Hybrid Proofs in Cryptography

2019-2020

Contents

1 Proof of El-Gamal

1.1 El-Gamal scheme

- $\mathrm{Gen}(1^\lambda) : s \overset{\$}{\leftarrow} \mathbb{Z}_p$, $pk = g^s$, $sk = s$
- $\mathrm{Enc}(pk, m) : r \overset{\$}{\leftarrow} \mathbb{Z}_p$, $ct = (ct_1, ct_2) = (g^r, g^{sr}.m)$
- $\mathrm{Dec}(sk, (ct_1, ct_2)) = \frac{ct_2}{ct_1^{sk}}$

We want to quantify $\mathscr{A}^{\texttt{IND-CPA}}(\mathscr{A}) = |2\mathbb{P}(\mathscr{A}(pk, \mathrm{Enc}(pk, m_b)) = b) - 1|$

Lemma 1. $|2\mathbb{P}(\mathscr{A}(pk, m_b) = b) - 1| = |\mathbb{P}(\mathscr{A}(pk, ct_0)) = 0) - |\mathbb{P}(\mathscr{A}(pk, ct_1) = 0)| = |\mathbb{P}(\mathscr{A}(pk, ct_0)) = 1) - |\mathbb{P}(\mathscr{A}(pk, ct_1) = 1)|$

Proof.

$$|2\mathbb{P}(\mathscr{A}(pk, ct_b) = b) - 1|$$
$$= |2(\mathbb{P}(\mathscr{A}(pk, ct_0)) = 0)\mathbb{P}(b = 0) + \mathbb{P}(\mathscr{A}(pk, ct_1)) = 1)\mathbb{P}(b = 1)) - 1|$$
$$= |\mathbb{P}(\mathscr{A}(pk, ct_0) = 0) + \mathbb{P}(\mathscr{A}(pk, ct_1) = 1) - 1|$$

The claim now follows from the fact that

$$1 = \mathbb{P}(\mathscr{A}(pk, ct_1) = 1) + \mathbb{P}(\mathscr{A}(pk, ct_1) = 0)$$

. □

1.2 2-IND-CPA Games

The game b is defined by :

- $Init()$
  $(pk, sk) \xleftarrow{\$} \text{Gen}(1^\lambda)$
  Return pk
- $Challenge(m_0, m_1)$
  Return $\text{Enc}(pk, m_b)$
- $Finalize(b')$
  Return b'

1.3 Proof of El-Gamal

For all games :
$Init()$
$(pk, sk) \xleftarrow{\$} \text{Gen}(1^\lambda)$
Return $pk = g^s$
$Finalize(b')$
Return b'

Game 0
Challenge(m_0, m_1) :
$r \xleftarrow{\$} \mathbb{Z}_p$
Return $ct = (g^r, g^{sr}.m_0)$

Hyb 1
Challenge(m_0, m_1) :
$r \xleftarrow{\$} \mathbb{Z}_p, c \xleftarrow{\$} \mathbb{Z}_p$
Return $ct = (g^r, g^c.m_0)$

Hyb 2
Challenge(m_0, m_1) :
$r \xleftarrow{\$} \mathbb{Z}_p, c \xleftarrow{\$} \mathbb{Z}_p$
Return $ct = (g^r, g^c)$

Hyb 3
Challenge(m_0, m_1) :
$r \xleftarrow{\$} \mathbb{Z}_p, c \xleftarrow{\$} \mathbb{Z}_p$
Return $ct = (g^r, g^c.m_1)$

Game 1
Challenge(m_0, m_1) :
$r \xleftarrow{\$} \mathbb{Z}_p$
Return $ct = (g^r, g^{rs}.m_1)$

Lemma 2. *Assuming DDH Game 0 and Hyb 1 are computationnaly indistinguishable.*

Proof. Let A be a distinguisher between Game 0 and Hyb 1, we built a distinguisher B for DDH. B gets a DDH tuple : (g, g^a, g^b, z) where $z = g^{ab}$ or $z = c, c \overset{\$}{\leftarrow} \mathbb{Z}_p$. B sets $pk = g^a$ and sends pk to A as Init. On query (m_0, m_1) for Challenge, B sets implicitly r as b and outputs $ct = (g^b, z.m_0)$. If $z = g^{ab}$ we have $ct = (g^b, g^{ab}.m_0)$ which correspond to challenge in Game 0, otherwise $ct = (g^b, g^c.m_0)$ which correspond to challenge in Hyb 1. When A halts with output b', so does B.
$\mathbb{P}[B(g, g^a, g^b, g^{ab}) = 1] = \mathbb{P}[A(pk, ct, g^r, g^{st}.m_0) = 1]$ and
$\mathbb{P}[B(g, g^a, g^b, g^c) = 1] = \mathbb{P}[A(pk, ct, g^r, g^c.m_0) = 1]$.
So $\mathrm{Adv}^{\texttt{DDH}}(B) \geq \mathrm{Adv}^{\text{Game 0-Hyb 1}}(A)$.

□

$\Rightarrow |\mathbb{P}[A^{\text{Game 0}} = 1] - \mathbb{P}[A^{\text{Hyb 1}} = 1]| \leq \mathrm{Adv}^{\texttt{DDH}}(B)$ with B turns in same time as A.

Lemma 3. *Hyb 1 ≡ Hyb 2.*

Proof. The two distributons are the same. □

$\Rightarrow |\mathbb{P}[A^{\text{Hyb 1}} = 1] - \mathbb{P}[A^{\text{Hyb 2}} = 1]| = 0$

Lemma 4. *Hyb 2 ≡ Hyb 3.*

Proof. Same reason. □

$\Rightarrow |\mathbb{P}[A^{\text{Hyb 2}} = 1] - \mathbb{P}[A^{\text{Hyb 3}} = 1]| = 0$

Lemma 5. *Assuming DDH Game 1 and Hyb 3 are computationnaly indistinguishable.*

Proof. Same as the proof for Game 0 and Hyb 1. □

$\Rightarrow |\mathbb{P}[A^{\text{Hyb 3}} = 1] - \mathbb{P}[A^{\text{Game 1}} = 1]| \leq \mathrm{Adv}^{\texttt{DDH}}(B)$
Let A be an adversary against `IND-CPA` security of El-Gamal scheme :

$$\mathrm{Adv}^{\texttt{IND-CPA}}(A) = |\mathbb{P}[A^{Game0} = 1] - \mathbb{P}[A^{Game1} = 1]|$$
$$= \mathbb{P}[A^{Game0} = 1] - \mathbb{P}[A^{Hyb1} = 1] + \mathbb{P}[A^{Hyb1} = 1] - \mathbb{P}[A^{Hyb2} = 1]$$
$$+\mathbb{P}[A^{Hyb2} = 1] - \mathbb{P}[A^{Hyb3} = 1] + \mathbb{P}[A^{Hyb3} = 1] - \mathbb{P}[A^{Game1} = 1]|$$
$$\leq 2\,\mathrm{Adv}^{\texttt{DDH}}(B)$$

where B's running time is the same as A's.

2 Proof of Boneh-Franklin IBE

2.1 Scheme

- Setup(1^λ) : $s \overset{\$}{\leftarrow} \mathbb{Z}_p$, $mpk = g^s$, $msk = s$
  Return (msk, mpk)

- KeyGen(msk, ID) : $h = H(ID)$ with $H : \{0,1\}^* \to G$ a random oracle.
  Return $(s_{ID} = h^s)$

- Enc(mpk, ID, m) : $h = H(ID)$, $r \overset{\$}{\leftarrow} \mathbb{Z}_p$
  Return $(ct = (f^r, e(h, mpk)^r.m)$

- Dec$(s_{ID}, (ct_1, ct_2))$: Return $\frac{ct_2}{e(s_{ID}, ct_1)}$

2.2 Construction of the proof

The two following functions are accessible in all games for adversary :

- KeyGenOracle(ID) :
  $h = H(ID)$
  Return $s_{ID} = h^s$
- RandomOracle($x \in \{0,1\}^*$)
  If $x \in T$: Return $T[x]$
  Else : $h \xleftarrow{\$} G$, $T[x] \leftarrow h$
  Return $T[x]$

Definition 1. *The decisionnal bilinear Diffie-Helman assumption (DBDH) is for* $d \xleftarrow{\$} \mathbb{Z}_p$

$$(e, g, g^a, g^b, g^c, e(g,g)^{abc}) \simeq_c (e, g, g^a, g^b, g^c, e(g,g)^d)$$

For all games : Init(mpk, msk) $\leftarrow$ Setup(1^λ) ,Return $mpk = g^s$

Game 0

Challenge(ID, m_0, m_1) :
Return $ct = (g^r, e(H(ID), g^s)^r.m_0)$

Hyb 1

Challenge(ID, m_0, m_1) :
$d \xleftarrow{\$} \mathbb{Z}_p$
Return $ct = (g^r, e(g,g)^d.m_0)$

Hyb 2

Challenge(ID, m_0, m_1) :
$d \xleftarrow{\$} \mathbb{Z}_p$
Return $ct = (g^r, e(g,g)^d)$

Hyb 3

Challenge(ID, m_0, m_1) :
$d \xleftarrow{\$} \mathbb{Z}_p$
Return $ct = (g^r, e(g,g)^d.m_1)$

Game 1

Challenge(ID, m_0, m_1) :
Return $ct = (g^r, e(H(ID), g^s)^r.m_1)$

2.3 Proof of security

Theorem 1. *Assuming DBDH, Game 0* $\simeq_c$ *Hyb 1.*

Proof. Let A denote a distinguisher for Game 0 - Hyb 1 that makes q random oracle queries. We build B a distinguisher for DBDH as follows :

The key idea here is that we want the challenge ciphertext to depend on a, b, c, so we use a as the secret key by using g^a as mpk, we keep b as randomness for the challenge ciphertext, and then we want to use g^c as the hash value for the challenge idendity. Since we do not know in advance which idendity is going to be the challenge identity, we guess it. Another idea to avoid guessing the challenge idendity would be to use $(g^c)^t$ to define the outputs of H so that we are certain that the hash value of the target identity depends on c. The issue doing so is that one cannot generate secret keys if $H(ID) = (g^c)^t$ since we would need to compute g^{cta} and we only know g^a, g^c and t but not a nor c. This is not possible unless CDH is easy (in which case so is DBDH).

B gets a DBDH tuple (g^a, g^b, g^c, z) where $z = e(g, g)^{abc}$ or $z \overset{\$}{\leftarrow} G$.
For Init, B implicitly sets $s = a$ by letting $mpk = g^a$ and sends it to A. In addition B picks $j \overset{\$}{\leftarrow} [1, q]$ as a guess of ID^* being the j-th query to the random oracle H.
Initiate T of hash values and $ctr = 0$. When A asks for a hash query ID, B does the following :

Algorithm 1 Hash query

```
if $T[ID] \neq \perp$ then
  return $T[ID]$
else
  $ctr \leftarrow ctr + 1$
  if $ctr = j$ then
    $T[ID] = (\perp, g^c)$
  else
    $t \overset{\$}{\leftarrow} \mathbb{Z}_p$
    $T[ID] = (t, g^t)$
  end if
  return $T[ID]$
end if
```

When A makes a KeyGen query ID, B does the following:

Algorithm 2 KeyGen query

```
if $T[ID] = \perp$ then
  define it (add it as a hash query)
end if
if $T[ID] = (\perp, g^c)$ then
  Guess was incorrect :$B$ halts and outputs $b' \overset{\$}{\leftarrow} \{0, 1\}$
else
  $T[ID] = (t, g^t)$
  return $s_{ID} = mpk^t = (g^s)^t = H(ID)^s$
end if
```

When A challenges (ID, m_0, m_1) :
If $T[ID] \neq (\perp, g^c)$: B halts and outputs $b' \overset{\$}{\leftarrow} \{0, 1\}$
Else : return $ct = (g^b, z.m_0)$.
If $z = e(g, g)^{abc}$, B simulate Game 0, otherwise Hyb 1. When A halts with output b so does B.

Conclusion :

$$|\mathbb{P}[B^A(g^a, g^b, g^c, e(g, g)^{abc} = 1] - \mathbb{P}[B^A(g^a, g^b, g^c, e(g, g)^d = 1]|$$

$$
\begin{aligned}
&= |\mathbb{P}[B^A(g^a, g^b, g^c, e(g,g)^{abc} = 1|\text{guess correct}]\mathbb{P}[\text{guess correct}] \\
&+\mathbb{P}[B^A(g^a, g^b, g^c, e(g,g)^{abc} = 1|\text{guess wrong}]\mathbb{P}[\text{guess wrong}] \\
&-\mathbb{P}[B^A(g^a, g^b, g^c, e(g,g)^{d} = 1|\text{guess correct}]\mathbb{P}[\text{guess correct}] \\
&-\mathbb{P}[B^A(g^a, g^b, g^c, e(g,g)^{d} = 1|\text{guess wrong}]\mathbb{P}[\text{guess wrong}] \\
&= |\mathbb{P}[B^A(g^a, g^b, g^c, e(g,g)^{abc} = 1|\text{guess correct}] - \mathbb{P}[B^A(g^a, g^b, g^c, e(g,g)^{d} = 1|\text{guess correct}] \\
&= \frac{1}{q}(\mathbb{P}[A^{\text{Game 0}} = 1] - \mathbb{P}[A^{\text{Hyb 1}} = 1]
\end{aligned}
$$

Fianlly

$$\frac{1}{q}\,\text{Adv}^{\text{Game 0 - Hyb 1}}(A) \leq \text{Adv}^{\text{DBDH}}(B)$$

□

Lemma 6. *Hyb 1 $\equiv$ Hyb 2 $\equiv$ Hyb 3.*

Proof. It's the same distribution □

Lemma 7. *Assuming DBDH, Hyb 3 $\simeq_c$ Game 1.*

Proof. Same as the proof for Game 0 and Hyb 1. □

Putting everything together give $\text{Adv}^{\texttt{IND-ID-CPA}}(A) \leq 2q\,\text{Adv}^{\text{DBDH}}(B)$ where B's running time is roughly the same as A and q is the number of random oracle query. We would prefer on dependance of q_{KeyGen}, the number of corruped identities.

2.4 Exercise

Inprove the bound with $\text{Adv}^{\texttt{IND-ID-CPA}}(A) \leq 2eq_{\text{KeyGen}}\,\text{Adv}^{\text{DBDH}}(B)$.
Hint : Generate $H(ID)$ as a random group element obtained from g^c, i.e. $g^{c\alpha}$ with known α with proba $\frac{1}{q_{\text{KeyGen}}}$. With a certain probability (to compute), c is embedded in the challenge but is not in every KeyGen queries.