Network Security

Assignment#3

MIM Attack on Diffie-Hellman Key Exchange

Name: Zain Ul Ebad

Roll: P166024

Submitted to: Sir Waqas Ali

Date: 15/May/2020

Introduction

Diffie–Hellman (DH) key exchange is a method of securely exchanging cryptographic keys over a public channel and was one of the first public-key protocols

Where is the Diffie-Hellman key exchange used?

The main purpose of the Diffie-Hellman key exchange is to securely develop shared secrets that can be used to derive keys. These keys can then be used with symmetric-key algorithms to transmit information in a protected manner.

Symmetric algorithms tend to be used to encrypt the bulk of the data because they are more efficient than public key algorithms. Technically, the Diffie-Hellman key exchange can be used to establish public and private keys.

The ElGamal algorithm, which was used heavily in PGP, is based on the Diffie-Hellman key exchange, so any protocol that uses it is effectively implementing a kind of Diffie-Hellman.

How does the Diffie-Hellman key exchange work?

The Diffie-Hellman key exchange is complex and it can be difficult to understand. To make things a bit easier to understand, we will start by explaining the Diffie-Hellman key exchange with an analogy. Once you have a big-picture idea of how it works, we'll move on to a more technical description of the underlying processes.

The best analogy for the Diffie-Hellman scheme is to think of two people mixing paint. Let's use the cryptography standard, and say that their names are Alice and Bob. They both agree on a random colour to start with. Let's say that they send each other a message and decide on yellow as their common colour.

Let's say that Alice chooses red, while Bob chooses a slightly-greenish blue.

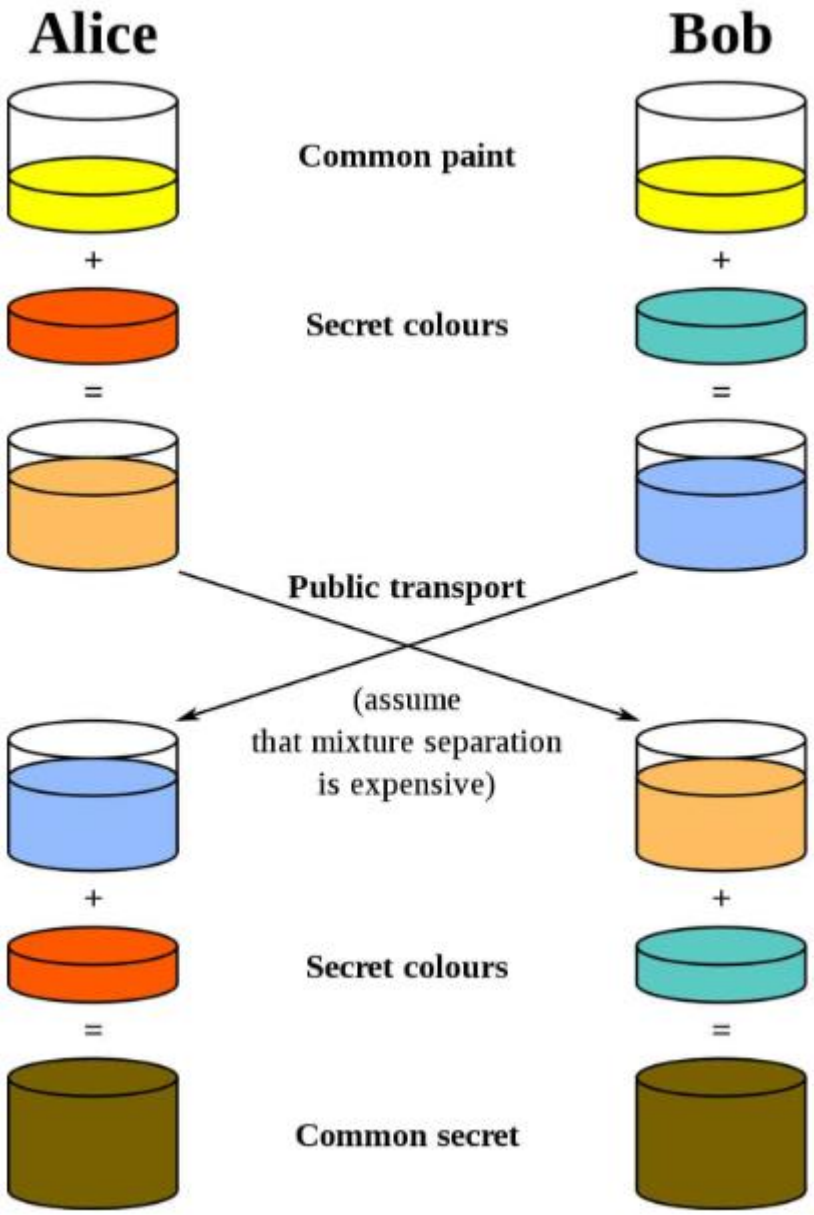
The next step is for both Alice and Bob to mix their secret colour (red for Alice, greenish-blue for Bob) with the yellow that they mutually agreed upon. According to the diagram, Alice ends up with an orangish mix, while Bob's result is a deeper blue.

Once they have finished the mixing, they send the result to the other party. Alice receives the deeper blue, while Bob is sent the orange-coloured paint.

Once they have received the mixed result from their partner, they then add their secret colour to it. Alice takes the deeper blue and adds her secret red paint, while Bob adds his secret greenish-blue to the orange mix he just received.

The result? They both come out with the same color, which in this case is a disgusting brown. It may not be the kind of color that you would want to paint your living room with, but it is a shared color nonetheless. This shared color is referred to as the common secret.

The critical part of the Diffie-Hellman key exchange is that both parties end up with the same result, without ever needing to send the entirety of the common secret across the communication channel. Choosing a common color, their own secret colours, exchanging the mix and then adding their own color once more, gives both parties a way to arrive at the same common secret without ever having to send across the whole thing.



The technical details of the Diffie-Hellman key exchange

It follows a similar premise as the analogy shown above, but instead of mixing and sending colors, the Diffie-Hellman scheme actually makes calculations based on exceptionally-large prime numbers, then sends them across.

To ensure security, it is recommended that the prime (p) is at least 2048 bits long.

In the most basic form of the Diffie-Hellman key exchange, Alice and Bob begin by mutually deciding upon two numbers to start with, as opposed to the single common paint in the example above. These are the modulus (p) and the base (g).

In practical use, the modulus (p) is a very large prime number, while the base (g) is relatively small to simplify calculations. The base (g) is derived from a cyclic group (G) that is normally generated well before the other steps take place.

Example

For our example, let's say that the modulus (p) is 17, while the base (g) is 4.

Once they have mutually decided on these numbers, Alice settles on a secret number (a) for herself, while Bob chooses his own secret number (b). Let's say that they choose:

a = 3

b = 6

Alice then performs the following calculation to give her the number that she will send to Bob:

A = ga mod p

In the above calculation, mod signifies a modulo operation. These are essentially calculations to figure out the remainder after dividing the left side by the right. As an example:

15 mod 4 = 3

If you understand how modulo operations work, you can do them yourself in the following calculations, otherwise you can use an online calculator.

So let's put our numbers into the formula:

A = 43 mod 17

A = 64 mod 17

A = 13

When we do the same for Bob, we get:

B = 46 mod 17

B = 4096 mod 17

B = 16

Alice then sends her result (A) to Bob, while Bob sends his figure (B) to Alice. Alice then calculates the shared secret (s) using the number she received from Bob (B) and her secret number (a), using the following formula:

s = Ba mod p

s = 163 mod 17

s = 4,096 mod 17

s = 16

Bob then performs what is essentially the same calculation, but with the number that Alice sent him (A), as well as his own secret number (b):

s = Ab mod p

s = 136 mod 17

s = 4,826,809 mod 17

s = 16

As you can see, both parties ended up with the same result for s, 16.

Establishing a shared key between multiple parties

The Diffie-Hellman key exchange can also be used to set up a shared key with a greater number of participants. It works in the same manner, except further rounds of the calculations are needed for each party to add in their secret number and end up with the same shared secret.

Just like in the two-party version of the Diffie-Hellman key exchange, some parts of the information are sent across insecure channels, but not enough for an attacker to be able to compute the shared secret.

Authentication & the Diffie-Hellman key exchange

In the real world, the Diffie-Hellman key exchange is rarely used by itself. The main reason behind this is that it provides no authentication, which leaves users vulnerable to man-in-the-middle attacks.

These attacks can take place when the Diffie-Hellman key exchange is implemented by itself, because it has no means of verifying whether the other party in a connection is really who they say they are. Without any form of authentication, users may actually be connecting with attackers when they think they are communicating with a trusted party.

For this reason, the Diffie-Hellman key exchange is generally implemented alongside some means of authentication. This often involves using digital certificates and a public-key algorithm, such as RSA, to verify the identity of each party.

Authentication & the Diffie-Hellman key exchange

In the real world, the Diffie-Hellman key exchange is rarely used by itself. The main reason behind this is that it provides no authentication, which leaves users vulnerable to man-in-the-middle attacks.

These attacks can take place when the Diffie-Hellman key exchange is implemented by itself, because it has no means of verifying whether the other party in a connection is really who they say they are. Without any form of authentication, users may actually be connecting with attackers when they think they are communicating with a trusted party.

For this reason, the Diffie-Hellman key exchange is generally implemented alongside some means of authentication. This often involves using digital certificates and a public-key algorithm, such as RSA, to verify the identity of each party.