Cryptography 2017
Lecture 7
- RSA, ElGamal, Intro to Protocols
What have seen? What are we discussing today? What is coming later?

Lecture 6

- Introduction to number theory

Lecture 7

- Recap on RSA
- Security concepts: IND-CPA & IND-CCA
- The Dlog problems
- Recap on Group Theory
- ElGamal encryption scheme
- Introduction to protocols

Lecture 8

- Key exchange protocols (Diffie-Hellman)
- Identification protocol (Fiat-Shamir)
- Quick look at Zero-Knowledge (ZK) and Sigma (Σ) protocols
Textbook RSA encryption Scheme (Rivest-Shamir-Adleman)

Useful for the exercises/exams!

Textbook RSA

KeyGen($\lambda$) $\rightarrow$ (pk, sk)

1. generate two distinct $\lambda$-bit primes $p$ and $q$, compute $N = pq$ and $\Phi(N)$.
2. choose an integer $e \xleftarrow{\$} \mathbb{Z}_{\Phi(N)}$ such that $\text{GCD}(e, \Phi(N)) = 1$ and compute its (modular) inverse $d = e^{-1} \mod \Phi(N)$.
3. set: $\text{pk} = (N, e)$ and $\text{sk} = (N, d)$

$$\text{Enc(pk, m)} \rightarrow c : \text{compute} \quad c = m^e \pmod{N}$$

$$\text{Dec(sk, m)} \rightarrow m : \text{compute} \quad m = c^d \pmod{N}$$

$\Phi$ denotes Euler's phi (or Totient) function i.e., the number of positive integers less than $n$ and relatively prime to $n$ (more info about this in lec06).

For now remember $\Phi(N) = \Phi(pq) = (p-1)(q-1)$.

It holds $\mathbb{Z}_N = \{0, 1, 2, \ldots, N-1\}$ where $N$ is a positive integer.
Textbook RSA: Correctness Property

Does the decryption work correctly?

**Correctness Property:**

\[
\text{Dec}(sk, \text{Enc}(pk, m)) = m
\]

Let \( c = \text{Enc}(pk, m) = m^e \pmod{N} \)
Then,

\[
\text{Dec}(sk, c) = c^d \pmod{N} \equiv (m^e)^d \pmod{N} \equiv m^{ed} \pmod{N} \quad (1)
\]

But it holds: \( ed = 1 \pmod{\Phi(N)} \).
This implies, there is a \( k \) such that \( k\Phi(N) + 1 = ed \) \( (2) \)

We also have by **Euler Theorem**: \( a^{\Phi(N)} \equiv 1 \pmod{N} \Rightarrow a^{\Phi(N)+1} \equiv a \pmod{N} \) \( (3) \)

Thus, from (1), (2) and (3) we get:

\[
\text{Dec}(sk, c) = m^{ed} \pmod{N} \equiv m^{k\Phi(N)+1} \pmod{N} \equiv m \pmod{N}
\]
What is a ‘good’ cipher?

Recall - Principles of Modern Cryptography

Describe clearly what you want to achieve so that you know if and when you have achieved it!

- **Formal Definitions**
- Precise Assumptions
- Proof of Security
How can we define the security of a PKE cipher?

Step 1: Decide what the attacker can do (power) & what he wants to achieve (goal).

Power

- Know all the public information (i.e., algorithms, pk, public parameters)
- Encrypt any message (possible with the pk)
- Decrypt some chosen ciphertexts! (CCA)

Goal

- Decrypt ciphertexts.
- Find the secret key.

**IND-CPA**
Stands for indistinguishability **chosen plaintext attack**. The most basic security requirement for PKE.

**IND-CCA**
Stands for indistinguishability **chosen ciphertext attack**. Similar to IND-CPA but with the ability of the attacker $A$ to decrypt selected ciphertexts.

**IND - indistinguishability:** This implies that no adversary $A$ can identify the plaintext message of a ciphertext, even if the ciphertext comes from one of two possible messages selected by $A$. Formally, we say that $A$ has negligible advantage in guessing which message was encrypted.
Security notion: IND-CPA (Chosen Plaintext Attack)

\[ \text{Attacker } A \]

\[ \text{Challenger } C \]

\[ \text{KeyGen}(\lambda) \rightarrow (pk, sk) \]

\[ A \text{ can encrypt polynomially many messages that he chooses} \]

\[ m_0, m_1 \leftarrow \mathcal{D M} \]

\[ \text{len}(m_0) = \text{len}(m_1) \]

\[ b \leftarrow \mathcal{R} \{0, 1\} \]

\[ c = \text{Enc}(pk, m_b) \]

\[ \text{if } b=0, \text{ c is the encryption of } m_0 \]

\[ \text{if } b=1, \text{ c is the encryption of } m_1 \]

\[ b' \in \{0, 1\} \]

\[ \text{a guess for } b \]

\[ \text{Challenge Phase} \]

**Definition**

A public key cipher \((\text{KeyGen}, \text{Enc}, \text{Dec})\) is secure under CPA if for any ‘efficient’ adversary it holds: \( P(b' = b) < \frac{1}{2} + \text{negligible} \)
Security notion: IND-CCA (Chosen Ciphertext Attack)

Attacker $\mathcal{A}$ can encrypt polynomially many messages that he chooses.

\[ \text{KeyGen}(\lambda) \rightarrow (pk, sk) \]

Challenger $\mathcal{C}$

\[ \text{KeyGen}(\lambda) \rightarrow (pk, sk) \]

An encryption scheme that is IND-CCA is also IND-CPA (since IND-CCA contains the definition of IND-CPA).

Definition

A public key cipher $\langle \text{KeyGen}, \text{Enc}, \text{Dec} \rangle$ is secure under CCA if for any 'efficient' adversary it holds: $P(b' = b) < \frac{1}{2} + \text{negligible}$
Reminder: Textbook RSA Security

Textbook RSA is deterministic!

i.e., for a fixed key pair \((pk, sk)\), a message \(m\), always encrypts to the same ciphertext.

Vulnerable to many attacks!

**Brute Force attacks:** Since textbook RSA is deterministic, if the message \(m\) is chosen from a small list of possible values, then it is possible to determine \(m\) from the ciphertext \(c = m^e \pmod{N}\) by trying each value of \(m\).

**Question:** What if the message space is large, is it secure? **NO!**

Other attacks:

- Meet-in-the-middle attack.
- Chosen Plaintext attacks (we can distinguish \(m\) from \(m'\), next slide).
- Chosen Ciphertext attacks.
RSA is not semantically secure (under IND-CPA)

**Exercise:** Prove that the textbook-RSA encryption scheme is not secure under IND-CPA

**Solution:** Explain how textbook RSA and IND-CPA work.

The adversary chooses two messages \( m_0 \neq m_1 \) with \( \text{len}(m_0) = \text{len}(m_1) \).

He computes
\[
\begin{align*}
c_0 &= \text{Enc}(pk, m_0) \\
c_1 &= \text{Enc}(pk, m_1)
\end{align*}
\]

Let \( W_0 \) be the event that \( C \) chooses \( b = 0 \) and \( A \) outputs \( b' = 0 \).
Let \( W_1 \) be the event that \( C \) chooses \( b = 1 \) and \( A \) outputs \( b' = 0 \).

\( A \) has the **public key** and the encryption is **deterministic**, so he knows that for \( m_0 \) it holds \( c = c_0 \) and for \( m_1 \) it holds \( c = c_1 \).

\( A \) can output \( b' = 0 \), when \( c = c_0 \) and \( b' = 1 \) when \( c = c_1 \).

When \( C \) chooses \( b = 0 \), \( A \) outputs \( b' = 0 \).
When \( C \) chooses \( b = 1 \), \( A \) outputs \( b' = 1 \).

Then, we have: \(|P(W_0) - P(W_1)| = |1 - 0| = 1|
How about a randomised PKE?

The paper published by ElGamal in 1985 contained the basis for:
- the ElGamal Encryption Scheme, and the
- DSS (Digital Signature Standard) adopted by NIST
Recap on group theory

Definition

A group $G$ is a set together with an operation $\cdot$ satisfying the following properties:

1. Closure: $\forall g, h \in G$ it holds that $g \cdot h \in G$
2. Associativity: $\forall g, h, k \in G$ it holds that $(g \cdot h) \cdot k = g \cdot (h \cdot k)$
3. Identity: there exists an $e \in G$ such that $e \cdot g = g \cdot e = g$, $\forall g \in G$
4. Inverse: $\forall g \in G$ there exists a $h \in G$ such that $h \cdot g = e = g \cdot h$

Examples of groups: $(\mathbb{Z}_N, +), (\mathbb{Z}_N^*, \cdot)$

Remember:

- $\mathbb{Z}_N = \{0, 1, 2, \ldots, N - 1\}$. Example: $\mathbb{Z}_6 = \{0, 1, 2, 3, 4, 5\}$
- $\mathbb{Z}_N^*$ is the set of integers modulo $N$ and relatively prime to $N$, that is: $\mathbb{Z}_N^* = \{a \in \mathbb{Z}_N \text{ such that } \text{GCD}(a, N) = 1\}$. Example: $\mathbb{Z}_6^* = \{1, 5\}$
Recap on group theory

Definition

- The order of a group $G$ is the number of elements that belong in the group. It is denoted as: $\text{ord}(G)$ or $|G|$.
  
  Example: For $\mathbb{Z}_{11}^* = \{1, 2, 3, 4, 5, 6, 7, 8, 9, 10\}$ it holds: $|\mathbb{Z}_{11}^*| = 10$.

- The order of an element $g \in G$ (multiplicative group) is the smallest possible integer $n$ such that: $g^n = 1 \in G$. It is denoted as: $g \in G$ as $\text{ord}(g)$ or $|g|$.
  
  Example: For $2 \in \mathbb{Z}_{11}^*$ it holds $2^{10} = 1024 = 93 \times 11 + 1 = 1 \pmod{11}$.
  It holds $|2| = 10$.

Facts about the order of a group element

- For any $g \in G$ it holds: $g^{\text{ord}(G)} = 1$.

- If the order of a group element is equal to the order of a group set i.e., $|g| = |G|$ then $g \in G$ is called generator of $G$ and it holds:
  
  $\langle g \rangle = \{1 = g^0, g^1, g^2, g^3, \ldots, g^{|G|}\}$.

  Example: 2 is the generator of $\mathbb{Z}_{11}^*$ since it holds $2^0 = 1, 2^1 = 2, 2^2 = 4, 2^3 = 8,$ $2^4 = 5, \ldots,$ $2^{10} = 1024 = 1$ (remember everything $\pmod{11}$).

Definition

A group $G$ that has a generator $g \in G$ is called cyclic. Example: $\mathbb{Z}_{11}^*$ is cyclic!

Note: If $\mathbb{Z}_N$ is cyclic it has $\Phi(N)$ generators and if $\mathbb{Z}_N^*$ is cyclic then it has $\Phi(\Phi(N))$ generators!
Reminder: What is a ‘good’ cipher?

SEEN in LEC02

Recall - Principles of Modern Cryptography

Describe clearly what you want to achieve so that you know if and when you have achieved it!

- Formal Definitions
- Precise Assumptions
- Proof of Security
Computationally hard problems - The Dlog problem

Modern cryptography is often based on the assumption that a certain problem cannot be solved in polynomial time (i.e., a computationally hard problem).

The Discrete Logarithm (Dlog) Problem

- Given a cyclic group $G$, with generator $g$ ($G = \langle g \rangle$) and a random element $h \in G$, compute $x \in \{0, 1, \ldots, \text{ord}(G)\}$ such that $g^x = h$.

Example:

<table>
<thead>
<tr>
<th>$g^x \mod 11$</th>
<th>$\text{Dlog}_2(\cdot)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$x$</td>
<td>0</td>
</tr>
<tr>
<td>$D \log_2(\cdot)$</td>
<td>2^0</td>
</tr>
</tbody>
</table>

Remember:
The Dlog problem is a NP and BQP problem and used extensively in cryptography including in RSA, El Gamal encryption, DH key exchange, El Gamal signatures and Elliptic curve cryptography.
Other relevant problems - The Diffie-Hellman assumptions

Consider a cyclic group $G$ of order $q$.

The Computational Diffie Hellman assumption:

Given $(g, g^a, g^b)$ for a randomly chosen generator $g \in G$ and random $a, b \in \{0, \ldots, q - 1\}$ it is computationally intractable to compute the value $a^b$. 
Consider a cyclic group $G$ of order $q$.

**The Computational Diffie Hellman assumption:**
Given $(g, g^a, g^b)$ for a randomly chosen generator $g \in G$ and random $a, b \in \{0, \ldots, q - 1\}$ it is computationally intractable to compute the value $g^{ab}$.

**The Decisional Diffie Hellman assumption:**
Given $g^a$ and $g^b$ for uniformly and independently chosen $a, b \in \mathbb{Z}_q$ the value $g^{ab}$ “looks like” a random element in $G$. 
ElGamal Encryption Scheme

ElGamal

- **KeyGen**($\lambda$) $\rightarrow$ (pk, sk):

  1. generate a description of a cyclic group $G = \langle g \rangle$ of order $q$ (that is a $\lambda$-bits long integer)
  2. choose a random value $x \in \{1, 2, \ldots, q - 1\}$, and compute $h = g^x$
  3. set: pk = ($G, g, q, h$) and sk = (x)

- **Enc**(pk, m) $\rightarrow$ c (function from G to G)

  1. pick a random $r \in \{1, 2, \ldots, q - 1\}$ and compute $c_1 = g^r$
  2. compute $c_2 = m \cdot h^r \in G$, the ciphertext is $c = (c_1, c_2)$

- **Dec**(sk, c) $\rightarrow$ m (function from G to G)

  1. compute $k = c_1^x$
  2. decrypt $m = c_2k^{-1} = c_2c_1^{-x}$
Do ciphertexts decrypt correctly with ElGamal encryption?

**Correctness Property:**

\[ \text{Dec}(sk, \text{Enc}(pk, m)) = m \]

Let \( c = (c_1, c_2) \) be the ElGamal encryption of \( m \), i.e.,

\[
\begin{align*}
    c_1 &= g^r, \\
    c_2 &= m \cdot h^r, \\
    h &= g^x
\end{align*}
\]

Then, we have:

\[
\begin{align*}
    \text{Dec}(sk, c) &= c_2 k^{-1} = c_2 c_1^{-x} \\
                       &= (m \cdot h^r)(g^r)^{-x} \\
                       &= (m(g^x)^r)(g^r)^{-x} \\
                       &= mg^{xr} g^{-xr} \\
                       &= m
\end{align*}
\]
ElGamal Encryption: an example

Let \( q = 11 \), consider \( G = \mathbb{Z}_q^* = \{1, 2, \ldots, 10\} \) and \( g = 2 \)

**KeyGen**

\[
\begin{align*}
\text{sk} &= x \overset{R}{\leftarrow} \mathbb{Z}_q^* \\
\text{pk} &= (G, g, q, h) \\
\text{sk} &= 3 \\
\text{pk} &= (\mathbb{Z}_{11}^*, 2, 11, h) \\
h &= g^x = 2^3 = 8 \quad \text{(mod 11)} \\
\text{pk} &= (\mathbb{Z}_{11}^*, 2, 11, 8)
\end{align*}
\]

**Encrypt**

Let's encrypt \( m = 10 \)

\[
\begin{align*}
r &\overset{R}{\leftarrow} \mathbb{Z}_q^* \\
c_1 &= g^r \\
c_2 &= m \cdot h^r \\
c &= (c_1, c_2)
\end{align*}
\]

\[
\begin{align*}
r &= 6 \\
c_1 &= 2^6 = 9 \quad \text{(mod 11)} \\
c_2 &= 10 \cdot 8^6 \quad \text{(mod 11)} \\
&= (-1) \cdot ((-3)^2)^3 \\
&= (-9)^3 = -(-2)^3 \\
&= 8 \quad \text{(mod 11)} \\
c &= (9, 8)
\end{align*}
\]

**Decrypt**

\[
\begin{align*}
k &= c_1^x \\
k &= 9^3 = (-2)^3 = -8 \\
&= 3 \quad \text{(mod 11)} \\
m &= c_2 k^{-1}
\end{align*}
\]
ElGamal Encryption: an example

Let $q = 11$, consider $G = \mathbb{Z}_q^* = \{1, 2, \ldots, 10\}$ and $g = 2$

### KeyGen

- $sk = x \leftarrow \mathbb{Z}_q^*$
- $pk = (G, g, q, h)$
- $sk = 3$
- $pk = (\mathbb{Z}_{11}^*, 2, 11, h)$
- $h = g^x = 2^3 = 8 \pmod{11}$
- $pk = (\mathbb{Z}_{11}^*, 2, 11, 8)$

### Encrypt

- Let's encrypt $m = 10$
- $r \leftarrow \mathbb{Z}_q^*$
- $c_1 = g^r$
- $c_2 = m \cdot h^r$
- $c = (c_1, c_2)$
- $r = 6$
- $c_1 = 2^6 = 9 \pmod{11}$
- $c_2 = 10 \cdot 8^6 \pmod{11}$
- $= (-1) \cdot ((-3)^2)^3$
- $= (-9)^3 = -(-2)^3$
- $= 8 \pmod{11}$
- $c = (9, 8)$

### Decrypt

- $k = c_1^x$
- $k = 9^3 = (-2)^3 = -8$
- $= 3 \pmod{11}$
- $m = c_2 k^{-1}$
- $m = c_2 \cdot 3 \pmod{11}$
- $m = 8 \cdot 3 \pmod{11}$
- $m = 24 \pmod{11}$
- $m = 10$
ElGamal Encryption: an example

Let $q = 11$, consider $G = \mathbb{Z}_q^* = \{1, 2, \ldots, 10\}$ and $g = 2$

**KeyGen**

$sk = x \overset{R}{\leftarrow} \mathbb{Z}_q^*$  \hspace{1cm} $pk = (G, g, q, h)$

$sk = 3$

$h = g^x = 2^3 = 8 \pmod{11}$

$pk = (\mathbb{Z}_q^*_{11}, 2, 11, h)$

**Encrypt**

Lets encrypt $m = 10$

$r \overset{R}{\leftarrow} \mathbb{Z}_q^*$

$c_1 = g^r$  \hspace{1cm}  $r = 6$

$c_1 = 2^6 = 9 \pmod{11}$

$c_2 = m \cdot h^r$  \hspace{1cm}  $c_2 = 10 \cdot 8^6 \pmod{11}$

$= (−1) \cdot ((−3)^2)^3$

$= (−9)^3 = −(−2)^3$

$= 8 \pmod{11}$

$c = (c_1, c_2)$  \hspace{1cm}  $c = (9, 8)$

**Decrypt**

$k = c_1^x$  \hspace{1cm}  $k = 9^3 = (−2)^3 = −8$

$k = 3 \pmod{11}$

$m = c_2 k^{-1}$

$m = 10 \cdot 3 \pmod{11}$

$m = 30 \pmod{11}$

$m = 8 \pmod{11}$
ElGamal Encryption: an example

Let \( q = 11 \), consider \( G = \mathbb{Z}_q^* = \{1, 2, \ldots, 10\} \) and \( g = 2 \)

<table>
<thead>
<tr>
<th>KeyGen</th>
</tr>
</thead>
<tbody>
<tr>
<td>( sk = x \overset{R}{\leftarrow} \mathbb{Z}_q^* )</td>
</tr>
<tr>
<td>( sk = 3 )</td>
</tr>
<tr>
<td>( h = g^x = 2^3 = 8 \pmod{11} )</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Encrypt</th>
</tr>
</thead>
<tbody>
<tr>
<td>刺 encrypt ( m = 10 )</td>
</tr>
<tr>
<td>( r \overset{R}{\leftarrow} \mathbb{Z}_q^* )</td>
</tr>
<tr>
<td>( c_1 = g^r )</td>
</tr>
<tr>
<td>( c_2 = m \cdot h^r )</td>
</tr>
<tr>
<td>( c = (c_1, c_2) )</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Decrypt</th>
</tr>
</thead>
<tbody>
<tr>
<td>( k = c_1^x )</td>
</tr>
<tr>
<td>( m = c_2 k^{-1} )</td>
</tr>
</tbody>
</table>
ElGamal Encryption: an example

Let $q = 11$, consider $G = \mathbb{Z}_q^* = \{1, 2, \ldots, 10\}$ and $g = 2$

<table>
<thead>
<tr>
<th>KeyGen</th>
</tr>
</thead>
<tbody>
<tr>
<td>$sk = x \overset{R}{\leftarrow} \mathbb{Z}_q^*$</td>
</tr>
<tr>
<td>$sk = 3$</td>
</tr>
<tr>
<td>$pk = (G, g, q, h)$</td>
</tr>
<tr>
<td>$pk = (\mathbb{Z}_{11}^*, 2, 11, h)$</td>
</tr>
<tr>
<td>$h = g^x = 2^3 = 8 \pmod{11}$</td>
</tr>
<tr>
<td>$pk = (\mathbb{Z}_{11}^*, 2, 11, 8)$</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Encrypt</th>
</tr>
</thead>
<tbody>
<tr>
<td>Lets encrypt $m = 10$</td>
</tr>
<tr>
<td>$r \overset{R}{\leftarrow} \mathbb{Z}_q^*$</td>
</tr>
<tr>
<td>$c_1 = g^r$</td>
</tr>
<tr>
<td>$c_2 = m \cdot h^r$</td>
</tr>
<tr>
<td>$c = (c_1, c_2)$</td>
</tr>
<tr>
<td>$r = 6$</td>
</tr>
<tr>
<td>$c_1 = 2^6 = 9 \pmod{11}$</td>
</tr>
<tr>
<td>$c_2 = 10 \cdot 8^6 \pmod{11}$</td>
</tr>
<tr>
<td>$= (-1) \cdot ((-3)^2)^3$</td>
</tr>
<tr>
<td>$= (-9)^3 = (-2)^3$</td>
</tr>
<tr>
<td>$= 8 \pmod{11}$</td>
</tr>
<tr>
<td>$c = (9, 8)$</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Decrypt</th>
</tr>
</thead>
<tbody>
<tr>
<td>$k = c_1^x$</td>
</tr>
<tr>
<td>$k = 9^3 = (-2)^3 = -8 = 3 \pmod{11}$</td>
</tr>
<tr>
<td>$m = c_2 k^{-1}$</td>
</tr>
<tr>
<td>$m = 8 \pmod{11}$</td>
</tr>
</tbody>
</table>
ElGamal Encryption: an example

Let $q = 11$, consider $G = \mathbb{Z}_q^* = \{1, 2, \ldots, 10\}$ and $g = 2$

**KeyGen**

<table>
<thead>
<tr>
<th>$sk = x \leftarrow \mathbb{Z}_q^*$</th>
<th>$pk = (G, g, q, h)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$sk = 3$</td>
<td>$pk = (\mathbb{Z}_{11}^*, 2, 11, h)$</td>
</tr>
<tr>
<td></td>
<td>$h = g^x = 2^3 = 8 \pmod{11}$</td>
</tr>
<tr>
<td></td>
<td>$pk = (\mathbb{Z}_{11}^*, 2, 11, 8)$</td>
</tr>
</tbody>
</table>

**Encrypt**

Let's encrypt $m = 10$

<table>
<thead>
<tr>
<th>$r \leftarrow \mathbb{Z}_q^*$</th>
<th>$r = 6$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$c_1 = g^r$</td>
<td>$c_1 = 2^6 = 9 \pmod{11}$</td>
</tr>
<tr>
<td>$c_2 = m \cdot h^r$</td>
<td>$c_2 = 10 \cdot 8^6 \pmod{11}$</td>
</tr>
<tr>
<td></td>
<td>$= (-1) \cdot ((-3)^2)^3$</td>
</tr>
<tr>
<td></td>
<td>$= (-9)^3 = -(-2)^3$</td>
</tr>
<tr>
<td></td>
<td>$= 8 \pmod{11}$</td>
</tr>
<tr>
<td>$c = (c_1, c_2)$</td>
<td>$c = (9, 8)$</td>
</tr>
</tbody>
</table>

**Decrypt**

$k = c_1^x$ $k = 9^3 = (-2)^3 = -8$

$m = c_2 k^{-1}$ $m = 3 \pmod{11}$
ElGamal Encryption: an example

Let $q = 11$, consider $G = \mathbb{Z}_q^* = \{1, 2, \ldots, 10\}$ and $g = 2$

**KeyGen**

$sk = x \overset{R}{\leftarrow} \mathbb{Z}_q^*$

$sk = 3$

$pk = (G, g, q, h)$

$h = g^x = 2^3 = 8 \pmod{11}$

$pk = (\mathbb{Z}_{11}^*, 2, 11, 8)$

**Encrypt**

Let's encrypt $m = 10$

$r \overset{R}{\leftarrow} \mathbb{Z}_q^*$

$c_1 = g^r$

$c_2 = m \cdot h^r$

$c = (c_1, c_2)$

$r = 6$

$c_1 = 2^6 = 9 \pmod{11}$

$c_2 = 10 \cdot 8^6 \pmod{11}$

$= (-1) \cdot ((-3)^2)^3$

$= (-9)^3 = -(-2)^3$

$= 8 \pmod{11}$

$c = (9, 8)$

**Decrypt**

$k = c_1^x$

$k = 9^3 = (-2)^3 = -8$

$= 3 \pmod{11}$

$m = c_2 k^{-1}$
ElGamal Encryption: an example

Let $q = 11$, consider $G = \mathbb{Z}^*_q = \{1, 2, \ldots, 10\}$ and $g = 2$

**KeyGen**

| $sk = x \xleftarrow{\$} \mathbb{Z}^*_q$ | $pk = (G, g, q, h)$ |
| $sk = 3$ | $pk = (\mathbb{Z}^*_q, 2, 11, h)$ |
| $h = g^x = 2^3 = 8 \pmod{11}$ | $pk = (\mathbb{Z}^*_q, 2, 11, 8)$ |

**Encrypt**

Let's encrypt $m = 10$

| $r \xleftarrow{\$} \mathbb{Z}^*_q$ | $r = 6$ |
| $c_1 = g^r$ | $c_1 = 2^6 = 9 \pmod{11}$ |
| $c_2 = m \cdot h^r$ | $c_2 = 10 \cdot 8^6 \pmod{11}$ |
| $c = (c_1, c_2)$ | $c = (9, 8)$ |

**Decrypt**

$k = c_1^x$

| $k = 9^3 = (-2)^3 = -8$ | $= 3 \pmod{11}$ |

$\hat{m} = c_2 k^{-1}$

| $\hat{m} = c_2 k^{-1}$ | $\hat{m} = 8 \pmod{11}$ |
ElGamal Encryption: an example

Let $q = 11$, consider $G = \mathbb{Z}_q^* = \{1, 2, \ldots, 10\}$ and $g = 2$

**KeyGen**

| $sk = x \overset{R}{\leftarrow} \mathbb{Z}_q^*$ | $pk = (G, g, q, h)$ |
| $sk = 3$ | $pk = (\mathbb{Z}_{11}^*, 2, 11, h)$ |
| $h = g^x = 2^3 = 8 \pmod{11}$ | $pk = (\mathbb{Z}_{11}^*, 2, 11, 8)$ |

**Encrypt**

Let's encrypt $m = 10$

| $r \overset{R}{\leftarrow} \mathbb{Z}_q^*$ | $r = 6$ |
| $c_1 = g^r$ | $c_1 = 2^6 = 9 \pmod{11}$ |
| $c_2 = m \cdot h^r$ | $c_2 = 10 \cdot 8^6 \pmod{11}$ |
| | $= (-1) \cdot ((-3)^2)^3$ |
| | $= (-9)^3 = -(-2)^3$ |
| | $= 8 \pmod{11}$ |
| $c = (c_1, c_2)$ | $c = (9, 8)$ |

**Decrypt**

| $k = c_1^x$ | $k = 9^3 = (-2)^3 = -8$ |
| $k = 3 \pmod{11}$ | $m = c_2 k^{-1}$ |

| $m = 3 \pmod{11}$ | $c = (9, 8)$ |
ElGamal Encryption: an example

Let \( q = 11 \), consider \( G = \mathbb{Z}_q^* = \{1, 2, \ldots, 10\} \) and \( g = 2 \)

**KeyGen**

\[
\begin{align*}
  sk &= x \overset{R}{\leftarrow} \mathbb{Z}_q^* \\
  pk &= (G, g, q, h) \\
  pk &= (\mathbb{Z}_{11}^*, 2, 11, h) \\
  h &= g^x = 2^3 = 8 \pmod{11} \\
  pk &= (\mathbb{Z}_{11}^*, 2, 11, 8)
\end{align*}
\]

**Encrypt**

Let's encrypt \( m = 10 \)

\[
\begin{align*}
  r &\overset{R}{\leftarrow} \mathbb{Z}_q^* \\
  c_1 &= g^r \\
  c_2 &= m \cdot h^r \\
  c &= (c_1, c_2)
\end{align*}
\]

\[
\begin{align*}
  r &= 6 \\
  c_1 &= 2^6 = 9 \pmod{11} \\
  c_2 &= 10 \cdot 8^6 \pmod{11} \\
  &= (-1) \cdot ((-3)^2)^3 \\
  &= (-9)^3 = -(-2)^3 \\
  &= 8 \pmod{11} \\
  c &= (9, 8)
\end{align*}
\]

**Decrypt**

\[
\begin{align*}
  k &= c_1^x \\
  k &= 9^3 = (-2)^3 = -8 \\
  &= 3 \pmod{11} \\
  m &= c_2 k^{-1} \\
  &= 8 \pmod{11}
\end{align*}
\]
ElGamal Encryption: an example

Let $q = 11$, consider $G = \mathbb{Z}^*_q = \{1, 2, \ldots, 10\}$ and $g = 2$

KeyGen

$sk = x \overset{R}{\leftarrow} \mathbb{Z}^*_q$
$pk = (G, g, q, h)$

$sk = 3$
$pk = (\mathbb{Z}^*_q, 2, 11, h)$

$s = g^x = 2^3 = 8 \pmod{11}$

Encrypt

$pk = (\mathbb{Z}^*_q, 2, 11, 8)$

$pk = (\mathbb{Z}^*_q, 2, 11, h)$

$r \overset{R}{\leftarrow} \mathbb{Z}^*_q$
$c_1 = g^r$
$c_2 = m \cdot h^r$
$c = (c_1, c_2)$

$r = 6$
$c_1 = 2^6 = 9 \pmod{11}$
$c_2 = 10 \cdot 8^6 \pmod{11}$
$c = (9, 8)$

Decrypt

$k = c_1^x$
$k = 9^3 = (-2)^3 = -8 = 3 \pmod{11}$

$m = c_2 k^{-1}$
$m = 8 \pmod{11}$
ElGamal Encryption: an example

Let \( q = 11 \), consider \( G = \mathbb{Z}_q^* = \{1, 2, \ldots, 10\} \) and \( g = 2 \)

**KeyGen**

\[
\begin{align*}
\text{sk} &= x \overset{R}{\leftarrow} \mathbb{Z}_q^* \\
\text{pk} &= (G, g, q, h) \\
\text{sk} &= 3 \\
\text{pk} &= (\mathbb{Z}_{11}^*, 2, 11, h) \\
h &= g^x = 2^3 = 8 \pmod{11} \\
\text{pk} &= (\mathbb{Z}_{11}^*, 2, 11, 8)
\end{align*}
\]

**Encrypt**

Let's encrypt \( m = 10 \)

\[
\begin{align*}
r &= 6 \\
\text{c}_1 &= g^r \\
\text{c}_2 &= m \cdot h^r \\
c &= (\text{c}_1, \text{c}_2)
\end{align*}
\]

\[
\begin{align*}
\text{c}_1 &= 2^6 = 9 \pmod{11} \\
\text{c}_2 &= 10 \cdot 8^6 \pmod{11} \\
 &= (-1) \cdot ((-3)^2)^3 \\
 &= (-9)^3 = -(-2)^3 \\
 &= 8 \pmod{11} \\
c &= (9, 8)
\end{align*}
\]

**Decrypt**

\[
\begin{align*}
k &= \text{c}_1^x \\
k &= 9^3 = (-2)^3 = -8 \\
 &= 3 \pmod{11} \\
m &= \text{c}_2 k^{-1} \\
m &= 8 \pmod{11}
\end{align*}
\]
ElGamal Encryption: an example

Let $q = 11$, consider $G = \mathbb{Z}_q^* = \{1, 2, \ldots, 10\}$ and $g = 2$

**KeyGen**

$$
\begin{align*}
\text{sk} &= x \overset{R}{\leftarrow} \mathbb{Z}_q^* \\
\text{pk} &= (G, g, q, h) \\
\text{sk} &= 3 \\
\text{pk} &= (\mathbb{Z}_{11}^*, 2, 11, h) \\
&= (\mathbb{Z}_{11}^*, 2, 11, 2^3 = 8 \pmod{11}) \\
h &= g^x = 2^3 = 8 \pmod{11}
\end{align*}
$$

**Encrypt**

Let's encrypt $m = 10$

$$
\begin{align*}
r &\overset{R}{\leftarrow} \mathbb{Z}_q^* \\
c_1 &= g^r \\
c_1 &= 2^6 = 9 \pmod{11} \\
c_2 &= m \cdot h^r \\
c_2 &= 10 \cdot 8^6 \pmod{11} \\
&= (-1) \cdot ((-3)^2)^3 \\
&= (-9)^3 = -(-2)^3 \\
&= 8 \pmod{11}
\end{align*}
$$

$c = (c_1, c_2) = (9, 8)$

**Decrypt**

$$
\begin{align*}
k &= c_1^x \\
k &= 9^3 = (-2)^3 = -8 \\
&= 3 \pmod{11}
\end{align*}
$$

$m = c_2 k^{-1} = 8 \cdot 3^{-1} = 8 \cdot 3 = 24 = 3 \pmod{11}$
ElGamal Encryption: an example

Let $q = 11$, consider $G = \mathbb{Z}^*_q = \{1, 2, \ldots, 10\}$ and $g = 2$

**KeyGen**

| sk = $x \overset{R}{\leftarrow} \mathbb{Z}^*_q$ | pk = $(G, g, q, h)$ |
| sk = 3 | pk = $(\mathbb{Z}^*_q, 2, 11, h)$ |

$h = g^x = 2^3 = 8 \pmod{11}$

$pk = (\mathbb{Z}^*_q, 2, 11, 8)$

**Encrypt**

Let's encrypt $m = 10$

| $r \overset{R}{\leftarrow} \mathbb{Z}^*_q$ | $r = 6$ |
| $c_1 = g^r$ | $c_1 = 2^6 = 9 \pmod{11}$ |
| $c_2 = m \cdot h^r$ | $c_2 = 10 \cdot 8^6 \pmod{11}$ |

$c = (c_1, c_2)$

$c = (9, 8)$

**Decrypt**

$k = 9^3 = (-2)^3 = -8 = 3 \pmod{11}$

$m = c_2 k^{-1}$
ElGamal Encryption: an example

Let $q = 11$, consider $G = \mathbb{Z}_q^* = \{1, 2, \ldots, 10\}$ and $g = 2$

<table>
<thead>
<tr>
<th>KeyGen</th>
</tr>
</thead>
<tbody>
<tr>
<td>$sk = x \overset{R}{\leftarrow} \mathbb{Z}_q^*$</td>
</tr>
<tr>
<td>$sk = 3$</td>
</tr>
<tr>
<td>$pk = (G, g, q, h)$</td>
</tr>
<tr>
<td>$h = g^x = 2^3 = 8 \pmod{11}$</td>
</tr>
<tr>
<td>$pk = (\mathbb{Z}_{11}^*, 2, 11, 8)$</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Encrypt</th>
</tr>
</thead>
</table>
| L
| Lets encrypt $m = 10$ |
| $r \overset{R}{\leftarrow} \mathbb{Z}_q^*$ |
| $c_1 = g^r$ |
| $c_2 = m \cdot h^r$ |
| $c = (c_1, c_2)$ |
| $r = 6$ |
| $c_1 = 2^6 = 9 \pmod{11}$ |
| $c_2 = 10 \cdot 8^6 \pmod{11}$ |
| $= (-1) \cdot ((-3)^2)^3$ |
| $= (-9)^3 = (-2)^3$ |
| $= 8 \pmod{11}$ |
| $c = (9, 8)$ |

<table>
<thead>
<tr>
<th>Decrypt</th>
</tr>
</thead>
<tbody>
<tr>
<td>$k = c_1^x$</td>
</tr>
<tr>
<td>$k = 9^3 = (-2)^3 = -8$</td>
</tr>
<tr>
<td>$= 3 \pmod{11}$</td>
</tr>
<tr>
<td>$m = c_2 k^{-1}$</td>
</tr>
<tr>
<td>$m = 8 \pmod{11}$</td>
</tr>
<tr>
<td>$c = (9, 8)$</td>
</tr>
</tbody>
</table>
ElGamal Encryption: an example

Let $q = 11$, consider $G = \mathbb{Z}_q^* = \{1, 2, \ldots, 10\}$ and $g = 2$

**KeyGen**

\[
\begin{align*}
\text{sk} &= x \overset{R}{\leftarrow} \mathbb{Z}_q^* \\
\text{pk} &= (G, g, q, h) \\
\text{sk} &= 3 \\
\text{pk} &= (\mathbb{Z}_{11}^*, 2, 11, h) \\
h &= g^x = 2^3 = 8 \pmod{11} \\
\text{pk} &= (\mathbb{Z}_{11}^*, 2, 11, 8)
\end{align*}
\]

**Encrypt**

Let's encrypt $m = 10$

\[
\begin{align*}
r &\overset{R}{\leftarrow} \mathbb{Z}_q^* \\
c_1 &= g^r \\
c_2 &= m \cdot h^r \\
c &= (c_1, c_2)
\end{align*}
\]

\[
\begin{align*}
r &= 6 \\
c_1 &= 2^6 = 9 \pmod{11} \\
c_2 &= 10 \cdot 8^6 \pmod{11} \\
&= (-1) \cdot ((-3)^2)^3 \\
&= (-9)^3 = -(-2)^3 \\
&= 8 \pmod{11}
\end{align*}
\]

\[
c = (9, 8)
\]

**Decrypt**

\[
\begin{align*}
k &= c_1^x \\
k &= 9^3 = (-2)^3 = -8 \\
&= 3 \pmod{11} \\
m &= c_2 k^{-1} \\
m &= 8 \cdot (3)^{-1}
\end{align*}
\]

EEA: stands for Extended Euclidean Algorithm.
ElGamal Encryption: an example

Let $q = 11$, consider $G = \mathbb{Z}_q^* = \{1, 2, \ldots, 10\}$ and $g = 2$

**KeyGen**

$sk = x \overset{R}{\leftarrow} \mathbb{Z}_q^*$

$pk = (G, g, q, h)$

$sk = 3$

$h = g^x = 2^3 = 8 \pmod{11}$

$pk = (\mathbb{Z}_{11}^*, 2, 11, 8)$

**Encrypt**

Let's encrypt $m = 10$

$r \overset{R}{\leftarrow} \mathbb{Z}_q^*$

$c_1 = g^r$

$c_2 = m \cdot h^r$

$c = (c_1, c_2)$

$r = 6$

$c_1 = 2^6 = 9 \pmod{11}$

$c_2 = 10 \cdot 8^6 \pmod{11}$

$= (-1) \cdot ((-3)^2)^3$

$= (-9)^3 = -(-2)^3$

$= 8 \pmod{11}$

$c = (9, 8)$

**Decrypt**

$k = c_1^x$

$k = 9^3 = (-2)^3 = -8$

$= 3 \pmod{11}$

$m = c_2 k^{-1}$

$m = 8 \cdot (3)^{-1}$ EEA

$= 8 \cdot 4 = 32 = -1$

EEA: stands for Extended Euclidean Algorithm.
ElGamal Encryption: an example

Let $q = 11$, consider $G = \mathbb{Z}_q^* = \{1, 2, \ldots, 10\}$ and $g = 2$

**KeyGen**

| sk = $x \overset{R}{\leftarrow} \mathbb{Z}_q^*$ | $pk = (G, g, q, h)$ |
| sk = 3 | $pk = (\mathbb{Z}_q^*, 2, 11, h)$ |
| $h = g^x = 2^3 = 8 \pmod{11}$ | $pk = (\mathbb{Z}_q^*, 2, 11, 8)$ |

**Encrypt**

Let's encrypt $m = 10$

| $r \overset{R}{\leftarrow} \mathbb{Z}_q^*$ | $r = 6$ |
| $c_1 = g^r$ | $c_1 = 2^6 = 9 \pmod{11}$ |
| $c_2 = m \cdot h^r$ | $c_2 = 10 \cdot 8^6 \pmod{11}$ |
| $c_2 = (-1) \cdot ((-3)^2)^3$ | $c_2 = (-9)^3 = -(-2)^3$ |
| $c_2 = 8 \pmod{11}$ | $c = (9, 8)$ |

**Decrypt**

| $k = c_1^x$ | $k = 9^3 = (-2)^3 = -8$ |
| $k = 3 \pmod{11}$ | $m = c_2 k^{-1}$ |
| $m = 8 \cdot (3)^{-1}$ EEA | $m = 8 \cdot 4 = 32 = -1$ |
| $m = 10 \pmod{11}$ | $EEA$: stands for Extended Euclidean Algorithm. |
Crypto Magic! El Gamal and homomorphism

The El Gamal encryption has the homomorphic property!

Let us encrypt two messages, \( m_1 \) and \( m_2 \) with El Gamal encryption (and the same key):

\[
\text{Enc}(m_1) = (g^{r_1}, h^{r_1} \cdot m_1) \\
\text{Enc}(m_2) = (g^{r_2}, h^{r_2} \cdot m_2)
\]

for a cyclic group \( G = \langle g \rangle \) of order \( q \) and \( h = g^x \).

\( pk = (G, g, q, h) \) and \( sk = (x), r_1, r_2, x \overset{R}{\leftarrow} G \).

Then it holds:

\[
\text{Enc}(m_1) \cdot \text{Enc}(m_2) = (g^{r_1}, h^{r_1} \cdot m_1) \cdot (g^{r_2}, h^{r_2} \cdot m_2) \\
= g^{r_1+r_2}, h^{r_1+r_2} \cdot m_1 \cdot m_2 = \text{Enc}(m_1 \cdot m_2).
\]

- Having the encryption of two messages you can get the encryption of their product without the secret key!
- Homomorphic encryption is useful for performing operations over encrypted data.

Think: Has textbook RSA the same property?
Question 3. (c) Show that El Gamal is not secure against a chosen ciphertext attack.

\[ A \text{ can encrypt polynomially many messages that he chooses} \]

\[ \text{for } i = 1, \ldots q \text{ do} \]
\[ \text{choose } \bar{c}_i \in C \]
\[ \text{learn } \bar{m}_i \]

\[ m_0, m_1 \xleftarrow{D} \mathcal{M} \]
\[ \text{len}(m_0) = \text{len}(m_1) \]

\[ c = \text{Enc}(pk, m_b) \]
\[ \text{if } b = 0, \text{ c is the encryption of } m_0 \]
\[ \text{if } b = 1, \text{ c is the encryption of } m_1 \]

\[ \bar{c}' = \text{Enc}(pk, m_b) \cdot \text{Enc}(pk, m') \]
\[ = \text{Enc}(pk, m_b \cdot m') \]
\[ \text{Output } b' \in \{0, 1\} \]

Quiz Question: What should the attacker do next?
Overview - What have we covered in Public Key Crypto?

Alice

- Plaintext
- Encryption
- Bob’s Public Key
- Ciphertext
- See you today at 5pm

Bob

- Secret Key
- Bob’s Private Key
- Decryption
- Ciphertext
- See you today at 5pm

Encryption Schemes
- Textbook RSA
- El Gamal

Other Material
- Public Key Infrastructure
- Key Management (Certification Authorities)

Security Concepts
- IND-CPA
- IND-CCA
- Integer Factorisation Problem
- Discrete Log Problem

Useful tools/theory
- Prime Numbers
- Primality testing
- Extended Euclidean Algorithm
- Euler Theorem
- Group Theory

November 20, 2017
Home Assignment 2 (announcement)

Home assignment 2, Cryptography course

0. Introduction

In this assignment you will work with the ElGamal encryption system. To gain an understanding of this system and the algorithms for modular exponentiation and extended gcd, you will do various computations by hand. Of course, this implies that we will need to work with very small numbers. In fact, the setting of the assignment, common for all of you, is $\mathbb{Z}_{23}^*$. The assignment has three parts.

**Deadline:** next Tuesday (November 21)!

**Remember:** You must upload at least one solution to Home Assignment 1 Before next Tuesday (21/11/2017) at midnight!

Try to have Assignment 1 accepted as soon as possible!
Protocol: definition

- Encryption schemes, signatures, MACs and hash functions are **cryptographic primitives**.
- To achieve useful services such as identification, authentication, key agreement, e-voting, share common secrets...
- We need **cryptographic protocols** (that employ cryptographic primitives)
- **Example of protocols**: TLS (Transport Layer Security) to achieve secure web connections (HTTPS).

**Definition**

A **cryptographic protocol** is a distributed algorithm describing precisely the interactions between two or more entities, and achieving certain security objectives.
Protocols: notations

To describe the protocols in this lecture we use the following notation:

- **A, B**: names of parties involved in the protocol
- **S**: Trusted Third Party (TTP), e.g., a Key Distribution Centre (KDC)
- **C(A)**: C pretending to be A
- **N_A**: nonce (i.e., number used once) generated by party A
- **k_{AB}**: shared (symmetric) key for communication between A and B
- **\{m\}_{k_{AB}}**: the message \(m\) is encrypted using the key \(k_{AB}\)
- **m_1|m_2**: \(m_1\) concatenated with \(m_2\) (i.e., \(m_2\) is appended to \(m_1\))
Key Exchange Protocols: Why do we need them?

**Goal:** to enable two (or more) parties to establish a common secret key, in a secure way!
A Simple Key Exchange Protocol

I want a secret key shared with Bob

Hi, it’s Alice, let’s talk using this secret key

This protocol has multiple problems

Quiz Question: Which of the following problems?

1. S knows all the session keys!
2. Protocol insecure against replay attacks
3. Bob is not sure he is talking to Alice
4. Alice is not sure she is talking to Bob
5. An eavesdropper can learn the shared-key
6. Bob can impersonate Alice
Key-Exchange Protocols

Roger Needham

“Crypto protocols are three line programs that people still manage to get wrong!”
The Needham-Schroeder protocol

- Forerunner of Kerberos (used in e.g. Windows and Mac OS X).

1. \textbf{A} \rightarrow \textbf{S} : A, B, N_A \\
2. \textbf{S} \rightarrow \textbf{A} : \{N_A, B, K_{AB}, \{K_{AB}, A\}_K_{BS}\}_K_{AS} \\
3. \textbf{A} \rightarrow \textbf{B} : \{K_{AB}, A\}_K_{BS} \\
4. \textbf{B} \rightarrow \textbf{A} : \{N_B\}_K_{AB} \\
5. \textbf{A} \rightarrow \textbf{B} : \{N_B - 1\}_K_{AB}

\textbf{Known Key Attack}

If \textbf{C} gets an old session key \(K_{AB}\), then \textbf{C} can reuse the old key \(K_{AB}\) in messages 3 and 5 to impersonate \textbf{A}!

3. \textbf{C}(\textbf{A}) \rightarrow \textbf{B} : \{K_{old}, A\}_K_{BS} \\
4. \textbf{B} \rightarrow \textbf{C}(\textbf{A}) : \{N_B\}_K_{old} \\
5. \textbf{C}(\textbf{A}) \rightarrow \textbf{B} : \{N_B - 1\}_K_{old}
The Needham-Schroeder protocol

- Forerunner of Kerberos (used in e.g. Windows and Mac OS X).

1. \( A \rightarrow S : A, B, N_A \)
2. \( S \rightarrow A : \{N_A, B, K_{AB}, \{K_{AB}, A\}^K_{BS}\}^K_{AS} \)
3. \( A \rightarrow B : \{K_{AB}, A\}^K_{BS} \)
4. \( B \rightarrow A : \{N_B\}^K_{AB} \)
5. \( A \rightarrow B : \{N_B - 1\}^K_{AB} \)

**Problem:** Message 3 does not have any freshness (i.e., nonce) to guarantee that \( K_{AB} \) is a freshly generated key!

**Better solution:**

2. \( S \rightarrow A : \{K_{AB}\}^K_{AS}, MAC_{AS}(B, N_A, \{K_{AB}\}^K_{AS}), \{K_{AB}\}^K_{BS}, MAC_{BS}(A, N_A, \{K_{AB}\}^K_{BS}) \)

**Separation of concerns:** encryption for confidentiality, MAC for authentication.
Things to Remember

- Prove that textbook RSA is not IND-CPA secure.
- The discrete Log problem!
- Prove that El Gamal is not IND-CCA secure.
- Describe the Needham Schroeder protocol.
Attention Changes in the Schedule!

Friday 24/11/2017

08:00-10:00 Exercise Session

10:00-12:00 Lecture 8
References:

- “Cryptography and Network Security: Principles and practice” (Chapters 8.5, 10.2)
- “Introduction to Modern Cryptography”, Lindell and Katz (Chapter 8.3.1, 8.3.2, 9.0, 11.4.1)

Thank you for your attention!