Modern Cryptography	January 7, 2019

Solutions to Homework 10

Lecturer: Daniel Slamanig, TA: Karen Klein

1. DL-related Problems

   - **[8.15 in book, 2nd edition]** Prove that hardness of the CDH problem relative to $\mathcal{G}$ implies hardness of the discrete-logarithm problem relative to $\mathcal{G}$, and that hardness of the DDH problem relative to $\mathcal{G}$ implies hardness of the CDH problem relative to $\mathcal{G}$.

     Solution: Let $(G, q, g) \leftarrow \mathcal{G}(1^n)$, where G is a cyclic group of order q with bit-size $||q|| = O(n)$ and g a generator of G.
     To prove that hardness of the CDH implies hardness of the discrete-logarithm problem, we show that any algorithm that solves the discrete-logarithm can be used to solve CDH. Let $\mathcal{A}$ be an arbitrary PPT algorithm for the discrete-logarithm problem with respect to $\mathcal{G}$, i.e., on input (G, q, g, g^x) it outputs $x' \in \mathbb{Z}_q$ and wins the game if $g^{x'} = g^x$, i.e., $x' = x$.[1] We construct an algorithm $\mathcal{A}'$ for CDH as follows: Given a CDH instance (G, q, g, g^x, g^y), $\mathcal{A}'$ queries $\mathcal{A}$ on (G, q, g, g^x) and receives $x' \in \mathbb{Z}_q$. Then $\mathcal{A}'$ computes $(g^y)^{x'}$. Clearly, $\mathcal{A}'$ succeeds if and only if $\mathcal{A}$ succeeds: $(g^y)^{x'} = \mathsf{DH}_g(g^x, g^y) \iff x' = x$. Hardness of CDH relative to $\mathcal{G}$ now implies that the success probability of *every* PPT algorithm – in particular that of $\mathcal{A}'$ – is bounded by some negligible function $\mathsf{negl}(n)$. Thus, we get

     $$\mathsf{Pr}[\mathsf{DLog}_{\mathcal{A},\mathcal{G}}(n) = 1] = \mathsf{Pr}[\mathcal{A}'(G, q, g, g^x, g^y) = g^{xy}] \leq \mathsf{negl}(n).$$

     To prove that CDH is harder than the DDH problem, let $\mathcal{A}$ be an arbitrary PPT algorithm for CDH with respect to $\mathcal{G}$, i.e., on input (G, q, g, g^x, g^y) it outputs $h \in G$ and wins the game if $h = \mathsf{DH}_g(g^x, g^y) = g^{xy}$. We construct an algorithm $\mathcal{A}'$ for DDH as follows: Given access to $\mathcal{A}$ and a DDH instance (G, q, g, g^x, g^y, h'), where either $h' = g^{xy}$ or $h' = g^z$ for a $z \in \mathbb{Z}_q$ chosen uniformly at random[2], the algorithm $\mathcal{A}'$ queries $\mathcal{A}$ on (G, q, g, g^x, g^y) and receives h. $\mathcal{A}'$ outputs 1 if $h' = h$ and 0 else. Thus,

     $$\mathsf{Pr}[\mathcal{A}'(G, q, g, g^x, g^y, g^{xy}) = 1] = \mathsf{Pr}[\mathcal{A}(G, q, g, g^x, g^y) = g^{xy}]$$

     On the other hand,

     $$\mathsf{Pr}[\mathcal{A}'(G, q, g, g^x, g^y, g^z) = 1] = \frac{1}{q}.$$

     Assuming that DDH is hard with respect to $\mathcal{G}$, we get

     $$|\mathsf{Pr}[\mathcal{A}'(G, q, g, g^x, g^y, g^z) = 1] - \mathsf{Pr}[\mathcal{A}'(G, q, g, g^x, g^y, g^{xy}) = 1]| \leq \mathsf{negl}(n).$$

     This implies

     $$\mathsf{Pr}[\mathcal{A}(G, q, g, g^x, g^y) = g^{xy}] \leq \mathsf{negl}(n) + \frac{1}{q},$$

     which is negligible since $||q|| = n$. This proves hardness of CDH. □

[1] Note, $g^{x'} = g^x$ implies $x' = x$, since for any generator g of G the map $(\mathbb{Z}_q, +) \to (G, \cdot)$, $x \mapsto g^x$ is an isomorphism.
[2] Note, if z is chosen uniformly at random from $\mathbb{Z}_q$ this implies that g^z is uniformly random in G.

- **[8.19 in book, 2nd edition]** Can the following problem be solved in polynomial time? Given a prime p, a value $x \in \mathbb{Z}_{p-1}^*$, and $y := [g^x \bmod p]$ (where g is a uniform value in $\mathbb{Z}_p^*$), find g, i.e., compute $y^{1/x} \bmod p$. If your answer is "yes", give a polynomial-time algorithm. If your answer is "no", show a reduction to one of the assumptions introduced in lecture 10.

  Solution: Yes, the above problem can be solved in polynomial time as follows: As shown in HW9, exercise 2c, the extended Euclidean algorithm can be used to compute the inverse $1/x$ of $x \in \mathbb{Z}_{p-1}^*$. Hence, we can compute $g = y^{1/x} \bmod p$. $\square$

- Let G be a cyclic group of prime order q and g a generator. The square Diffie-Hellman (sq-DH) problem is given (G, q, g, g^a) for $a \in \mathbb{Z}_q^*$ to compute g^{a^2}. Show that sq-DH $\iff$ CDH (Hint: $(x+y)^2$).

  Solution: First, we show that hardness of sq-DH implies hardness of CDH: Let $\mathcal{A}$ be an arbitrary PPT algorithm for CDH. We construct an algorithm $\mathcal{A}'$ for sq-DH as follows: Given an sq-DH instance (G, q, g, g^a), the algorithm $\mathcal{A}'$ chooses $r_1, r_2 \in \mathbb{Z}_q$ uniformly at random and queries $\mathcal{A}$ on $(G, q, g, (g^a)^{r_1}, (g^a)^{r_2})$. Note that $x = ar_1, y = ar_2$ are uniformly distributed in $\mathbb{Z}_q$, so $(G, q, g, g^{ar_1}, g^{ar_2})$ is a valid CDH instance. After receiving some value h from $\mathcal{A}$, the algorithm $\mathcal{A}'$ outputs $h' := h^{1/(r_1 r_2)}$ if $r_1 r_2$ is invertible in $\mathbb{Z}_q$, otherwise it outputs some uniformly random $h' \in G$. Clearly, if $\mathcal{A}$ succeeds and $r_1 r_2 \in \mathbb{Z}_q^*$, then $g^{a^2 r_1 r_2/(r_1 r_2)} = g^{a^2}$ is a solution to sq-DH. More precisely, if $r_1 r_2 \in \mathbb{Z}_q^*$, then $\mathcal{A}'$ succeeds if and only if $\mathcal{A}$ succeeds. Thus, we can compute the success probability of $\mathcal{A}'$ as follows:

$$\begin{aligned}\mathsf{Pr}[\mathcal{A}'(G,q,g,g^a) = g^{a^2}] = \ & \mathsf{Pr}[\mathcal{A}(G,q,g,g^{ar_1},g^{ar_2}) = g^{a^2 r_1 r_2}] \cdot \mathsf{Pr}[r_1 r_2 \in \mathbb{Z}_q^*] \\ & + \mathsf{Pr}[h' = g^{a^2}] \cdot \mathsf{Pr}[r_1 r_2 \notin \mathbb{Z}_q^*] \\ = \ & \mathsf{Pr}[\mathcal{A}(G,q,g,g^x,g^y) = g^{xy}] \cdot \tfrac{(q-1)^2}{q^2} + \tfrac{1}{q} \cdot (\tfrac{2}{q} - \tfrac{1}{q^2})\end{aligned}$$

  If the sq-DH assumption holds, i.e., sq-DH is hard with respect to the group generator $\mathcal{G}$, by definition there exists a negligible function negl such that

$$\mathsf{Pr}[\mathcal{A}'(G,q,g,g^a) = g^{a^2}] \le \mathsf{negl}(n)$$

  and by the above it follows

$$\mathsf{Pr}[\mathcal{A}(G,q,g,g^x,g^y) = g^{xy}] \le \frac{q^2}{(q-1)^2} \cdot (\mathsf{negl}(n) - \frac{1}{q} \cdot (\frac{2}{q} - \frac{1}{q^2})),$$

  which is negligible. Since $||q|| = n$ and $\mathcal{A}$ was an arbitrary algorithm for CDH, this implies hardness of CDH.

  To prove equivalence of sq-DH and CDH, we still have to prove that hardness of CDH implies hardness of sq-DH, i.e., that CDH can be solved using any algorithm $\mathcal{A}$ for sq-DH. To this aim, let $\mathcal{A}$ be an arbitrary PPT algorithm for sq-DH, (G, q, g, g^x, g^y) be an instance of CDH and note that $(x+y)^2 = x^2 + y^2 + 2xy$. We construct an algorithm $\mathcal{A}'$ for CDH as follows: If $g^x = 1$ or $g^y = 1$ then it must hold $x = 0$ or $y = 0$ and $\mathcal{A}'$ outputs the correct solution $1 = g^0 = g^{xy}$, i.e., $\mathcal{A}'$ succeeds with probability 1 in this case. If $g^x, g^y \neq 1$ but $g^x g^y = 1$ (i.e., $x + y = 0 \bmod q$), then $\mathcal{A}'$ queries $\mathcal{A}$ on (G, q, g, g^x).

After receiving h from $\mathcal{A}$, the algorithm $\mathcal{A}'$ outputs h^{-1}. Note, that if $\mathcal{A}$ succeeds, then $h = g^{x^2}$ and $\mathcal{A}'$ succeeds since $y = -x \bmod q$. Hence, $\mathcal{A}'$ has the same success probability as $\mathcal{A}$ in this case. Finally, if $g^x, g^y, g^x g^y \neq 1$, then $\mathcal{A}'$ chooses $r \in \mathbb{Z}_q^*$ uniformly at random and queries $\mathcal{A}$ three times to obtain $h_1 = \mathcal{A}(G, q, g, g^x)$, $h_2 = \mathcal{A}(G, q, g, g^y)$ and $h_3 = \mathcal{A}(G, q, g, (g^x g^y)^r)$. Then $\mathcal{A}'$ computes $1/2 \bmod q$ and $1/(2r^2) \bmod q$ (note that both 2 and r are invertible modulo q) and outputs $h' = h_3^{1/(2r^2)}(h_1 h_2)^{-1/2}$. If $\mathcal{A}$ succeeds on all three instances, then $h_1 = g^{x^2}$, $h_2 = g^{y^2}$ and $h_3 = g^{(r(x+y))^2}$, so it follows

$$h' = h_3^{1/(2r^2)}(h_1 h_2)^{-1/2} = (g^{r^2(x+y)^2})^{1/(2r^2)}(g^{x^2} g^{y^2})^{-1/2} = g^{((x+y)^2 - x^2 - y^2)/2} = g^{xy}.$$

Since $\mathcal{A}$ is queried on three independent looking properly distributed sq-DH instances, we can lower-bound the success probability of $\mathcal{A}'$ as follows:

$$\mathsf{Pr}[\mathcal{A}'(G, q, g, g^x, g^y) = g^{xy}] \geq (\mathsf{Pr}[\mathcal{A}(G, q, g, g^x) = g^{x^2}])^3.$$

If CDH is hard, it hold $\mathsf{Pr}[\mathcal{A}'(G, q, g, g^x, g^y) = g^{xy}] \leq \mathsf{negl}(n)$. Thus, we get

$$\mathsf{Pr}[\mathcal{A}(G, q, g, g^x) = g^{x^2}] \leq (\mathsf{negl}(n))^{1/3}$$

which is negligible. Thus, we proved hardness of sq-DH.

$\square$

2. Key-Exchange

- Let p be a prime and g be a generator of $\mathbb{Z}_p^*$. Argue why we are not able to prove $\widehat{\mathsf{KE}}^{\mathsf{eav}}_{\mathcal{A},\Pi}$ security of the Diffie Hellman key-exchange protocol in this setting. Construct a polynomial-time distinguisher (Hint: quadratic residues).

  Solution: The clue for breaking security of $\widehat{\mathsf{KE}}^{\mathsf{eav}}_{\mathcal{A},\Pi}$ over $\mathbb{Z}_p^*$ is to consider the subgroup $QR_p \leq \mathbb{Z}_p^*$ of quadratic residues mod p.
  Recall, $y \in \mathbb{Z}_p^*$ is called a *quadratic residue modulo* p if there exists an $x \in \mathbb{Z}_p^*$ such that $x^2 = y \bmod p$; such an x is then called a *square root* of y. It can be shown that each quadratic residue modulo p has precisely two distinct square roots, namely x and its additive inverse $-x$ in $\mathbb{Z}_p$ (which also lies in $\mathbb{Z}_p^*$). If we denote the set of quadratic residues as QR_p, it is easy to see that QR_p forms a subgroup and $QR_p = \{g^{2i} \mid i \in \{0, \ldots, \frac{p-1}{2}\}\}$. In particular, $|QR_p| = \frac{p-1}{2} = \frac{|\mathbb{Z}_p^*|}{2}$. Furthermore, there is an efficient algorithm to compute quadratic residuosity as

  $$\mathcal{J}_p(x) := x^{\frac{p-1}{2}} = \begin{cases} +1 & \text{if } x \in QR_p \\ -1 & \text{if } x \notin QR_p. \end{cases}$$

  $\mathcal{J}_p(x)$ is called the Jacobi (or Legendre) symbol.
  In the $\widehat{\mathsf{KE}}^{\mathsf{eav}}_{\mathcal{A},\Pi}(b)$ security game, an adversary $\mathcal{A}$ knows the public parameters $(\mathbb{Z}_p^*, p-1, g) \leftarrow \mathcal{G}(1^n)$ as well as a tuple $(k^*, trans)$ with $trans = (g^x, g^y)$ for some uniformly random secret $x, y \in \mathbb{Z}_{p-1}^*$. If $b = 0$ then $k^* = \mathsf{DH}_g(g^x, g^y) = g^{xy}$, otherwise k^* is a uniformly random element in $\mathbb{Z}_p^*$. The adversary $\mathcal{A}$ wins the game if he can guess the bit b with non-negligible probability.

Now, consider the case $b = 1$ where $k^* \leftarrow \mathbb{Z}_p^*$ is uniformly random. Then $k^* \in QR_p$ with probability $\frac{1}{2}$. On the other hand, if $b = 0$, then $k^* = g^{xy}$ where $x, y \leftarrow \mathbb{Z}_{p-1}$ are chosen independently and uniformly at random. It holds $k^* \in QR_p$ if and only if $xy \bmod p-1$ is even, i.e., x or y is even, which happens with probability $1 - \mathsf{Pr}[x \,\text{odd} \,\wedge\, y \,\text{odd}] = \frac{3}{4}$. We use this observation to construct an efficient adversary $\mathcal{A}$ against $\widehat{\mathsf{KE}}^{\mathsf{eav}}_{\mathcal{A},\Pi}(b)$:
On input $(\mathbb{Z}_p^*, p-1, g, k^*, trans)$, $\mathcal{A}$ computes $\mathcal{J}_p(k^*)$. If $\mathcal{J}_p(k^*) = +1$, he outputs $b' = 0$, if $\mathcal{J}_p(k^*) = -1$ he outputs $b' = 1$. $\mathcal{A}$ wins the game with probability

$$\begin{aligned} \mathsf{Pr}[b' = b] &= \mathsf{Pr}[b' = b | b = 0] \cdot \mathsf{Pr}[b = 0] + \mathsf{Pr}[b' = b | b = 1] \cdot \mathsf{Pr}[b = 1] \\ &= \tfrac{1}{2}(\mathsf{Pr}[b' = 0 | b = 0] + \mathsf{Pr}[b' = 1 | b = 1]) \\ &= \tfrac{1}{2}(\tfrac{3}{4} + \tfrac{1}{2}) = \tfrac{5}{8} > \tfrac{1}{2} + \mathsf{negl}(n). \end{aligned}$$

Note, the adversary $\mathcal{A}$ above does not even use the information in the transcript $trans = (g^x, g^y)$ to break the scheme. One can improve the attack as follows. It holds

$$g^{xy} \in QR_p \iff (x = 0 \bmod 2 \vee y = 0 \bmod 2) \iff g^x \in QR_p \vee g^y \in QR_p.$$

We construct an adversary $\mathcal{A}'$ as follows. $\mathcal{A}'$ computes $\mathcal{J}_p(g^x), \mathcal{J}_p(g^y), \mathcal{J}_p(k^*)$ to decide whether g^x, g^y, k^* are quadratic residues. Then he defines bits b_x, b_y, b^* as

$$b_x = \begin{cases} 0 & \text{if } x \notin QR_p \\ 1 & \text{if } x \in QR_p. \end{cases} \quad b_y = \begin{cases} 0 & \text{if } y \notin QR_p \\ 1 & \text{if } y \in QR_p. \end{cases} \quad b^* = \begin{cases} 0 & \text{if } k^* \notin QR_p \\ 1 & \text{if } k^* \in QR_p. \end{cases}$$

Finally, $\mathcal{A}'$ outputs $b' = 0$ if $b^* = b_x \vee b_y$ and $b' = 1$ else. Now, consider the case $b = 0$, i.e., $k^* = g^{xy}$. Then $b^* = b_x \vee b_y$ and $\mathcal{A}'$ will output $b' = 0 = b$ with probability 1. In the case $b = 1$, on the other hand, k^* will be uniformly random. In this case, the probability of k^* being a quadratic residue or nonresidue is $\frac{1}{2}$, respectively. This means that the bit b^* is uniformly random and independent of b_x, b_y. Hence, with probability $\frac{1}{2}$ it will hold $b^* = b_x \vee b_y$. It follows that $\mathcal{A}'$ wins the game $\widehat{\mathsf{KE}}^{\mathsf{eav}}_{\mathcal{A},\Pi}(b)$ with probability

$$\begin{aligned} \mathsf{Pr}[b' = b] &= \mathsf{Pr}[b' = 0 | b = 0] \cdot \mathsf{Pr}[b = 0] + \mathsf{Pr}[b' = 1 | b = 1] \cdot \mathsf{Pr}[b = 1] \\ &= \tfrac{1}{2}(\mathsf{Pr}[b^* = b_x \vee b_y | b = 0] + \mathsf{Pr}[b^* = b_x \vee b_y | b = 1]) \\ &= \tfrac{1}{2}(1 + \tfrac{1}{2}) = \tfrac{3}{4}. \end{aligned}$$

$\square$