Diffie-Hellman Key Exchange

Welcome back. So far, when looking at symmetric encryption and decryption. We've ignored one key point. How do both parties share a key in a secure manner? Of course, there's always meeting physically. But often times, when you're communicating over the internet or something like that, it's not going to be practical to physically get together in order to share the key. So how do we go about solving this issue? What we need is a key exchange algorithm.

In this video, we're going to be looking at the Diffie-Hellman key exchange algorithm. This is a way of generating a shared key over and insecure channel, which allows future secure communications. It's a very elegant solution to a challenging problem. And you'll see that the use of mathematics allows us to share information over an insecure channel in such a way that an interceptor or attacker would not be able to generate the private shared key. Diffie-Hellman is still in use and modern cryptography. In particular, adaptions of Diffie-Hellman such as triple Diffie-Hellman are used in applications such as WhatsApp.

The details that we'll go over today refer specifically to what was presented in the original paper. But of course, the amendments follow the same principle. So for our purposes, it's sufficient to look at that. So what does this look like? Let's have a look at the Diffie-Hellman key exchange at high level. Often, this is likened to combining different paint colours. This is a helpful analogy in terms of figuring out, fundamentally, the key component of how this works.

To do this, I will demonstrate on the diagram shown here using shapes instead of colours. Alice is going to pick a private colour. Bob will do the same. At no point does Bob or Alice send over these private colours through the public communication channel. We also have to agree a public colour. Now, Eve can access the public colour at any point, as can Alice and Bob. What we do then is we take the public colour and combine it with our private colours as Bob and Alice have selected. So Alice can combine these two colours, which results in a colour which cannot be taken apart to figure out the component parts. Even though Eve has access to the public colour, it would be incredibly difficult--

what we refer to as computationally infeasible--

to try and extract what Alice's private colour was.

Similarly, Bob can do the same sort of thing. He can take his private colour and the public colour and combine the two in such a way that he can communicate it. But Eve is not going to be able to do anything with that. She's not going to be able to extract the private value within there. Now, what we can do is we can see Alice has access to the combination of the public colour and Bob's private colour. So she can add her own private colour to that mix to end up with our secret key our shared secret key. And Bob can apply his

private colour to the combination of Alice's private colour applied to the public colour. So you can see in a very elegant way, we've managed to come up with the same colour on both sides.

We need part of a secret in order to be able to generate that final key. But Eve is never going to be able to access either one of those from the information that she has available to her. So you can see how the combination of different colours of paints is an effective way of showing how the communication takes place. We'll now turn to looking at this in a little bit more detail at the mathematics behind this. Let's revisit this now looking at the numbers in a little bit more depth. In terms of the publicly agreed information, we have two values. We have a large prime key, and we also have a number g.

These two values are mathematically related in order to make this work. But we won't go into the full details of how we calculate those values. Of course, if you are particularly interested in the detail here, there are plenty of resources online which discuss this. Alice and Bob have to define their own private integers. We'll refer to these as a private number a and a private number b. And those are never communicated across the public channel.

Now, a has to be less than p as does b, and they're both positive integers. We then get into the calculations. So Alice is able to calculate g to the power of a mod p. And we will label the result of this calculation as capital A. Similarly, Bob completes an equivalent calculation. He instead calculates g to the power of b mod p. So capital A and capital B can be communicated to the other party over this insecure channel. Knowing p and knowing g, it is still computationally infeasible for Eve to be able to figure out the private integer b for Bob or a for Alice. Alice can then calculate the shard key by taking B, which has been provided by Bob, raising that to the power of her private integer a, and calculating the result mod p.

Similarly, Bob can calculate the secret key by taking capital A--

so remember, this is g to the power a mod p--

raising that to the power of his private integer and calculating the result mod p. Now, you might think that these two values are going to be different, but they aren't. The reason for this is, if you look at the make up of what capital B is it's g to the power of b. Now, the way that the mathematics works, if we're taking an exponential value and then raising it again to another value, that's the equivalent of multiplying those two values.

So what we actually get there is g to the power of b times a mod p. And on Bob's side, similarly, we are saying that g to the power of a is taken to the power of the private integer that Bob has, mod p. And we end up with g to the power of a times b mod p. Now, the way that multiplication of integers works, b times a is equivalent to a times b. So we're actually ending up with the same key value. And even though Eve has access to capital B and capital A, p, and g, she's unable to derive the final secret key and she doesn't have access to either a or b here.

It's a very elegant solution and is so important in modern cryptography. That's us now, had a look at how Diffie-Hellman manages to agree a key over an insecure channel. I think you'll agree it's a particularly elegant solution. I hope you've enjoyed video, and I'll see you next time.