ETH Zürich, D-INFK
Spring 2019

Prof. Ueli Maurer
Fabio Banfi
Daniel Jost
Jiamin Zhu

# Cryptography Foundations
# Solution Exercise 4

### 4.1 The ElGamal Public-Key Cryptosystem

**a)** A ciphertext $(y_A, c) = (g^x, m \cdot g^{x \cdot x_B})$ is decrypted with the secret key $x_B$ as follows:

$$c \cdot ((y_A)^{x_B})^{-1} = (m \cdot g^{x \cdot x_B}) \cdot ((g^x)^{x_B})^{-1} = m.$$

**b)** From a distinguisher $D$ for the bit-guessing problem $[\![ S^{\textsf{pke-cpa}}; B ]\!]$, we construct a distinguisher $D'$ for the problem of distinguishing $\textsf{DDH}^0$ and $\textsf{DDH}^1$. Upon obtaining $(R, S, T) \in G^3$, $D'$ first sends $S$ to $D$ as the public key. When $D$ submits two challenge messages $m_0$ and $m_1$, $D'$ chooses a bit $B$ uniformly at random and sends $(R, m_B \cdot T)$ to $D$. When $D$ issues a bit $Z$, $D'$ outputs the bit $Z' \coloneqq Z \oplus B$.

Note that when $D'$ interacts with $\textsf{DDH}^0$, it perfectly emulates $[\![ S^{\textsf{pke-cpa}}; B ]\!]$ towards $D$. Assume $D'$ interacts with $\textsf{DDH}^1$; in this case, $(R, S, T)$ are uniform and independent, and therefore $m_B \cdot T$ is distributed uniformly by Exercise 1.2. Moreover, $R$, $S$, and $m_B \cdot T$ are independent and therefore the bit $Z$ output by $D$ in this case is statistically independent of $B$, and thus $Z$ equals $B$ with probability $\frac{1}{2}$. Finally, for the advantage of $D'$ we have:

$$\begin{aligned}
\Delta^{D'}(\textsf{DDH}^0, \textsf{DDH}^1) &= \Pr^{D'\textsf{DDH}^1}[Z' = 1] - \Pr^{D'\textsf{DDH}^0}[Z' = 1] \\
&= \Pr^{D'\textsf{DDH}^1}[\underbrace{Z \oplus B = 1}_{Z \neq B}] - \Pr^{D'\textsf{DDH}^0}[\underbrace{Z \oplus B = 1}_{Z \neq B}] \\
&= \frac{1}{2} - (1 - \Pr^{DS^{\textsf{pke-cpa}}}[Z = B]) \\
&= \Pr^{DS^{\textsf{ind}}}[Z = B] - \frac{1}{2} \\
&= \frac{1}{2} \cdot \Lambda^D\big([\![ S^{\textsf{pke-cpa}}; B ]\!]\big).
\end{aligned}$$

### 4.2 On the (In)security of RSA

**a)** Let $\mathcal{M}_0 \coloneqq \{m \in \mathbb{N} \mid m^3 < n\}$. For any message $m \in \mathcal{M}_0$, the corresponding ciphertext equals $c = m^e \bmod n = m^3$. Since $m^3$ is not reduced modulo $n$, one can efficiently compute $m = c^{1/3}$ over the integers using numerical methods and hence recover the message from the ciphertext.

**b)** If the $n_i$ are not pairwise coprime, one can find a nontrivial factor of one of the $n_i$ with the extended Euclidean algorithm and hence compute the corresponding secret key. With the secret key, the corresponding ciphertext $c_i$ can be decrypted to find the message.

Now assume the $n_i$ are pairwise coprime and the following ciphertexts are intercepted:

$$\begin{aligned}
c_1 &\equiv m^3 \pmod{n_1}, \\
c_2 &\equiv m^3 \pmod{n_2}, \\
c_3 &\equiv m^3 \pmod{n_3}.
\end{aligned}$$

Using the Chinese Remainder Theorem, one can efficiently compute

$$c \equiv m^3 \pmod{n_1 n_2 n_3}.$$

Since $m < n_i$ for $i = 1, 2, 3$, we have $m^3 < n_1 n_2 n_3$, so $c = m^3$ is not reduced. Therefore, one can again recover $m$ by calculating the cubic root of $c$ in the integers.

This attack is called *Håstad attack*. It analogously works for any $e > 3$ if the same message is encrypted with at least $e$ different public keys.

**c)** We have $n = pq$ and $\varphi(n) = (p-1)(q-1) = pq-p-q+1$. This implies $p = n-q+1-\varphi(n)$ and therefore $n = nq-q^2+q-\varphi(n)q$. Since $n$ and $\varphi(n)$ are known, this quadratic equation can be solved efficiently for $q$, which also yields $p$.

### 4.3 Homomorphic Public-Key Encryption

**a)** For the ElGamal cryptosystem we have that the message space is a finite abelian (and cyclic) group $\langle \mathbb{G}; \circ \rangle$ of order $q := |\mathbb{G}|$ with generator $g$. The ciphertext space is $\mathbb{G}^2$, on which we define $\otimes$ as the elementwise extension of $\circ$ (that is we consider the product group, where for $a, b \in \mathbb{G}^2$ with $a := (a_1, a_2)$ and $b := (b_1, b_2)$, we define $a \otimes b := (a_1 \circ b_1, a_2 \circ b_2)$).

For any fixed public key $y_B := g^{x_B} \in \mathbb{G}$ (where $x_B \in \mathbb{Z}_q$ is the private key) and arbitrary messages $m_1, m_2$, let the ciphertexts be $c_i := E(m_i, y_B) = (g^{x_i}, m_i \circ y_B^{x_i})$, where $i \in \{1, 2\}$ and $x_i \in \mathbb{Z}_q$ are uniformly distributed. For $c_1 \otimes c_2$ we obtain

$$\begin{aligned} c_1 \otimes c_2 &= (g^{x_1}, m_1 \circ y_B^{x_1}) \otimes (g^{x_2}, m_2 \circ y_B^{x_2}) \\ &= (g^{x_1} \circ g^{x_2}, m_1 \circ y_B^{x_1} \circ m_2 \circ y_B^{x_2}) \\ &= (g^{x_1+x_2}, (m_1 \circ m_2) \circ y_B^{x_1+x_2}) \end{aligned}$$

where the last step follows since we assumed the group to be abelian.

The last line corresponds to an encryption of $(m_1 \circ m_2)$ with the first element of the ciphertext having exponent $x := x_1 + x_2$. Therefore a valid encryption (with respect to the public key $y_B$) of $m_1 \circ m_2$ can be computed as $c_1 \otimes c_2$.

**b)** For the naïve RSA cryptosystem we have that both the message and ciphertext spaces are the finite multiplicative group $\mathbb{Z}_n^*$ of integers modulo $n := p \cdot q$ for $p$ and $q$ primes and with order $t := |\mathbb{Z}_n^*| = \varphi(n) = (p-1)(q-1)$.

For any fixed public key $e \in \mathbb{Z}_t^*$ (relative to some private key $d \in \mathbb{Z}_t^*$ such that $e \cdot d \equiv_t 1$) and arbitrary messages $m_1, m_2$, let the ciphertexts be $c_i := E(m_i, (n, e)) = [m_i^e]_n$ (for $i = 1, 2$ and where $[x]_n$ denotes the remainder of $x$ when divided by $n$). We have:

$$\begin{aligned} c_1 \cdot c_2 &\equiv_n E(m_1, (n, e)) \cdot E(m_2, (n, e)) \\ &\equiv_n m_1^e \cdot m_2^e \\ &\equiv_n (m_1 \cdot m_2)^e \\ &\equiv_n E(m_1 \cdot m_2, (n, e)). \end{aligned}$$

Therefore the encryption of $[m_1 \cdot m_2]_n$ is computed as $[c_1 \cdot c_2]_n$.

**c)** Let $\langle \mathbb{G}; \circ \rangle$ be the group associated with the message space of $(E, d)$. Due to the assumption of the exercise, we can assume the existence of an (efficiently computable) function $\gamma$ that, given two encryptions $c, c'$ of messages $m, m'$ (with respect to $pk$), computes a valid encryption of $m \circ m'$ (with respect to $pk$).

Then the following distinguisher wins the CCA bit-guessing problem with probability 1: first, it chooses three messages $m_0, m_1, m_2$ that are (pairwise) different and $m_2$ not equal the neutral element. It first asks for the encryption of $m_2$, thus obtaining $c_2 := E(m_2, pk)$.

Then it presents the challenge $(m_0, m_1)$ to the challenger, thus obtaining $c := E(m_b, pk)$ for $b \in \{0, 1\}$ uniformly distributed. At this point the adversary can exploit the fact that $(E, d)$ is homomorphic by computing $\widetilde{c} := \gamma(c, c_2)$, which is a valid encryption of $m_b \circ m_2$ relative to the public key $pk$.

Due to correctness of the encryption scheme, $\widetilde{c} \neq c$ and hence asking for the decryption of $\widetilde{c}$ is an allowed query to the decryption oracle. The decryption of $\widetilde{c}$ yields $m' := m_b \circ m_2$ and computing $m' \circ (m_2)^{-1} = m_b$ completely recovers $m_b$. This is sufficient to win the CCA game with probability 1.

**d)** There are two important use-cases for homomorphic encryption schemes.

- In electronic voting schemes, the homomorphic property can be used to count the votes. In a 0/1-vote (i.e., yes/no-votes without abstentions) for example, one could compute the sum of all encrypted votes without the need to decrypt each vote. This protects *voter privacy*. Only the final result will be decrypted in the end. Note that to make such a procedure sound, several other techniques are needed in combination, as treated for example in the lecture on *Cryptographic Protocols*. Homomorphic encryption is just one helpful tool to achieve the goal of a secure voting protocols.
- Assume a client wants to outsource a specific computation to a server. The server is trusted not to tamper with the data, but the client might be afraid that the data he sends to the server could leak to an intruder. So, the client sends his encrypted inputs for the computation to the server and the server then performs all operations on the respective ciphertexts. Thanks to the homomorphic property, this will translate to operations performed on plaintexts. The server returns the encrypted result and the client simply needs to decrypt the returned value to obtain the result of the computation. Note that for this to be even more powerful, one might need an encryption scheme that is homomorphic with respect to ring operations, such as addition and multiplication simultaneously. This topic however, which is typically referred to as *fully homomorphic encryption*, will not be of primary interest in this lecture.

### 4.4 The Rabin Trapdoor One-Way Permutation

**a)** Consider the function

$$x \mapsto x^2 \colon \mathbb{Z}_p^* \to \mathcal{QR}_p. \tag{1}$$

It is clearly surjective. Moreover for every $x$, $-x$ is mapped to the same quadratic residue $x^2 = (-x)^2$ and $-x \neq x$ because otherwise $2x \equiv 0 \pmod p$, which would imply that 2 divides $p$. On the other hand, from $x^2 = (x')^2$ it follows that $(x'x^{-1})^2 = 1$. The latter implies that $x'x^{-1} = \pm 1 \Leftrightarrow x' = \pm x$ as the polynomial $X^2 - 1 = (X + 1)(X - 1)$ only has the two roots $\pm 1$. We thus have established that the function (1) is 2-to-1 and that the number of quadratic residues in $\mathbb{Z}_p^*$ is $\frac{1}{2}$ of the number of elements in $\mathbb{Z}_p^*$, i.e. $|\mathcal{QR}_p| = \frac{p-1}{2}$.

**b)** Observe that since $p \equiv 3 \pmod 4$, $p + 1$ is divisible by 4. Now set $y := x^{\frac{p+1}{4}}$. As $x$ is a quadratic residue modulo $p$, there exists an integer $z$ such that $z^2 \equiv x \pmod p$. We find

$$y^2 \equiv x^{\frac{p+1}{2}} \equiv x^{\frac{p-1}{2}+1} \equiv x^{\frac{p-1}{2}} x \equiv (z^2)^{\frac{p-1}{2}} x \equiv z^{p-1} x \equiv x \pmod p,$$

where in the last step we used that by Lagrange, $z^{|\mathbb{Z}_p^*|} = z^{p-1} = 1$.

It remains to show that $y \in \mathcal{QR}_p$. To this end, observe that $\mathcal{QR}_p$ is a subgroup of $\mathbb{Z}_p^*$. Especially for $x, x' \in \mathcal{QR}_p$, we have $xx' = y^2(y')^2 = (yy')^2$ and hence $\mathcal{QR}_p$ is closed under multiplication. Since $x \in \mathcal{QR}_p$, therefore $x^{\frac{p+1}{4}} \in \mathcal{QR}_p$ as well.

**c)** By the Chinese remainder theorem, the function $\phi\colon \mathbb{Z}_n \to \mathbb{Z}_p \times \mathbb{Z}_q$ given by

$$\phi(x) = (x \bmod p, x \bmod q)$$

is an isomorphism of rings[1]. We say that $\mathbb{Z}_n$ is (canonically) isomorphic to $\mathbb{Z}_p \times \mathbb{Z}_q$ and write $\mathbb{Z}_n \cong \mathbb{Z}_p \times \mathbb{Z}_q$. It follows that an element $x \in \mathbb{Z}_n$ is a unit, which means that $x$ has a multiplicative inverse, $x \in \mathbb{Z}_n^*$, if and only if the corresponding elements $x \bmod p$ and $x \bmod q$ are units in $\mathbb{Z}_p$ and $\mathbb{Z}_q$, respectively. Therefore, the restriction of $\phi$ to $\mathbb{Z}_n^*$ gives an isomorphism $\mathbb{Z}_n^* \cong \mathbb{Z}_p^* \times \mathbb{Z}_q^*$.

Similarly, $x \in \mathbb{Z}_n^*$ is a quadratic residue modulo $n$ if and only if $x \bmod p$ and $x \bmod q$ are quadratic residues modulo $p$ and $q$. From subtask **a)** we know that $|\mathcal{QR}_p| = \frac{p-1}{2}$ and analogously that there are $\frac{q-1}{2}$ quadratic residues in $\mathbb{Z}_q^*$. Therefore, we see that there are $\frac{p-1}{2}\frac{q-1}{2}$ quadratic residues in $\mathbb{Z}_n^*$, which are $\frac{1}{4}$ of its elements.

**d)** Recall from **c)** the isomorphism $\phi$, such that $\mathcal{QR}_n \cong \mathcal{QR}_p \times \mathcal{QR}_q$. From **b)** we know how to compute square roots modulo $p$ and $q$. Let $a = y^{\frac{p+1}{4}} \bmod p$ and $b = y^{\frac{q+1}{4}} \bmod q$. Then, $(a, b) \in \mathcal{QR}_p \times \mathcal{QR}_q$ is a square root of $\phi(y) = (y \bmod p, y \bmod q) \in \mathcal{QR}_p \times \mathcal{QR}_q$. Hence, $f$ is subjective on $\mathcal{QR}_n$ and thus a bijection. In summary, the inverse $f^{-1}\colon \mathcal{QR}_n \to \mathcal{QR}_n$ is given by

$$f^{-1}(y) = \phi^{-1}\left(y^{\frac{p+1}{4}} \bmod p, y^{\frac{q+1}{4}} \bmod q\right)$$

and is efficiently computable if $p$ and $q$ are known.

**e)** The idea here is to find two square roots $x, y$ modulo $n$ of $z$ such that $x \not\equiv \pm y \pmod{n}$. Then $x^2 - y^2 = (x+y)(x-y) \equiv 0 \pmod{n}$ and it follows that $\gcd(x-y, n)$ is one of the prime divisors of $n$.

Let $X$ be a uniformly distributed random variable over $\mathbb{Z}_n^*$. Then, $Z = X^2$ is uniformly distributed over $\mathcal{QR}_n$ since $x \mapsto x^2\colon \mathbb{Z}_n^* \to \mathcal{QR}_n$ is 4-to-1. So, if $A$ denotes the algorithm from the exercise sheet and $Y = A(Z) = A(X^2)$, then

$$\Pr(Y \in \mathcal{QR}_n \wedge Y^2 \equiv X^2) = \alpha.$$

Moreover, observe that since $x \mapsto x^2\colon \mathbb{Z}_n^* \to \mathcal{QR}_n$ is 4-to-1, $\Pr(\pm X \notin \mathcal{QR}_n) = \frac{1}{2}$. However, clearly $Y \in \mathcal{QR}_n$ and $\pm X \notin \mathcal{QR}_n$ together imply that $Y \in \mathbb{Z}_n^* \wedge Y \not\equiv \pm X \pmod{n}$. Hence,

$$\begin{aligned}
&\Pr(Y \in \mathbb{Z}_n^* \wedge Y \not\equiv \pm X \wedge Y^2 \equiv X^2) \\
&\quad \geq \frac{1}{2}\Pr(Y \in \mathbb{Z}_n^* \wedge Y \not\equiv \pm X \wedge Y^2 \equiv X^2 \mid \pm X \notin \mathcal{QR}_n) \\
&\quad \geq \frac{1}{2}\Pr(Y \in \mathcal{QR}_n \wedge Y^2 \equiv Z \mid \pm X \notin \mathcal{QR}_n) \\
&\quad = \frac{1}{2}\Pr(Y \in \mathcal{QR}_n \wedge Y^2 \equiv Z) \\
&\quad = \frac{1}{2}\alpha,
\end{aligned}$$

where in the third step we used that $Z$ remains uniformly distributed when we condition on $\pm X \notin \mathcal{QR}_n$ since $x \mapsto x^2\colon (\mathbb{Z}_n^* \setminus \pm\mathcal{QR}_n) \to \mathcal{QR}_n$ is 2-to-1.

The success probability can be amplified as follows. The probability that we succeed in finding a prime divisor of $n$ with $k$ independent runs of our algorithm is equal to $1 - (1 - \frac{1}{2}\alpha)^k$. This can be made arbitrarily close to 1 for sufficiently large $k$.

[1] That means, $\phi$ is a bijection that is compatible with addition and multiplication, i.e., $\phi(x+y) = \phi(x) + \phi(y)$ and $\phi(x \cdot y) = \phi(x) \cdot \phi(y)$.