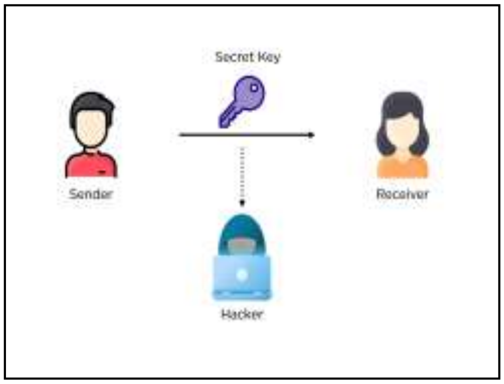
Diffie-Hellman Key Exchange

A significant portion of today's internet is encrypted using secret encryption keys. Apart from bolstering security, encrypting communication channels maintains both the confidentiality and integrity of data being transmitted. The exchange of these secret keys had always been arduous, irrespective of scope, until the development of the Diffie-Hellman Key Exchange algorithm for public use. It helps solve the problem of exchanging symmetric encryption keys without compromising data integrity.

To understand why the Diffie-Hellman key exchange algorithm is a global standard, let us first see why it was created.

Why Is the Diffie-Hellman Key Exchange Algorithm Necessary?

Symmetric encryption has always been a reliable method of cryptography for the exchange of private information. A glaring flaw has always been the difficulty in sharing the requisite secret key with the receiver of the message. It can intercept any key transmitted over an insecure channel by hackers, who can then use the same key to decrypt the encrypted ciphertexts.



The Diffie Hellman algorithm solves this problem using one-way functions that enable only the sender and receiver to decrypt the message using a secret key. Now, you will learn more about how the one-way functions help in the transmission of keys.

How Do One Way Functions Work?

One-way functions follow a type of algorithm where you can calculate an output for every input. However, it is theoretically impossible to derive the respective input from a random result. To better understand how this helps in the Diffie-Hellman exchange, you can use color theory to realize its effectiveness.

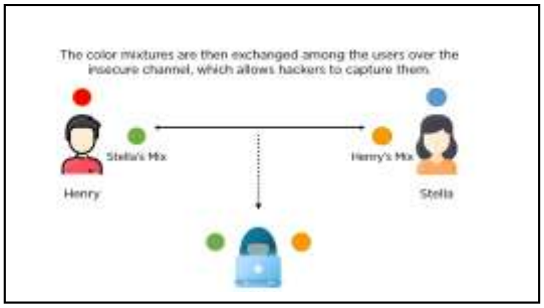
Step 1: Let your two users choose a publicly accepted color they both agree to. They must also decide on a private color which is to be kept as a secret.

Secret Colour
Secret Colour
Publicly accepted colour
Publicly accepted colour

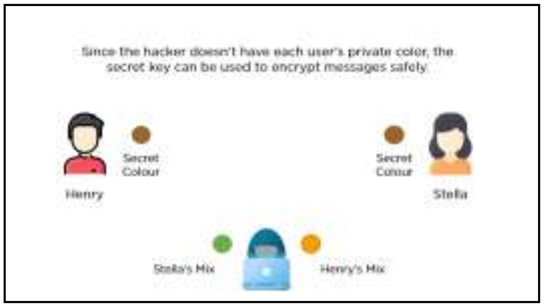
Step 2: The private and public colors are mixed on each side to form a newly acquired color mixture.

Mixture #1
Mixture #2

Step 3: They then exchange the mixture among the users over an insecure communication channel, even though it may be open for hackers to intercept.


The color mixtures are then exchanged among the users over the insecure channel, which allows hackers to capture them.
Stella's Mix
Henry's Mix
Henry
Stella

Step 4: The private colors are then mixed with the received mixture to finally acquire the actual secret color (key).


Since the hacker doesn't have each user's private color, the secret key can be used to encrypt messages safely
Secret Colour
Secret Colour
Henry
Stella
Stella's Mix
Henry's Mix

As you can notice, despite the critical exchange taking place over a channel with hackers present, the malicious users received mixed colors, but not the secret key. Both users can now encrypt their messages using the private key generated without fear of hackers reading their conversations.

Now, since you know the basic procedure behind one-way functions, you can learn more about the origin and application of the Diffie-Hellman key exchange algorithm.

What Is the Diffie-Hellman Key Exchange?

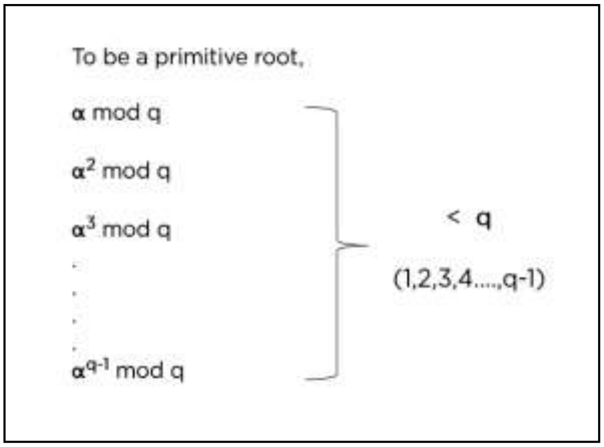
The Diffie-Hellman algorithm is a method for securely exchanging cryptographic keys over insecure channels without compromising the security and integrity of data transmission. It was developed and published in 1976 by Martin Hellman and Whitefield Diffie. Until you received the asymmetric encryption algorithms that never relied on any category of key exchange, symmetric encryption was the only way to communicate securely. A secure method to exchange the private keys for this brand of cryptography was much needed.

There are three distinct steps to exchange keys, right from its generation up to its transmission which has been discussed in the next section.

Steps in Key Exchange

The steps needed for the Diffie-Hellman key exchange are as follows:

Step 1: You choose a prime number **q** and select a primitive root of **q** as α.
To be a primitive root,
it must satisfy the following criteria:



Step 2: You assume the private key for our sender

as **Xa** where $Xa < q$.

The public key can be calculated as

$Ya = \alpha^{xa} \bmod q$.

So, the key pair for your sender becomes {**Xa**, **Ya**}.

Assume the private key for the receiver to be

Xb where $Xb < q$.

The public key for the receiver is calculated as

$Yb = \alpha^{xb} \bmod q$.

For the receiver, the key pair becomes {**Xb**, **Yb**}.

Step 3: To generate the final secret key,

you use three parameters.

For the sender, you need the private key (**Xa**),

the receiver's public key (**Yb**), and the original **q**.

The formula to calculate the key is $K = (Yb)^{Xa} \bmod q$.

For the receiver, you need the private key (**Ya**),

sender's public key (**Xb**), and the original **q**.

The formula to calculate the secret key is $K = (Ya)^{Xb} \bmod q$.

If both the values of **K** generated are equal,

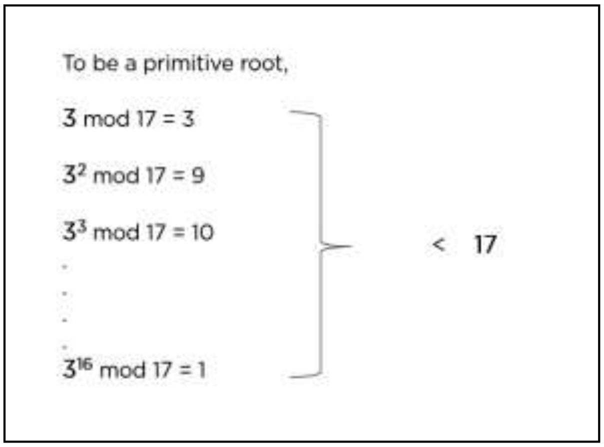
the Diffie-Hellman key exchange algorithm is complete.

Now, apply the above algorithm to real-world values to understand how the process works.

Practical Implementation

The steps needed to test the Diffie-Hellman key exchange are as follows:

Step 1: You choose the prime number **q** to be **17**. For its primitive root, you select the value of **α** to be **3,** since it satisfies the following criteria:



To be a primitive root,

$$\left.\begin{array}{l} 3 \bmod 17 = 3 \\ 3^2 \bmod 17 = 9 \\ 3^3 \bmod 17 = 10 \\ \cdot \\ \cdot \\ \cdot \\ 3^{16} \bmod 17 = 1 \end{array}\right\} < 17$$

Step 2:

You assume the sender's private key **Xa** to be **15**.

The public key can be calculated as

Ya = **3^15** mod **17** = **6**.

The key pair for our sender becomes {**15**, **6**}.

For the receiver's end, you assume private key **Xb** to be **13**.

The public key is calculated as

Yb = **3^13** mod **17** = **12**.

The key pair for the receiver is now {**13**, **12**}.

Step 3: The secret key generated on the sender's side is
K = **12^15** mod **17** = **10**.

The secret key generated from the receiver's side is
K = **6^13** mod **17** = **10**.

Since both the keys generated are equal, the Diffie-Hellman exchange is valid, and the secret key **'10'** can encrypt messages between the users.

Now that you know how the key exchange works, let us learn about its applications in the real world.

Applications of Diffie-Hellman Algorithm

- ➢ Public Key Infrastructure: The public-key infrastructure (PKI) is a set of tools and rules to enforce public key cryptography with multiple entities. It also governs the issuance of digital certificates over the internet to maintain data confidentiality. With the Diffie-Hellman algorithm as the base, the PKI system was created to enable the exchange of public keys with anyone who requests for it and has the appropriate permissions.

- ➢ SSL/TLS Handshake: Internet browsers are authenticated with website servers using an SSL/TLS certificate and many keys. This is possible only because of the key exchange algorithm, which enables the secure exchange of cryptographic entities over all channels.

- ➢ Secure Shell Access (SSH): SSH is a cryptographic protocol used to access system terminals from a third-party appliance or application. The Diffie-Hellman algorithm assists in exchanging the keys between both systems before enabling remote access.