext of her choice (by encrypting it by herself)

- Chosen Ciphertext (adaptively):

  the adversary has furthermore access to a decryption oracle which decrypts any message of her choice, but the specific challenge

Required Security

- OW-CPA: *(basic level of security)*
  - enough in some scenarios
  - not enough in many others
- CC-Attacks easy to perform

  $\Rightarrow$ attack to be made unuseful

- Plaintext-space often limited

  (“sell” - “buy” -- “yes” - “no” -- …)

  $\Rightarrow$ IND very often required

Main Security Notions

- OW-CPA: *(the weakest)*

$$\Pr_{m,\omega}\left[A(y) = m \middle| y = \mathbf{E}(m;\omega)\right] = \text{Succ negligible}$$

- IND-CCA: *(the strongest* - BDPR C '98*)*

$$\Pr_{\omega,b}\left[A_2^{\mathbf{D}}(m_0, m_1, c, s) = b \middle| \begin{array}{c}(m_0, m_1, s) \leftarrow A_1^{\mathbf{D}}(\mathrm{pk}) \\ c \leftarrow \mathbf{E}(m_b, \omega)\end{array}\right] - \frac{1}{2}$$

= Adv negligible

Cryptosystems: OW-CPA

Many Candidates:

- RSA: modular e-th root (RSA problem)
  - Original RSA
  - Paillier: Higher Residuosity Classes
  - Pointcheval: Dependent-RSA / RSA
- Rabin: Factorization $n=pq$
- El Gamal: Computational Diffie-Hellman
- Okamoto-Uchiyama: Factorization $n=p^2q$

Cryptosystems: IND-CCA

Few Candidates:

- Cramer-Shoup: Decisional DH
- with Random Oracle:
  - OAEP-RSA: RSA
  - Paillier-Pointcheval: Higher Residuosity
  - EPOC: Factorization
  - DH-RSA: Dependent-RSA / RSA

Observations

- Many OW-CPA cryptosystems
- Few IND-CCA ones
  - worse security (only decisional problems)
  - specific hard proofs of security
  - efficiency not optimal

$\Rightarrow$ necessity of optimal conversions
from OW-CPA to IND-CCA

Previous Conversions (1/2)

- OAEP (Bellare-Rogaway EC '94)

  optimal conversion of
  any ***trapdoor one-way permutation***
  into an IND-CCA cryptosystem

  Application: RSA
  (the sole candidate as
  trapdoor one-way permutation!)

Previous Conversions (2/2)

- Fujisaki-Okamoto (PKC '99)

  conversion of
  any ***IND-CPA cryptosystem***
  into an IND-CCA cryptosystem

  Drawback: security relative to decisional problems (D-DH, Higher Residuosity, …)

  Improvement: Crypto '99
  similar result as the present work
  (both done independently)

New Conversion (1/2)

- Setting:
  - Original OW-CPA scheme:

    E: X×Y → Z and **D**: Z → X
  - Security Parameter: $k = k_0 + k_1$
    - s k_0: size of the plaintext
    - s k_1: IND-parameter
    - s Y is the random set assumed large enough:
      |Y| = CCA-parameter
  - Hash Functions:

    $\mathbf{H}: \{0,1\}^k \rightarrow$ Y and $\mathbf{G}$: X $\rightarrow \{0,1\}^k$

New Conversion (2/2)

- New scheme:
  - Encryption of $m \in \{0,1\}^{k_0}$

    $r \in$ X and $s \in \{0,1\}^{k_1}$ randomly chosen

    $a = \mathrm{E}(r, \mathbf{H}(m||s))$
    $b = (m||s) \oplus \mathbf{G}(r)$
    } **E'**$(m;r||s) = (a,b)$
  - Decryption of (a, b)

    $r = \mathbf{D}(a)$
    $\mathrm{M} = b \oplus \mathbf{G}(r)$
    } if $a = \mathrm{E}(r, \mathbf{H}(\mathrm{M}))$
    $\rightarrow$ **D'**$(a,b) = [\mathrm{M}]_{k_0}$

Semantic Security

Given $a = \mathrm{E}(r, \mathbf{H}(m\|s))$ and $b = (m\|s) \oplus \mathbf{G}(r)$

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

- $(m\|s)$ to $\mathbf{H}$ (and check $a = \mathrm{E}(r, \mathbf{H}(m\|s)))$
  probability less than $q_\mathbf{H}/2^{k_1}$
- r to $\mathbf{G}$ (and extract $(m\|s)$ from b)
  because of the randomness of $\mathbf{G}$

Probability that r has been asked to $\mathbf{G}$ greater than Adv - $q_\mathbf{H}/2^{k_1}$

Plaintext Extractor

More than CCA, this scheme is Plaintext-Aware (Bellare-Rogaway EC ‘94)

(a,b) valid $\Leftrightarrow (\exists r)\ a = \mathrm{E}(r, \mathbf{H}(b \oplus \mathbf{G}(r)))$

$\Rightarrow$ one has to have asked both

- r to $\mathbf{G}$ (but with probability less than $1/2^{k_1}$)
- $\mathrm{M}=(m\|s)=b \oplus \mathbf{G}(r)$ to $\mathbf{H}$
  (but with probability less than 1/|Y|)

Probability that the plaintext can be extracted from queries asked to $\mathbf{G}$ and $\mathbf{H}$ greater than $1 - 1/|\mathbf{Y}| - 1/2^{k_1}$

CCA Security

After q_D queries to the decryption oracle

- all the decryptions are correctly simulated with probability greater than

$$(1 - 1/|\mathbf{Y}| - 1/2^{k_1})^{q_D} \geq 1 - q_D / |\mathbf{Y}| - q_D / 2^{k_1}$$

- r has been asked to $\mathbf{G}$ with probability greater than

$$\mathrm{Adv} - \frac{q_D + q_H}{2^{k_1}} - \frac{q_D}{|\mathbf{Y}|}$$

General Case

A, adversary against IND-CCA,
after q_D queries to the decryption oracle,
but q_G and q_H queries to oracles $\mathbf{G}$ and $\mathbf{H}$

by picking at random an element in the list of queries asked to $\mathbf{G}$, one breaks OW-CPA with probability greater than

$$\frac{1}{q_G} \times \left(\mathrm{Adv} - \frac{q_D + q_H}{2^{k_1}} - \frac{q_D}{|\mathbf{Y}|} \right)$$

Random Self-Reducible Case

A is run twice (Shoup EC '97):

- once to output a list of candidates
  (all the queries asked to **G**)
- then to check if one is the right one
  (on a randomly-reduced instance)

after 21t/δ expected replays of ***A***, one breaks OW-CPA with error probability less than 2^{-t}

where

$$\delta = \mathrm{Adv} - \frac{q_D + q_H}{2^{k_1}} - \frac{q_D}{|\mathbf{Y}|} \approx \mathrm{Adv}$$

Example: El Gamal

$\Gamma = \langle g \rangle$ of order q

$\mathbf{H} : \{0,1\}^k \to \mathbf{Z}_q$ and $\mathbf{G} : \Gamma \to \{0,1\}^k$

Secret key : $x \in \mathbf{Z}_q$

Public Key : $y = g^x$

$r \in \Gamma$ and $s \in \{0,1\}^{k_1}$

$d = \mathbf{H}(m \| s)$

$$\mathrm{Enc}(m, r \| s) = \begin{cases} a = g^d \\ b = y^d r \\ c = (m \| s) \oplus \mathbf{G}(r) \end{cases}$$

$$\mathrm{Dec}(a,b,c) = \begin{cases} r = b / a^x \\ t = c \oplus \mathbf{G}(r) \\ \text{if } a = g^{\mathbf{H}(t)} \text{ then } m = [t]_{k_0} \end{cases}$$

IND-CCA equivalent to C-DH

after 30t/Adv expected replays of ***A***,

one breaks C-DH with error prob. less than 2^{-t}

Properties

Our new EG-scheme:

- based on C-DH problem and Random Oracle
- 2 exponentiations (encryption and decryption)

Comparison with previous El Gamal variants:

- Tsiounis-Yung (PKC '98)
  D-DH -- RO and non-standard assumption
  not efficient: 3 exp./Enc - 3 exp./Dec
- Shoup-Gennaro (EC '98)
  D-DH -- RO -- 5 exp./Enc - 7 exp./Dec
- Cramer-Shoup (Crypto '98)
  D-DH -- Standard Model -- also very slow

Easy Verifiable Case

A is run once, and then one checks
if one candidate in the list of queries
asked to G is correct

after 1 execution of ***A***,
one breaks OW-CPA
with probability greater than

$$\mathrm{Adv} - \frac{q_D + q_H}{2^{k_1}} - \frac{q_D}{|\mathbf{Y}|} \approx \mathrm{Adv}$$

Example: Okamoto-Uchiyama

$n = p^2 q,\ g \in \mathbf{Z}_n^*$ and $h = g^n \bmod n$
with $g_p = g^{p-1} \bmod p^2$ of order p

$\mathbf{H} : \{0,1\}^k \to \mathbf{Z}_n$
and $\mathbf{G} : \mathbf{Z}_n \to \{0,1\}^k$

$r \in \mathbf{Z}_n$ and $s \in \{0,1\}^{k_1}$

$$\mathrm{Enc}(m, r \| s) = \begin{cases} a = g^r h^{H(m\|s)} \bmod n \\ b = (m \| s) \oplus \mathbf{G}(r) \end{cases}$$

$$\mathrm{Dec}(a,b) = \begin{cases} r = L(y_p)/L(g_p) \bmod p \\ t = b \oplus \mathbf{G}(r) \\ \text{if } a = g^r h^{\mathbf{H}(t)} \text{ then } m = [t]_{k_0} \end{cases}$$

IND-CCA equivalent to Factorization

after 1 execution of ***A***, one gets the factorization of n with probability greater than

$$\mathrm{Adv} - (q_{\mathbf{D}} + q_{\mathbf{H}})/2^{k_1} - q_{\mathbf{D}}/|\mathbf{Y}| \approx \mathrm{Adv}$$

Properties

Our new OU-scheme:

- based on Factorization
- and Random Oracle Model

Original Scheme (OU - EC '98):

- OW-CPA = Factorization
- IND-CPA = higher residuosity (only)
- but CCA leads to a total break!

Conclusion

This conversion is

- very general:

  from any OW-CPA scheme

  (any partially trapdoor one-way function)

  one derives an IND-CCA scheme

- very efficient:
  - optimal encryption: + 2 hash and 1 XOR
  - hybrid encryption can be used:

    $b = (m\|s) \oplus \mathbf{G}(r)$, is a one-time pad

    $b = E_K(m\|s)$, using the "session" key K= $\mathbf{G}(r)$