

MODERN CRYPTOGRAPHY

Bachelor Computer Science, University of Amsterdam, 2019/20

TEACHER: Christian Schaffner, TA: Jan Czajkowski, Jana Sotáková

Problem Set 6

Problem 1: El Gamal encryption

As in Example 11.17 in [KL], let \mathbb{G} be the subgroup of \mathbb{Z}_{167}^* generated by $g = 4$. We have that the order $q = |\mathbb{G}| = 83$ is prime. Let the secret key be $x = 23 \in \mathbb{Z}_{83}$ and so the public key is $pk = \langle p, q, g, h \rangle = \langle p, q, g, g^x \rangle$

- (a) Compute the h component in the public key.
- (b) Compute the encryption of message $m = 19 \in \mathbb{G}$ with randomness $y = 44$.
- (c) Decrypt the ciphertext $\langle c_1, c_2 \rangle = \langle 132, 44 \rangle$.

Problem 2: RSA

- (a) **RSA encryption** As in Example 11.27 in [KL], say GenRSA outputs $(N, e, d) = (1005973, 89, d)$. Note that $1005973 = 997 \cdot 1009$.

1. Encrypt the message $m = 1234 \in \mathbb{Z}_{1005973}^*$
2. Compute the private key (N, d) corresponding to the public key $(N, e) = (1005973, 89)$.
3. Decrypt the ciphertext $c = 530339$.

- (b) **Attacks on Plain RSA**
 1. For the RSA public key $(N, e) = (10000799791, 3)$, decrypt the ciphertext $c = 1000000$. Can you do it without factoring N ?
 2. Suppose we would like to use plain RSA with public exponent $e = 3$ as public-key encryption in a hybrid scheme together with AES-256 in CBC mode. We choose N to have roughly 2048 bits. Use the previous subexercise to argue the insecurity of this hybrid scheme.

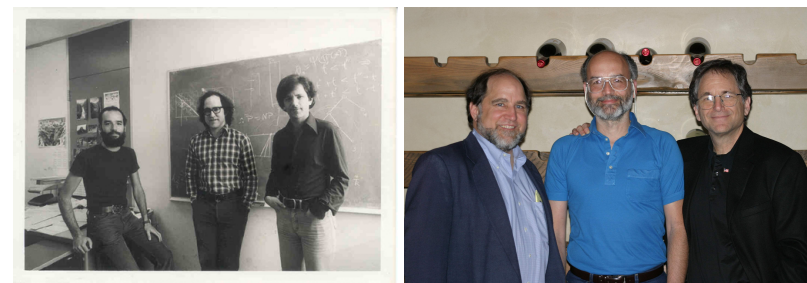


Figure 1: Adi Shamir, Ron Rivest, and Len Adleman as MIT-students and in 2003

Image credit: <http://www.ams.org/samplings/feature-column/fcarc-internet>, <http://www.usc.edu/dept/molecular-science/RSA-2003.htm>

Problem 3: CCA security of multiple encryptions

Claim 11.7 in [KL] states that if $\Pi = (\text{Gen}, \text{Enc}, \text{Dec})$ is a CPA-secure public-key encryption scheme for fixed-length messages, then the new encryption scheme $\Pi' = (\text{Gen}, \text{Enc}', \text{Dec}')$ with $\text{Enc}'_{pk}(m_1 \| m_2 \| \dots \| m_\ell) = \text{Enc}_{pk}(m_1) \| \text{Enc}_{pk}(m_2) \| \dots \| \text{Enc}_{pk}(m_\ell)$ is CPA secure for arbitrary-length messages.

Show that Claim 11.7 does not hold in the setting of CCA-security: Exhibit a concrete attack on a scheme $\Pi' = (\text{Gen}, \text{Enc}', \text{Dec}')$ constructed from a fixed-length CCA secure encryption scheme $\Pi = (\text{Gen}, \text{Enc}, \text{Dec})$ by defining

$$\text{Enc}'_{pk}(m_1 \| m_2 \| \dots \| m_\ell) = \text{Enc}_{pk}(m_1) \| \text{Enc}_{pk}(m_2) \| \dots \| \text{Enc}_{pk}(m_\ell)$$

Make sure you specify the whole CCA attacker \mathcal{A} explicitly: what are the challenge messages, what are the encryption/decryption oracle queries, how is the guess bit b' computed? Then compute $\Pr[\text{PubK}_{\mathcal{A}, \Pi'}^{\text{cca}}(n) = 1]!$

Problem 4: RSA Signatures

(a) **Plain RSA Signatures** Say the public key is $\langle N, e \rangle = \langle 91, 11 \rangle$.

1. Verify that $(43, 36)$ is a valid message-signature pair.
2. Compute $\phi(N)$.
3. Calculate the private key d .
4. Sign the message $m = 28$.

(b) **Insecurity of plain RSA Signatures** In Section 12.4.1 we showed an attack on the plain RSA signature scheme in which an attacker forges a signature on an arbitrary message using two signing queries. Show how an attacker can forge a signature on an arbitrary message using a *single* signing query.

Hint: Use the no-query and the two-query attacks.

Problem 5: One-time-secure signature scheme?

A signature scheme is one-time-secure if no PPT adversary making a *single* query can output a valid forgery.

Let f be a one-way permutation (it is hard to calculate the inverse of f). Consider the following signature scheme for messages in the set $\{1, \dots, n\}$:

- To generate keys, choose uniform $x \in \{0, 1\}^n$ and set $y := f^{(n)}(x)$ (where $f^{(i)}(\cdot)$ refers to i -fold iteration of f , and $f^{(0)}(x) = x$). The public key is y and the private key is x .
- To sign message $i \in \{1, \dots, n\}$, output $f^{(n-i)}(x)$.
- To verify signature σ on message i with respect to public key y , check whether $y = f^{(i)}(\sigma)$.

- (a) Show that the verification procedure will output 1 for every legal message-signature pair.
- (b) Show that the above is not a one-time-secure signature scheme. Given a signature on a message i , for what messages j can an adversary output a forgery?

(c) Prove that no PPT adversary given a signature on i can output a forgery on any message $j > i$ except with negligible probability.

Problem 6: El-Gamal variant

Consider the following public-key encryption scheme. The public key is (G, q, g, h) and the private key is x , generated exactly as in the El-Gamal encryption scheme. In order to encrypt a bit b , the sender does the following:

1. If $b = 0$ then choose independent random $y, z \leftarrow \mathbb{Z}_q$, compute $c_1 = g^y$ and $c_2 = g^z$, and set the ciphertext equal to (c_1, c_2) .
2. If $b = 1$ then choose a random $y \leftarrow \mathbb{Z}_q$ and compute $c_1 = g^y$ and $c_2 = h^y$. The ciphertext is (c_1, c_2) .

Show that it is possible to decrypt efficiently given knowledge of x . Prove that this encryption scheme is CPA-secure if the decisional Diffie-Hellman problem is hard relative to \mathcal{G} .

★ **Problem 7: Perfectly secure public-key encryption?**

Assume a public-key encryption scheme for single-bit messages with no decryption error. Show that, given pk and a ciphertext c computed via $c = \text{Enc}_{pk}(m)$, it is possible for an unbounded adversary to determine m with probability 1.

★ **Problem 8: Another one-time secure signature scheme**

Let f be a permutation and $f^{(i)}(x)$ the i -fold iteration of f , and $f^{(0)}(x) := x$. Let us consider the following signature scheme $\Pi = (\text{Gen}, \text{Sign}, \text{Vrfy})$ for messages $m \in \{1, \dots, p\}$ with $p = p(n)$ polynomial in n .

$\text{Gen}(1^n)$: Choose $sk_1, sk_2 \in_R \{0, 1\}^n$, $pk_1 := f^p(sk_1)$ and $pk_2 := f^p(sk_2)$.
Set $sk := (sk_1, sk_2)$ and $pk := (pk_1, pk_2)$.

$\text{Sign}_{sk}(m)$: Compute $\sigma_1 := f^{(p-m)}(sk_1)$ and $\sigma_2 := f^{(m-1)}(sk_2)$.
Return $\sigma := (\sigma_1, \sigma_2)$.

$\text{Vrfy}_{pk}(m, \sigma)$: If $pk_1 = f^{(m)}(\sigma_1)$ and $pk_2 = f^{(p-m+1)}(\sigma_2)$ return 1, else return 0.

- (a) Show that Π is correct.
- (b) Prove that Π is a one-time-secure signature scheme, if f is a one-way permutation.

★ **Problem 9: Hash-based signatures**

Read Section 12.6 in [KL] to learn about hash-based signatures, one of the prime candidates for a signature scheme which remains secure against quantum attackers.