Advanced Cryptography — Final Exam

Serge Vaudenay

26.6.2018

- duration: 3h
- any document allowed
- a pocket calculator is allowed
- communication devices are not allowed
- the exam invigilators will **not** answer any technical question during the exam
- readability and style of writing will be part of the grade

1 Ciphertext Collision in Semantically Secure Cryptosystems

We consider a public-key cryptosystem $(\mathsf{Gen}, \mathcal{M}, \mathsf{Enc}, \mathsf{Dec})$. We assume perfect correctness, i.e. for all s and all $x \in \mathcal{M}$, if $(K_p, K_s) \leftarrow \mathsf{Gen}(1^s)$ then

$$\Pr[\mathsf{Dec}_{K_s}(\mathsf{Enc}_{K_p}(x)) = x] = 1$$

Given a probabilistic polynomial-time adversary $\mathcal{A}$, we consider the following game:

Game $\Gamma_{\mathcal{A}}(s)$:
1: $(K_p, K_s) \leftarrow \mathsf{Gen}(1^s)$
2: $X \leftarrow \mathcal{A}(K_p)$
3: $Y_0 \leftarrow \mathsf{Enc}_{K_p}(X)$
4: $Y_1 \leftarrow \mathsf{Enc}_{K_p}(X)$
5: **return** $1_{Y_0 = Y_1}$

Q.1 Prove that if the cryptosystem is $\mathsf{IND\text{-}CPA}$ secure, then $\Pr[\Gamma_{\mathcal{A}}(s) \to 1]$ is negligible.
Hint: construct an $\mathsf{IND\text{-}CPA}$ adversary with advantage related to $\Pr[\Gamma_{\mathcal{A}}(s) \to 1]$.

2 Non-Malleability in Adaptive Security

We consider a public-key cryptosystem $(\mathsf{Gen}, \mathcal{M}, \mathsf{Enc}, \mathsf{Dec})$. We assume perfect correctness, i.e. for all s and all $x \in \mathcal{M}$, if $(K_p, K_s) \leftarrow \mathsf{Gen}(1^s)$ then

$$\Pr[\mathsf{Dec}_{K_s}(\mathsf{Enc}_{K_p}(x)) = x] = 1$$

Given an adversary in two parts $\mathcal{A} = (\mathcal{A}_1, \mathcal{A}_2)$, a bit $b \in \{0, 1\}$, and the security parameter s, we define the IND-CCA game as follows:

Game $\mathsf{IND\text{-}CCA}^b_{\mathcal{A}}(s)$
$(K_p, K_s) \leftarrow \mathsf{Gen}(1^s)$
$(X_0, X_1, \sigma) \leftarrow \mathcal{A}_1^{\mathcal{O}_1(\cdot)}(K_p)$ ▷ σ is a "state" for $\mathcal{A}_1$ to transmit data to $\mathcal{A}_2$
$Y \leftarrow \mathsf{Enc}_{K_p}(X_b)$
$b' \leftarrow \mathcal{A}_2^{\mathcal{O}_2(\cdot)}(\sigma, Y)$
return b'

where the oracles $\mathcal{O}_1$ and $\mathcal{O}_2$ are defined as follows:

Oracle $\mathcal{O}_1(y)$:
return $\mathsf{Dec}_{K_s}(y)$
Oracle $\mathcal{O}_2(y)$:
if $y = Y$ **then**
    abort the game
end if
return $\mathsf{Dec}_{K_s}(y)$

We define the advantage

$$\mathsf{Adv}^{\mathsf{IND\text{-}CCA}}_{\mathcal{A}}(s) = \Pr[\mathsf{IND\text{-}CCA}^1_{\mathcal{A}}(s) \rightarrow 1] - \Pr[\mathsf{IND\text{-}CCA}^0_{\mathcal{A}}(s) \rightarrow 1]$$

We say that the cryptosystem is IND-CCA secure if for all probabilistic polynomial time (PPT) adversary $\mathcal{A}$, $\mathsf{Adv}^{\mathsf{IND\text{-}CCA}}_{\mathcal{A}}(s)$ is negligible.

Q.1 The definition of IND-CCA security which was given in the course (Def.5.5 on p.55–56 in the lecture notes, or slide p.404) was based on an interactive game between an adversary and a challenger. Prove that the two styles of definition for IND-CCA security are equivalent. (Carefully construct $(\mathcal{A}_1, \mathcal{A}_2)$ from an interactive adversary and an interactive adversary from $(\mathcal{A}_1, \mathcal{A}_2)$.)

Q.2 Let $\mathcal{A} = (\mathcal{A}_1, \mathcal{A}_2)$ be an IND-CCA adversary. We define another IND-CCA adversary as follows:

Algorithm $\mathcal{B}_1^{\mathcal{O}_1(\cdot)}(K_p)$
simulate $\mathcal{A}_1^{\mathcal{O}_1(\cdot)}(K_p) \rightarrow (X_0, X_1, \sigma)$
if $X_0 = X_1$ **then**
    set $\sigma' \leftarrow (\sigma, 1)$
    pick an arbitrary X such that $X \neq X_1$

return (X, X_1, σ')
else
set $\sigma' \leftarrow (\sigma, 0)$
return (X_0, X_1, σ')
end if

Algorithm $\mathcal{B}_2^{\mathcal{O}_2(\cdot)}(\sigma', Y)$
parse $\sigma' = (\sigma, c)$
if $c = 1$ **then**
return 0
else
simulate $\mathcal{A}_2^{\mathcal{O}_2(\cdot)}(\sigma, Y) \to b'$
return b'
end if

Prove that

$$\mathsf{Adv}_{\mathcal{A}}^{\mathsf{IND\text{-}CCA}}(s) = \mathsf{Adv}_{\mathcal{B}}^{\mathsf{IND\text{-}CCA}}(s)$$

Deduce that we can always assume $X_0 \neq X_1$ in an IND-CCA adversary.

We now define the NM-CCA game (for non-malleability) as follows:

Game $\mathsf{NM\text{-}CCA}_{\mathcal{A}}^{b}(s)$
$(K_p, K_s) \leftarrow \mathsf{Gen}(1^s)$
$(M, \sigma) \leftarrow \mathcal{A}_1^{\mathcal{O}_1(\cdot)}(K_p)$ ▷ σ is a "state" which allows $\mathcal{A}_1$ to transmit data to $\mathcal{A}_2$
$X_0 \leftarrow M$ ▷ M is a sampling algorithm defined by $\mathcal{A}_1$
$X_1 \leftarrow M$ ▷ we sample two independent plaintexts using M
$Y \leftarrow \mathsf{Enc}_{K_p}(X_1)$
$(R, Y'_1, \ldots, Y'_n) \leftarrow \mathcal{A}_2^{\mathcal{O}_2(\cdot)}(\sigma, Y)$ ▷ R is a poly. algo. returning a boolean
$X'_i \leftarrow \mathsf{Dec}_{K_s}(Y'_i)$, $i = 1, \ldots, n$
if $Y \notin \{Y'_1, \ldots, Y'_n\}$ and $\perp \notin \{X'_1, \ldots, X'_n\}$ and $R(X_b, X'_1, \ldots, X'_n)$ **then**
return 1
else
return 0
end if

We use the same oracles $\mathcal{O}_1$ and $\mathcal{O}_2$ as for IND-CCA. We define

$$\mathsf{Adv}_{\mathcal{A}}^{\mathsf{NM\text{-}CCA}}(s) = \Pr[\mathsf{NM\text{-}CCA}_{\mathcal{A}}^{1}(s) \to 1] - \Pr[\mathsf{NM\text{-}CCA}_{\mathcal{A}}^{0}(s) \to 1]$$

We say that the cryptosystem is NM-CCA secure if for all probabilistic polynomial time (PPT) adversary $\mathcal{A}$, $\mathsf{Adv}_{\mathcal{A}}^{\mathsf{NM\text{-}CCA}}(s)$ is negligible.

The goal of this exercise is to show the equivalence between NM-CCA security and IND-CCA security.

Q.3 We assume that $\mathcal{M}$ has a group structure (additively denoted), with at least two different elements 0 and 1, 0 being neutral. Assume that there is a polynomial algorithm Inc such that for all s,

$$\Pr\left[\mathsf{Dec}_{K_s}(\mathsf{Inc}_{K_p}(\mathsf{Enc}_{K_p}(X))) = X + 1\right] = 1$$

for $(K_p, K_s) \leftarrow \mathsf{Gen}(1^s)$. By constructing an adversary $\mathcal{A} = (\mathcal{A}_1, \mathcal{A}_2)$, prove that the cryptosystem is not $\mathsf{NM\text{-}CCA}$ secure.
(The precision of the proof is important.)
HINT: use M sampling in a set of two different plaintexts and R defined by $R(X, X') = 1_{X'=X+1}$.

Q.4 Given an $\mathsf{NM\text{-}CCA}$ adversary $\mathcal{A} = (\mathcal{A}_1, \mathcal{A}_2)$, we construct an $\mathsf{IND\text{-}CCA}$ adversary $\mathcal{B} = (\mathcal{B}_1, \mathcal{B}_2)$ as follows:

Algorithm $\mathcal{B}_1^{\mathcal{O}_1(\cdot)}(K_p)$
simulate $\mathcal{A}_1^{\mathcal{O}_1(\cdot)}(K_p) \to (M, \sigma)$
sample $z_0 \leftarrow M$
sample $z_1 \leftarrow M$
set $\sigma' \leftarrow (z_0, z_1, \sigma)$
return (z_0, z_1, σ')

Algorithm $\mathcal{B}_2^{\mathcal{O}_2(\cdot)}(\sigma', Y)$
parse $\sigma' = (z_0, z_1, \sigma)$
simulate $\mathcal{A}_2^{\mathcal{O}_2(\cdot)}(\sigma, Y) \to (R, Y'_1, \ldots, Y'_n)$
for $i = 1, \ldots, n$ **do**
  if $Y = Y'_i$ **then return** 0
  $X'_i \leftarrow \mathcal{O}_2(Y'_i)$
  if $X'_i = \perp$ **then return** 0
end for
compute $b' \leftarrow R(z_1, X'_1, \ldots, X'_n)$
return b'

Prove that

$$\mathsf{Adv}_{\mathcal{B}}^{\mathsf{IND\text{-}CCA}}(s) = \mathsf{Adv}_{\mathcal{A}}^{\mathsf{NM\text{-}CCA}}(s)$$

Deduce that $\mathsf{IND\text{-}CCA}$ security implies $\mathsf{NM\text{-}CCA}$ security.

Q.5 We assume that $\mathcal{M}$ has at least four elements.
Given an $\mathsf{IND\text{-}CCA}$ adversary $\mathcal{A} = (\mathcal{A}_1, \mathcal{A}_2)$, we construct an $\mathsf{NM\text{-}CCA}$ adversary $\mathcal{B} = (\mathcal{B}_1, \mathcal{B}_2)$ as follows:

Algorithm $\mathcal{B}_1^{\mathcal{O}_1(\cdot)}(K_p)$
simulate $\mathcal{A}_1^{\mathcal{O}_1(\cdot)}(K_p) \to (z_0, z_1, \sigma)$
define M sampling in $\{z_0, z_1\}$ with uniform distribution
set $\sigma' \leftarrow (\sigma, K_p, z_0, z_1)$
return (M, σ')

Algorithm $\mathcal{B}_2^{\mathcal{O}_2(\cdot)}(\sigma', Y)$
parse $\sigma' = (\sigma, K_p, z_0, z_1)$

take an injective function T on $\mathcal{M}$ such that $T(z_0) \notin \{z_0, z_1\}$ and $T(z_1) \notin \{z_0, z_1\}$

simulate $\mathcal{A}_2^{\mathcal{O}_2(\cdot)}(\sigma, Y) \to b'$

$Y' \leftarrow \mathsf{Enc}_{K_p}(T(z_{b'}))$

define $R(X, X') = 1_{T(X)=X'}$

return (R, Y')

Prove that

$$\mathsf{Adv}_{\mathcal{B}}^{\mathsf{NM\text{-}CCA}}(s) = \frac{1}{2}\mathsf{Adv}_{\mathcal{A}}^{\mathsf{IND\text{-}CCA}}(s)$$

Deduce that NM-CCA security implies IND-CCA security.
HINT$_1$: assume without loss of generality that $z_0 \neq z_1$
HINT$_2$: compute $\Pr[X_0 = z_{b'}]$, $\Pr[X_1 = z_{b'} | X_1 = z_1]$, and $\Pr[X_1 = z_{b'} | X_1 = z_0]$.

3 Unruh Transform from Σ to NIZK

We consider a Σ protocol (P, V) for a relation R. We let E be the set of challenges. Given some parameters t and $m \geq 2$, we define the following non-interactive zero-knowledge proof (NIZK), with input (x, w) such that $R(x, w)$ holds:

Algorithm $\mathsf{Proof}(x, w)$:

for $i = 1$ to t **do**
    pick a sequence of fresh coins ρ_i
    set $a_i \leftarrow P(x, w; \rho_i)$
    for $j = 1$ to m **do**
        pick $e_{i,j} \in E - \{e_{i,1}, \ldots, e_{i,j-1}\}$ at random
        set $z_{i,j} \leftarrow P(x, w, e_{i,j}; \rho_i)$
        set $h_{i,j} \leftarrow G(z_{i,j})$
    end for
end for
set $h \leftarrow H(x, (a_i, (e_{i,j}, h_{i,j})_{j=1,\ldots,m})_{i=1,\ldots,t})$
set $(J_1, \ldots, J_t) \leftarrow h$
set $z_i = z_{i,J_i}$ for $i = 1, \ldots, t$
set $\pi = (a_i, (e_{i,j}, h_{i,j})_{j=1,\ldots,m}, z_i)_{i=1,\ldots,t}$
return π

This algorithm uses two random oracles G and H. Oracle H is assumed to return a t-tuple of integers between 1 and m. We use the following verification algorithm (with some missing step):

Algorithm $\mathsf{Verify}(x, \pi)$:

parse $\pi = (a_i, (e_{i,j}, h_{i,j})_{j=1,\ldots,m}, z_i)_{i=1,\ldots,t}$
set $h \leftarrow H(x, (a_i, (e_{i,j}, h_{i,j})_{j=1,\ldots,m})_{i=1,\ldots,t})$
set $(J_1, \ldots, J_t) \leftarrow h$
verify $\cdots$
verify $V(x, a_i, e_{i,J_i}, z_i)$ for $i = 1, \ldots, t$
verify $h_{i,J_i} = G(z_i)$ for $i = 1, \ldots, t$
return 1 if all verifications passed

Q.1 By taking the verification with the missing step, give an algorithm to forge a proof given x but without the knowledge of w.
Which step should be added to have a sound proof?

Q.2 With the new verification step from the last question, given an algorithm with complexity $\mathcal{O}(m^t)$ to forge a valid π from x but without w.

Q.3 Construct a simulator in the random oracle model to show that the protocol is non-interactive zero-knowledge.

Q.4 Let $P^*(x)$ be an algorithm taking x as input, interacting with G and H, and forging a valid π with probability p. Use the next questions to prove that there is an extractor who can run P^* once to extract a witness w for x with probability at least $p - \mathsf{negl}$.

Q.4a Transform P^* into an algorithm P' who either aborts or makes a valid π. It returns π with probability p, and a complexity similar to P^*.

Q.4b Construct an extractor E on the previous P' such that by observing only one execution of P' with all queries to G and H, either P' aborts, or E finds a witness for x, or E aborts. But the probability that E aborts is bounded by $n_G n_H m t N^{-1} + n_H m^{-t}$, where n_G is the number of queries to G, n_H is the number of queries to H, and N is the size of the range of G.

Hint: say that a query q to H is good if it can be parsed in the form

$$q = x, (a_i, (e_{i,j}, h_{i,j})_{j=1,\ldots,m})_{i=1,\ldots,t}$$

Consider an extractor which aborts if any fresh query to G returns a value $h_{i,j}$ which is included in a previous good query q to H. Define another abort condition and extract a witness in remaining cases.