Modern Cryptography January 14, 2020

Solutions to Homework 10

Name: Karen Klein, Guillermo Pascual Perez *Due: 23.59 CET, Dec 18, 2019*

To get credit for this homework it must be submitted no later than Wednesday, December 11th via TUWEL. If you have not registered for the tutorial (192.063 Tutorial on Introduction to Modern Cryptography 2019W) on TUWEL, please do so. If you are unable to register for the course on TUWEL for some reason, submit your homework via email to via email to `michael.walter@ist.ac.at`, please use "MC19 Homework 10" as subject.

1. Key-Exchange

   - **[10.1 in book, 2nd edition]** Let Π be a key-exchange protocol, and $(\mathsf{Enc}, \mathsf{Dec})$ be a private-key encryption scheme. Consider the following interactive protocol Π' for encrypting a message: first, the sender and receiver run Π to generate a shared key k. Next, the sender computes $c \leftarrow \mathsf{Enc}_k(m)$ and sends c to the other party, who decrypts and recovers m using k.
     - Formulate a definition of indistinguishable encryptions in the presence of an eavesdropper (cf. Definition in part 1 of the lecture) appropriate for this interactive setting.
     - Prove that if Π is secure in the presence of an eavesdropper and $(\mathsf{Enc}, \mathsf{Dec})$ has indistinguishable encryptions in the presence of an eavesdropper, then Π' satisfies your definition.

     Solution: Let Π be an arbitrary key-exchange protocol, $\Pi_{\mathsf{Enc}} = (\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ a symmetric-key encryption scheme, and Π' the interactive protocol constructed from Π and Π_{Enc} as above. Define the experiment $\mathsf{KE}-\mathsf{Enc}^{\mathsf{eav}}_{\mathsf{A},\Pi,\Pi_{\mathsf{Enc}}}(n)$ between an eavesdropping adversary A and the protocol Π' as follows:
     - A receives the security parameter 1^n in unary, chooses two messages m_0, m_1 of equal length and outputs (m_0, m_1).
     - Let (k, trans) be the output of an (interactive) run of Π between two honest parties on input 1^n.
     - Let $b \leftarrow \{0,1\}$ uniformly random and $c \leftarrow \mathsf{Enc}_k(m_b)$ and encryption of m_b under key k.
     - The adversary A receives the transcript trans and the ciphertext c, and outputs a bit b^*.
     - The output of the experiment is 1 if $b^* = b$, 0 else.

     Definition 1 *The interactive protocol Π' has* indistinguishable encryptions in the presence of an eavesdropper *if for every PPT adversary A there exists a negligible function negl such that*

     $$\mathsf{Pr}[\mathsf{KE} - \mathsf{Enc}^{\mathsf{eav}}_{\mathsf{A},\Pi,\Pi_{\mathsf{Enc}}}(n) = 1] \leq 1/2 + \mathsf{negl}(n).$$

If Π is secure in the presence of an eavesdropper and Π_{Enc} has indistinguishable encryptions in the presence of an eavesdropper, then Π' satisfies definition 1. To prove this, we proceed in two steps: First, we show that by the security of the key-exchange protocol Π the encryption scheme Π' is indistinguishable from a (non-functional) scheme $\bar{\Pi}'$ where instead of using the key k output by Π, the encryption mechanism samples a uniformly random fresh key $k' \leftarrow \{0,1\}^n$ and encrypts m_b under k' (note that in order to use the eavesdropping security of Π_{Enc} k' should be the output of $\mathsf{Gen}(1^n)$, which might not be uniform; we ignore this technicality and assume that Gen outputs a uniform key). Further, observe that $\bar{\Pi}'$ uses a completely independent key, thus, in the second step, we argue that by the eavesdropping security of Π_{Enc} an encryption of m_0 under $\bar{\Pi}'$ is indistinguishable from an encryption of m_1.

Let A be an arbitrary PPT adversary and write

$$\mathsf{Pr}[\mathsf{KE}-\mathsf{Enc}^{\mathsf{eav}}_{\mathsf{A},\Pi,\Pi_{\mathsf{Enc}}}(n)=1] = \mathsf{Pr}[\mathsf{A}(\mathsf{trans}, c \leftarrow \mathsf{Enc}_k(m_b)) = b].$$

First, we prove

$$\mathsf{Pr}[\mathsf{A}(\mathsf{trans}, c \leftarrow \mathsf{Enc}_k(m_b)) = b] \leq \mathsf{Pr}[\mathsf{A}(\mathsf{trans}, c \leftarrow \mathsf{Enc}_{k^*}(m_b)) = b] + \mathsf{negl}_1(n) \qquad (1)$$

for a *uniformly random* key k^* by constructing an adversary A_{KE} against the security of the key-exchange protocol Π as follows:

- On input the security parameter 1^n, the adversary A_{KE} receives a challenge pair (k^*, trans) where $(k, \mathsf{trans}) \leftarrow \Pi(1^n)$ and $k^* := k$ if $b_{\mathsf{KE}} = 0$ and $k^* \leftarrow \mathcal{K}$ uniformly at random and independently of trans if $b_{\mathsf{KE}} = 1$, where $b_{\mathsf{KE}} \leftarrow \{0,1\}$ uniformly random and $\mathcal{K}$ is the range of $\mathsf{Gen}(1^n)$.
- Then A_{KE} runs A on input 1^n and receives two messages m_0, m_1.
- A_{KE} samples $b \leftarrow \{0,1\}$ uniformly at random, computes $c \leftarrow \mathsf{Enc}_{k^*}(m_b)$ and sends (trans, c) to A.
- A_{KE} receives b^* from A and outputs $b^*_{\mathsf{KE}} = 0$ if $b^* = b$ and $b^*_{\mathsf{KE}} = 1$ else.

If $b_{\mathsf{KE}} = 0$, then A_{KE} correctly simulates $\mathsf{KE}-\mathsf{Enc}^{\mathsf{eav}}_{\mathsf{A},\Pi,\Pi_{\mathsf{Enc}}}(n)$ and wins the game $\mathsf{KE}^{\mathsf{eav}}_{\mathsf{A}_{\mathsf{KE}},\Pi}(n)$ if and only if A wins. On the other hand, if $b_{\mathsf{KE}} = 1$, then A_{KE} wins the game if and only if A does not win the modified game. Thus, we have

$$\begin{aligned}
\mathsf{Pr}[\mathsf{KE}^{\mathsf{eav}}_{\mathsf{A}_{\mathsf{KE}},\Pi}(n)=1] &= \frac{1}{2}\cdot(\mathsf{Pr}[\mathsf{KE}^{\mathsf{eav}}_{\mathsf{A}_{\mathsf{KE}},\Pi}(n)=1 \mid b_{\mathsf{KE}}=0] + \mathsf{Pr}[\mathsf{KE}^{\mathsf{eav}}_{\mathsf{A}_{\mathsf{KE}},\Pi}(n)=1 \mid b_{\mathsf{KE}}=1]) \\
&= \frac{1}{2}\cdot(\mathsf{Pr}[\mathsf{A}(\mathsf{trans}, c \leftarrow \mathsf{Enc}_k(m_b)) = b] + \mathsf{Pr}[\mathsf{A}(\mathsf{trans}, c \leftarrow \mathsf{Enc}_{k^*}(m_b)) = 1-b]) \\
&= \frac{1}{2}\cdot(\mathsf{Pr}[\mathsf{A}(\mathsf{trans}, c \leftarrow \mathsf{Enc}_k(m_b)) = b] + 1 - \mathsf{Pr}[\mathsf{A}(\mathsf{trans}, c \leftarrow \mathsf{Enc}_{k^*}(m_b)) = b])
\end{aligned}$$

Since Π is secure in the presence of an eavesdropper, we have $\mathsf{Pr}[\mathsf{KE}^{\mathsf{eav}}_{\mathsf{A}_{\mathsf{KE}},\Pi}(n) = 1] \leq 1/2 + \mathsf{negl}'_1(n)$ for some negligible function negl'_1, which implies (1) with $\mathsf{negl}_1 = 2 \cdot \mathsf{negl}'_1$.

We will now show that the alternative scheme $\bar{\Pi}'$ has indistinguishable encryptions, namely that the following equation holds:

$$\mathsf{Pr}[\mathsf{A}(\mathsf{trans}, c \leftarrow \mathsf{Enc}_{k^*}(m_b)) = b] \leq \frac{1}{2} + \mathsf{negl}_2(n) \qquad (2)$$

for some negligible function $\mathsf{negl}_2(n)$. Plugging this into (1) we get the desired result, as $\mathsf{negl}_1(n) + \mathsf{negl}_2(n)$ is also negligible.

We will construct an adversary $\mathsf{A}_{\mathsf{PrivK}}$ attacking Π_{Enc}:

- $\mathsf{A}_{\mathsf{PrivK}}$ is given the security parameter 1^n and runs A on input 1^n to obtain two messages (m_0, m_1). $\mathsf{A}_{\mathsf{PrivK}}$ then outputs these same messages.
- $\mathsf{A}_{\mathsf{PrivK}}$ is given a ciphertext $c := \mathsf{Enc}_{k^*}(m_b)) = b$, the encryption of one of the messages.
- $\mathsf{A}_{\mathsf{PrivK}}$ runs the key-exchange protocol Π itself, playing both roles, to generate a transcript trans and a key k (key which it will ignore, only the transcript is needed).
- $\mathsf{A}_{\mathsf{PrivK}}$ runs $\mathsf{A}(\mathsf{trans}, c)$, which returns a bit b. $\mathsf{A}_{\mathsf{PrivK}}$ outputs this same bit.

In the indistinguishable encryptions in the presence of an eavesdropper game $\mathsf{PrivK}^{\mathsf{eav}}_{\mathsf{A}_{\mathsf{PrivK}},\Pi_{\mathsf{Enc}}}(n)$ (recall from Lecture 3) , the key k^* used to generate c us chosen uniformly and independently of trans; A is given an encryption of m_i whenever $\mathsf{A}_{\mathsf{PrivK}}$ is; and $\mathsf{A}_{\mathsf{PrivK}}$ outputs as guess whatever is output by A. Hence,

$$\mathsf{Pr}[\mathsf{PrivK}^{\mathsf{eav}}_{\mathsf{A}_{\mathsf{PrivK}},\Pi_{\mathsf{Enc}}}(n) = 1] = \mathsf{Pr}[\mathsf{A}(\mathsf{trans}, c \leftarrow \mathsf{Enc}_{k^*}(m_b)) = b].$$

By assumption Π_{Enc} achieves indistinguishable encryptions in the presence of an eavesdropper, so the following holds for some negligible function negl_2:

$$\mathsf{Pr}[\mathsf{PrivK}^{\mathsf{eav}}_{\mathsf{A}_{\mathsf{PrivK}},\Pi_{\mathsf{Enc}}}(n) = 1] \leq \frac{1}{2} + \mathsf{negl}_2(n).$$

This implies equation (2) and completes the proof. □

2. Discrete Logarithms (Baby-Step/Giant-Step Algorithm)

- Prove that the Baby-Step/Giant-Step algorithm computes the discrete logarithm x given $g^x = h \bmod p$.

  Solution: Let's briefly recall how the algorithm works over the concrete case $G = \mathbb{Z}_p^*$, where p is a prime. Let $t = \lfloor \sqrt{p-1} \rfloor$ be the floor of the square root of the order of the group. The algorithm first computes a list of *Giant Steps* $(g^0, g^t, \ldots, g^{\lfloor p-1/t \rfloor \cdot t})$ and another of *Baby Steps* $(g \cdot g^1, \ \ldots, \ h \cdot g^t)$; and then looks for a match between the two. That is, some i, k such that:

  $$g \cdot g^i = g^{k \cdot t}$$

  It then returns $x := (k \cdot t - i) \mod p - 1$.

  First, to show that such a collision is always found, note that any $x \in \{0, \ldots, p-1\}$ can be written as $x_j = k \cdot t - i \mod p - 1$ for $k \in \{0, \ldots, \lfloor (p-1)/t \rfloor\}$ and $i \in \{1, \ldots, t\}$. Now all we need to show is that g^x is indeed equal to h. To do this, we can just multiply by g^{-i} on both sides of the equality above getting:

  $$h = g^{k \cdot t} \cdot g^{-i} = g^{k \cdot t - i} = g^{(k \cdot t - i) \mod p-1} = g^x$$

  where the penultimate equality follows from Corollary 8.15, from Lecture 9. □

- How are the fact that g is a generator of $\mathbb{Z}_p^*$ and the uniqueness of the output of the algorithm related? When is the output not unique?

Solution: For generality, let's not assume p is prime and write $q = \phi(p)$ for the order of the group (note that if p is a prime, then all elements of $\mathbb{Z}_p^*$ are generators - we need p composite to allow for non-trivial non-generators in the group). If g is a generator, we can write the elements in the group as $\{g^0, g^1, g^2, \ldots, g^{q-1}\}$. In particular, this means that $g^i \neq g^j$ for distinct $i, j \in \{0, \ldots, q-1\}$. Now, the output x of the algorithm is a number in $\{0, \ldots, q-1\}$, so for any other $i \in \{0, \ldots, q-1\}$ we ave $g^i \neq g^x$, meaning the output is unique.

If the output is not unique, there exist distinct x_1, x_2 such that $g^{x_1} = g^{x_2}$. However, this implies:

$$1 = g^{x_2 - x_1} \mod q$$

which, in particular, means that the order of g is at most $x_2 - x_1 < q$ (since $x_1 \neq x_2$), and therefore g is not a generator. Moreover, if g is not a generator, its order $o(g)$ has to be at most $q/2$ (since this has to divide the order of the group q), which implies that for any i, $g^i = g(i + o(g)) \mod q$ and $i \neq i + o(g)$. Thus the discrete logarithm problem has a non-unique output if and only if g is not a generator.

However, this does not automatically imply that the algorithm will necessarily have a non-unique output, i.e. that it will find all the x_j such that $g^{x_j} = h$. To see that this is the case, similarly as in the first part, note that any $x_j \in \{0, \ldots, q-1\}$ can be written as $x_j = k_j \cdot t - i_j \mod q - 1$ for $k_j \in \{0, \ldots, \lfloor q/t \rfloor\}$ and $i_j \in \{1, \ldots, t\}$. This implies $g^{k_j \cdot t} = h \cdot g^{i_j}$ and so the algorithm will output x_j. □

- Compute the discrete logarithm for $g^x = h \bmod p$ with $g = 3$, $h = 13$ and $p = 29$ using the Baby-Step/Giant-Step algorithm.

  Solution: The order of the group $\mathbb{Z}_p^*$ is $\phi(29) = 29 - 1 = 28$, since 29 is prime. Thus, define $t := \lfloor \sqrt{28} \rfloor = 5$. We first compute the list of *Giant Steps*:

  $$\begin{aligned}
  &(g^0, g^t, \ldots, g^{\lfloor p-1/t \rfloor \cdot t}) = \\
  &= (3^0 \mod 29,\ 3^5 \mod 29,\ 3^{10} \mod 29,\ 3^{15} \mod 29,\ 3^{20} \mod 29,\ 3^{25} mod 29) \\
  &= (1, 11, 5, 26, 25, 14)
  \end{aligned}$$

  then the list of *Baby Steps*:

  $$\begin{aligned}
  &(h \cdot g^1,\ \ldots,\ h \cdot g^t) = \\
  &= (13 \cdot 3^1 \mod 29,\ 13 \cdot 3^2 \mod 29,\ 13 \cdot 3^3 \mod 29,\ 13 \cdot 3^4 \mod 29,\ 13 \cdot 3^5 \mod 29) \\
  &= (10, 1, 3, 9, 27)
  \end{aligned}$$

  We can see that there is a match, namely $g^0 = 1 = h \cdot g^2$, i.e. for $k = 0$, $i = 2$ according to the notation from the first part. Thus $x = (0 - 2) \mod 28 = 26 \mod 28$ (note that we need to compute this modulo the order of the group, not p). Indeed $3^{26} = 13 \mod 29$. □

3. Group generators

   - Provide a proof for the following statement: If there exists a generator modulo n, then there are $\varphi(\varphi(n))$ many of them.

Solution: Assume $\mathbb{Z}_n^*$ is cyclic and g a generator. We show that $h = g^x \bmod n$ with $x \in \mathbb{Z}_{\phi(n)}$ is a generator of $\mathbb{Z}_n^*$ if and only if $x \in \mathbb{Z}_{\phi(n)}^*$:
Since g is a generator, we have $\langle h \rangle = \mathbb{Z}_n^*$ if and only if $g \in \langle h \rangle$, i.e., there exists $y \in \mathbb{Z}_{\phi(n)}$ such that $g = h^y = g^{xy} = g^{xy \bmod \phi(n)} \bmod n$. In the lecture we have seen that for any generator g of a group G of order m, the map $\mathbb{Z}_m \to G$, $x \mapsto g^x$ is an isomorphism. Using this result, we get that $h = g^x \bmod n$ is a generator of $\mathbb{Z}_n^*$ if and only if there exists $y \in \mathbb{Z}_{\phi(n)}$ such that $1 = xy \bmod \phi(n)$. The latter holds if and only if x is invertible modulo $\phi(n)$, which is equivalent to $x \in \mathbb{Z}_{\phi(n)}^*$, as claimed.

Thus, the set of generators of $\mathbb{Z}_n^*$ is $\{g^x \mid x \in \mathbb{Z}_{\phi(n)}^*\}$. Since $\mathbb{Z}_{\phi(n)}^*$ consists of all elements in $\mathbb{Z}_{\phi(n)}$ which are coprime to $\phi(n)$, we have $|\mathbb{Z}_{\phi(n)}^*| = \phi(\phi(n))$, hence (using the fact that $\mathbb{Z}_m \to G$, $x \mapsto g^x$ is an isomorphism) we can deduce that there are exactly $\phi(\phi(n))$ distinct generators of $\mathbb{Z}_n^*$. □