Eidgenössische
Technische Hochschule
Zürich

Ecole polytechnique fédérale de Zurich
Politecnico federale di Zurigo
Federal Institute of Technology at Zurich

Departement of Computer Science
Prof. Dennis Hofheinz
Julia Kastner

Information Security Lab 2022
Module 02 - Reductions - Week 2
October 17 - 21

Solutions

Exercise 1 (Ex. 11.6 in Katz and Lindell [KL21])**.** Consider a cyclic group $\mathbb{G}$ of prime order p and let $\mathbf{g}$ be a generator of $\mathbb{G}$. We look at the following PKE scheme for bits. The public key is $\mathsf{pk} = (\mathbf{g}, \mathbf{h} = \mathbf{g}^x)$, the secret key is $\mathsf{sk} = x$. In order to encypt a bit b the sender does the following:

case $b = 0$**:** Choose a uniform $y \in \mathbb{Z}_p$ and compute $c_1 := \mathbf{g}^y$ and $c_2 := \mathbf{h}^y$. Output the ciphertext (c_1, c_2).

case $b = 1$**:** Choose uniform $y, z \in \mathbb{Z}_p$ and compute $c_1 := \mathbf{g}^y$ and $c_2 := \mathbf{g}^z$. Output the ciphertext (c_1, c_2).

Prove that this encryption scheme is IND-CPA secure if **DDH** is hard in $\mathbb{G}$. Name two disadvantages of this scheme, compared to the standard ElGamal PKE.

Solution. On input of a **DDH** challenge tuple $\mathbf{g}, \mathbf{g}^x, \mathbf{g}^y, \mathbf{Z}$ the reduction sets $\mathbf{h} = \mathbf{g}^x$ and sends the public key $\mathsf{pk} = (\mathbf{g}, \mathbf{h})$ to the adversary. When the adversary submits challenge messages $0, 1$ the reduction returns $\mathbf{g}^y, \mathbf{Z}$ as the ciphertext. If the adversary outputs 0, $\mathbf{Z} = \mathbf{g}^{xy}$, otherweise $\mathbf{Z} = \mathbf{g}^z$ for uniformly random $z \in \mathbb{Z}_p$.

Disadvantages:

- not efficient (if one wants to encrypt longer ciphertexts a lot of group elements are needed)
- does not have perfect correctness (sender may choose $z = xy$ by accident with probability $\frac{1}{p}$)

Exercise 2. Consider the following signature schemes:

$\Sigma_1 = (\mathsf{KeyGen}_1, \mathsf{Sign}_1, \mathsf{Verify}_1)$ over a group $\mathbb{G}$ of prime order p with generator $\mathbf{g}$ works as follows:

KeyGen_1 sample $x \xleftarrow{\$} \mathbb{Z}_p$. Set $\mathsf{sk} = x$ and $\mathsf{pk} = \mathbf{g}^x$.

$\mathsf{Sign}_1(\mathsf{sk} = x, m)$ for a message $m \in \mathbb{Z}_p$ it computes $\sigma = x + m$ and outputs σ.

$\mathsf{Verify}_1(\mathsf{pk}, m, \sigma)$ outputs 1 iff $\mathbf{g}^m \cdot \mathsf{pk} = \mathbf{g}^\sigma$.

$\Sigma_2 = (\mathsf{KeyGen}_2, \mathsf{Sign}_2, \mathsf{Verify}_2)$ over a group $\mathbb{G}$ of prime order p with generator $\mathbf{g}$ works as follows:

KeyGen_2 sample $x \xleftarrow{\$} \mathbb{Z}_p$. Set $\mathsf{sk} = x$ and $\mathsf{pk} = \mathbf{g}^x$.

$\mathsf{Sign}_2(\mathsf{sk} = x, m)$ For a message $m = (m_1, m_2) \in \mathbb{G} \times \mathbb{G}$ computes $\sigma = m_1^{-x} \cdot m_2$ and outputs σ. We note that this corresponds to the decryption algorithm of ElGamal.

$\mathsf{Verify}_2(\mathsf{pk}, m, \sigma)$ outputs 1 iff $\mathsf{ElGamal.Enc}(\mathsf{pk}, \sigma) = m$.

Answer the following questions about the schemes:

(a) Is the scheme correct?

(b) If the scheme is correct, assuming that the discrete logarithm problem is hard in $\mathbb{G}$, is the scheme EUF-NMA secure?

(c) If the scheme is correct, assuming that the discrete logarithm problem is hard in $\mathbb{G}$, is it EUF-CMA secure?

Solution. For scheme Σ_1 we answer the questions as follows:

(a) Yes, the scheme is correct. For any message $m \in \mathbb{Z}_p$, any $(\mathsf{sk}, \mathsf{pk}) \xleftarrow{\$} \mathsf{KeyGen}_1$s and $\sigma = \mathsf{Sign}_1(\mathsf{sk} = x, m)$ it holds that $\mathbf{g}^\sigma = \mathbf{g}^{m+x} = \mathbf{g}^m \cdot \mathsf{pk}$

(b) Yes, the scheme is EUF-NMA secure. Assume there exists an efficient adversary $\mathcal{A}$ that breaks EUF-NMA security. Consider the following reduction: The reduction receives a discrete logarithm challenge $\mathbf{h}$ from its challenger. It sets $\mathsf{pk} = \mathbf{h}$ and gives it to the adversary $\mathcal{A}$. If successful, the adversary provides the reduction with a message-signature pair $(m, \sigma) \in \mathbb{Z}_p \times \mathbb{Z}_p$. The reduction computes $\sigma - m$ and outputs it as its solution to the discrete logarithm problem.

(c) No, it is not EUF-CMA secure. An adversary can ask for a signature on a random message $m \xleftarrow{\$} \mathbb{Z}_p$ and then compute the secret key from the signature it received as the reduction did in the previous subtask. It can then use the secret key to sign arbitrary messages of its choice.

For scheme Σ_2 we answer as follows:

(a) No, the scheme is not correct. As the ElGamal encryption algorithm is randomized, the ciphertext generated during verification will not be equal to m with an overwhelming probability.

Exercise 3. Given an IND-CCA secure PKE scheme E_0 construct:

(a) a PKE scheme E_1 that is IND-CPA secure but not IND-CCA1 secure

(b) a PKE scheme E_2 that is IND-CCA1 secure but not IND-CCA secure

Solution.

(a) we define the algorithm $\mathsf{KeyGen}_1 = \mathsf{KeyGen}_0$, $\mathsf{Enc}_1(\mathsf{pk}, m) = \mathsf{Enc}_0(\mathsf{pk}, m\|1)$, $\mathsf{Dec}_1(\mathsf{sk}, c)$ executes $m\|b := \mathsf{Dec}_0(\mathsf{sk}, c)$ and if $b = 1$ outputs m, if $b = 0$ outputs sk. The IND-CCA1 attacker sends a ciphertext of $m\|0$ to the decryption oracle to obtain the secret key which then allows it to break the security of the scheme.

(b) we define $\mathsf{KeyGen}_2 = \mathsf{KeyGen}_0$, $\mathsf{Enc}_2(\mathsf{pk}, m) = \mathsf{Enc}_0(\mathsf{pk}, m)\|1$, $\mathsf{Dec}_2(\mathsf{sk}, c\|b) = \mathsf{Dec}_0(\mathsf{sk}, c)$, i.e. decryption ignores the last bit. The IND-CCA attacker replaces the last bit of its challenge ciphertext by 0 and sends it to the decryption oracle to obtain the message m_b.

For IND-CCA1 security we give a reduction under IND-CCA security of the original scheme. The reduction forwards the public key. When the adversary makes decryption queries it removes the last bit before forwarding them to its own decryption oracle. It submits the adversary's challenge messages to its own challenger and appends 1 to the challenge ciphertext before forwarding it to the adversary. The adversary is no longer allowed to make any decryption queries after seeing the challenge ciphertext. When the adversary outputs a bit, the reduction outputs the same bit to its challenger.

References

[KL21] Jonathan Katz and Yehuda Lindell. *Introduction to modern cryptography*. eng. Third edition. Chapman & Hall/CRC cryptography and network security. Boca Raton, Florida ; CRC Press, 2021. ISBN: 1-351-13303-9.