Joyce Yeo 5 Feb 2020

# Diffie-Hellman Key Exchange

What is the Diffie-Hellman Key Exchange protocol? What are the underlying mathematical assumptions?

## Key Exchange Protocols

In most cryptographic communication, both communicating parties (Alice and Bob) need to know a shared secret. This way, any messages encrypted using this secret key cannot be decrypted by malicious listeners.

However, before the communication can be encrypted using the key, Alice and Bob need to decide on the secret key that they will be using. During this decision process, the communication channel is still insecure and susceptible to eavesdropping. For instance, Alice cannot simply tell Bob that the key will be 81729312 as eavesdroppers will also know the key. Key Exchange Protocols are designed to allow Alice and Bob to agree on a shared secret key even with an eavesdropper present. One class of key exchange protocols is the Diffie-Hellman Key Exchange.

## Diffie-Hellman Mathematics

Having precise assumptions in cryptography is important. These assumptions become the only uncertainty in the cryptosystem. Mathematically intractable problems form the basis of most modern cryptosystems.

Diffie-Hellman mathematics is based on operations in cyclic groups. Consider a generator g and a modulus n, the group $<g>$ is defined as $<g, g^2, g^3, \ldots, >$ modulus n.

Example: g = 3, n = 23, then $<g> = <3, 3^2 \bmod 23, 3^3 \bmod 23 \ldots > = < 3, 9, 4, 12, 13, 16, 2, 6, 18, 8, 1 >$

Note that the group $<g>$ is finite (order = 11 elements) because the operations are done over modulus n. The maximum number of elements in the group is n.

## Diffie-Hellman Problems

To help relate group mathematics with the applications of the Diffie-Hellman Key Exchange Protocol, we first formulate a few problems.

1. Discrete Logarithm (DL) Problem: Given x, find r such that $x = g^r$.

   Example: Let g = 3, in modulus 23. Given x = 8, find r such that $8 = 3^r \bmod 23$.
   Since we have already generated the whole group earlier, we get r = 10. However, it is generally difficult to compute r without generating the entire group.

2. Computational Diffie-Hellman (CDH) Problem: Given $g^a$, $g^b$, find $g^{ab}$.

   Example: Let g = 3, in modulus 23. Given $3^a = 4$, $3^b = 16$, find $3^{ab}$.
   We solve for a = 3 and b = 6. Then, $3^{ab} = 3^{18} = 3^{11 + 7} = 3^{11} (3^7) = 1 (3^7) = 2$.
   We solve the DL problem to solve the CDH problem. Hence, DL is easy => CDH is easy.

3. Decisional Diffie-Hellman (DDH) Problem: Given $g^a$, $g^b$, distinguish between a random $g^c$ and $g^{ab}$.

   Example: Let g = 3, in modulus 23. Given $3^a = 4$, $3^b = 12$. Now, given 8 and 3, decide which is $g^{ab}$ and which is $g^c$.
   We solve for a = 3 and b = 4, $3^{ab} = 3^{12} = 3^{11} (3) = 3$. Hence $3 = 3^{ab}$ and $8 = g^c$.
   We solve the CDH problem to solve the DDH problem. Hence, CDH is easy => DDH is easy.

By comparing the three problems, we have that DL easy => CDH easy => DDH easy. If we can efficiently calculate the discrete logarithm in a cyclic group, then we can also solve the CDH and DDH problems in that group. Considering the contrapositive, we have that DDH hard => CDH hard => DL hard.

A group where the DDH problem is hard would be ideal for all cryptographic schemes relying on either the DDH, CDH or DL problems. CDH and DL would also be difficult in the group. However, we may not need such a powerful group satisfying all the mathematical assumptions.

Putting Everything Together

Precisely defining the Diffie-Hellman problems gives us a means to specify the mathematical assumptions made in designing the key exchange protocol.

In the Diffie-Hellman Key Exchange, the following events take place:

1. Alice and Bob establish **g** and a modulus **n**, which is public. This defines the cyclic group.
2. Alice and Bob each have a personal secret, **a** and **b**.
3. Alice sends Bob $\mathbf{A = g^a}$ and Bob calculates $\mathbf{K = g^{ab} = A^b}$. Bob now has the key K.
4. Bob sends Alice $\mathbf{B = g^b}$ and Alice calculates $\mathbf{K = g^{ab} = B^a}$. Alice now has the key K.

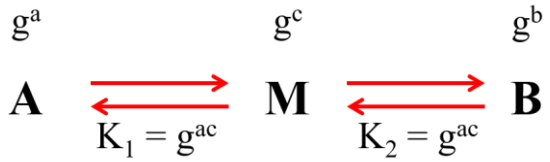
Since the communication is still not encrypted, any eavesdropper can retrieve everything that Alice and Bob sent to each other: g, n, $A = g^a$ and $B = g^b$. Given these values, the eavesdropper must find the key $K = g^{ab}$. This description is exactly the **CDH** problem. Minimally, we must use a mathematical group where the CDH problem is hard. We do not need a DDH-hard group, just a CDH-hard group.

A classic example of a group that is CDH-hard but not DDH-hard are the $Z_p^*$ groups. Let p be a prime number, then $Z_p^* = \{ 1, 2, 3, 4, 5, \dots , p - 1 \}$.

Diffie-Hellman problems form the cryptographic foundation for many other applications, not just the key exchange protocol. However, if a scheme is designed based on the DDH problem, then the users must ensure that the DDH problem is hard in the group used.

Attacking the Diffie-Hellman Key Exchange

However, in modern communication channels, eavesdropping is not the only problem. A CDH-hard group can only protect the key exchange from an eavesdropper. Attackers have other capabilities, such as **interception**.




A middleman can create a new secret c and calculate $g^c$.

- When Alice sends $g^a$, M intercepts it and sends Alice $g^c$. Alice communicates with M using $K_1 = g^{ac}$.
- When Bob sends $g^b$, M intercepts it and sends Bob $g^c$. Bob communicates with M using $K_2 = g^{bc}$.

Alice will think that she is communicating with Bob using $K_1$. Bob will think he is talking to Alice using $K_2$.

The problem is that the Diffie-Hellman Key Exchange protocol was never designed for **authentication**. The identities of both parties are never established. Even without the presence of M, Alice cannot confidently ascertain that the person she is talking to is indeed Bob.

Authentication protocols can be used to establish that $g^b$ was indeed generated by the person Bob. However, that was not the consideration in designing the Diffie-Hellman Key Exchange.