EE 720: An Introduction to Number Theory and Cryptography (Spring 2019)

Lecture 21 — April 4, 2019

Instructor: Saravanan Vijayakumaran *Scribe: Saravanan Vijayakumaran*

1 Lecture Plan

- Discrete Logarithm
- Diffie-Hellman Protocol
- Diffie-Hellman Problems

2 Discrete Logarithm Assumption

- Let $\mathcal{G}$ be a polynomial-time group generation algorithm. On input 1^n, it outputs a description of a cyclic group G, its order q (with $\|q\| = n$), and a generator $g \in G$.
  - The description of the group specifies how the group elements are represented as bit-strings. We assume that each group element is represented by a unique bit-string.
  - We require that there is an efficient algorithm for the group operation in G.
  - We require that there is an efficient algorithm for testing whether a given bit-string represents an element of G.
- Efficient algorithms for group operation imply efficient algorithms for exponentiation in G and for sampling a uniform element from G.
- **Definition:** If G is a cyclic group of order q with generator g, then we can express G as $\{g^0, g^1, g^2, \ldots, g^{q-1}\}$ if we denote the identity e of G as $e = g^q = g^0$. For $h \in G$ the unique $x \in \mathbb{Z}_q$ which satisfies $g^x = h$ is called the discrete logarithm of h with respect to g.
- Discrete logarithms obey many of the same rules as standard logarithms.
  - $\log_g e = 0$ where e is the identity of the group G.
  - $\log_g(h_1 h_2) = \left[\log_g h_1 + \log_g h_2\right] \bmod q$.
- The discrete logarithm problem is believed to be hard in cyclic groups of prime order. A subgroup of $\mathbb{Z}_p^*$ having prime order q is a good choice. Another possibility is elliptic curve groups of prime order.
- **The discrete logarithm experiment** $\texttt{DLog}_{\mathcal{A},\mathcal{G}}(n)$:
  1. Run $\mathcal{G}(1^n)$ to obtain (G, q, g) where G is a cyclic group of order q (with $\|q\| = n$), and a generator $g \in G$.
  2. Choose a uniform $h \in G$.

3. $\mathcal{A}$ is given G, q, g, h and it outputs $x \in \mathbb{Z}_q$.
4. Experiment output is 1 if $g^x = h$ and 0 otherwise.

- **Definition:** We say that **the discrete logarithm problem is hard relative to** $\mathcal{G}$ if for all PPT adversaries $\mathcal{A}$ there is a negligible function $\texttt{negl}$ such that

$$\Pr\left[\texttt{DLog}_{\mathcal{A},\mathcal{G}}(n) = 1\right] \leq \texttt{negl}(n).$$

3 Diffie-Hellman Protocol

- How do parties which use private-key cryptographic schemes share a secret key in the first place?
- One solution is to have a trusted party act as the key distribution center. But this center is a single point of failure. The DH protocol presents an alternative.
- **The Diffie-Hellman key-exchange protocol:**

  1. Alice runs a group generation algorithm to get (G, q, g) where G is a cyclic group of order q with generator g.
  2. Alice chooses a uniform $x \in \mathbb{Z}_q$ and computes $h_A = g^x$.
  3. Alice sends (G, q, g, h_A) to Bob.
  4. Bob chooses a uniform $y \in \mathbb{Z}_q$ and computes $h_B = g^y$. He sends h_B to Alice. He also computes $k_B = h_A^y$.
  5. Alice computes $k_A = h_B^x$.

  By construction, $k_A = k_B$.
- **How can one prove that this protocol is secure?** We first need to define what is meant by security of a key-exchange protocol.
- Suppose Alice and Bob run a probabilistic protocol Π generate a shared secret key k. The key k should be indistinguishable from a uniformly random key for a PPT adversary with access to the protocol transcript.
- **The key-exchange experiment** $\texttt{KE}^{\texttt{eav}}_{\mathcal{A},\Pi}(n)$:

  1. Two parties holding 1^n execute protocol Π. This results in a transcript $\texttt{trans}$ and a key $k \in \{0,1\}^n$ output by both of the parties.
  2. A uniform bit $b \in \{0,1\}$ is chosen. If $b = 0$ set $\hat{k} := k$, and if $b = 1$ then choose $\hat{k}$ uniformly from $\{0,1\}^n$.
  3. Adversary $\mathcal{A}$ is given $\texttt{trans}$ and $\hat{k}$, and outputs a bit b'.
  4. The adversary succeeds if $b' = b$ and the output of the experiment is 1. Otherwise, the output is 0.
- **Definition:** A key-exchange protocol Π is **secure in the presence of an eavesdropper** if for all PPT adversaries $\mathcal{A}$ there is a negligible function $\texttt{negl}$ such that

$$\Pr\left[\texttt{KE}^{\texttt{eav}}_{\mathcal{A},\Pi}(n) = 1\right] \leq \frac{1}{2} + \texttt{negl}(n).$$

4 Diffie-Hellman Problems

The Diffie-Hellman problems are related to the problem of computing discrete logarithms, but they are not known to be equivalent to it.

4.1 Computational Diffie-Hellman (CDH) Problem

- Let G be a cyclic group with generator $g \in G$. Given elements $h_1, h_2 \in G$, define

$$\mathrm{DH}_g(h_1, h_2) = g^{\log_g h_1 \cdot \log_g h_2}.$$

  That is, if $h_1 = g^{x_1}$ and $h_2 = g^{x_2}$ then

$$\mathrm{DH}_g(h_1, h_2) = g^{x_1 \cdot x_2} = h_1^{x_2} = h_2^{x_1}.$$

- If the discrete logarithms are easy to compute in a group, then the CDH problem is easy. But it is not clear if the hardness of the discrete logarithm problem implies hardness of the CDH problem.

- **The CDH experiment** $\mathtt{CDH}_{\mathcal{A},\mathcal{G}}(n)$:

  1. Run $\mathcal{G}(1^n)$ to obtain (G, q, g) where G is a cyclic group of order q (with $\|q\| = n$), and a generator $g \in G$.
  2. Choose a uniform $x_1, x_2 \in \mathbb{Z}_q$ and compute $h_1 = g^{x_1}, h_2 = g^{x_2}$.
  3. $\mathcal{A}$ is given G, q, g, h_1, h_2 and it outputs $h \in \mathbb{Z}_q$.
  4. Experiment output is 1 if $h = g^{x_1 \cdot x_2}$ and 0 otherwise.

- **Definition:** We say that **the CDH problem is hard relative to** $\mathcal{G}$ if for all PPT adversaries $\mathcal{A}$ there is a negligible function $\mathtt{negl}$ such that

$$\Pr\left[\mathtt{CDH}_{\mathcal{A},\mathcal{G}}(n) = 1\right] \leq \mathtt{negl}(n).$$

4.2 Decisional Diffie-Hellman (CDH) Problem

- The DDH problem is to distinguish $\mathrm{DH}_g(h_1, h_2)$ from a uniform group element when h_1, h_2 are uniformly chosen.

- **Definition:** We say that **the DDH problem is hard relative to** $\mathcal{G}$ if for all PPT adversaries $\mathcal{A}$ there is a negligible function $\mathtt{negl}$ such that

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

  where in each case the probabilities are taken over the experiment in which $\mathcal{G}$ outputs (G, q, g), and then uniform $x, y, z \in \mathbb{Z}_q$ are chosen.

5 Security of DH Key Exchange

- In the DH key exchange protocol, the generated key is not a bit string from $\{0,1\}^n$. It is a group element. So we modify the experiment $\mathtt{KE}^{\mathtt{eav}}_{\mathcal{A},\Pi}(n)$ to $\widehat{\mathtt{KE}}^{\mathtt{eav}}_{\mathcal{A},\Pi}(n)$ where the adversary in the latter experiment is asked to distinguish between the DH protocol generated key and a uniform group element.
- **Theorem**: If the DDH problem is hard relative to $\mathcal{G}$, then the Diffie-Hellman key-exchange protocol Π is secure in the presence of an eavesdropper (with respect to the modified experiment $\widehat{\mathtt{KE}}^{\mathtt{eav}}_{\mathcal{A},\Pi}(n)$).

6 References and Additional Reading

- Sections 8.3.2 from Katz/Lindell
- Sections 10.1, 10.2, 10.3 from Katz/Lindell