Block cipher principles

Each block cipher will have the principles, most symmetric block cipher in today's world is based on Feister cipher structure. DES structure basically, we need to have this kind of structure because, we need to decrypt ciphertext to recover messages efficiently. If your algorithm has the Feister structure then you can recover your plaintext back from cipher text, in other words it guarantees that, there could be a decryption and block ciphers looks like an extremely large substitution, in other words as I said, 64 bit input will map every 64 bit output and so on. We would need a table of 2 to the power 64 entries, why I am saying 64 bit entries is we need 64 bit block size for DES algorithm and if that is the case and if you are encrypting on the 64 bit block and your ciphertext is 64 bit mapping then that means you are mapping this 64 bit to another 64 bit ,so that's why it is called extremely large substitutions. Instead of building this kind of tables we are going to have smaller building blocks and the algorithms, and we are going to use the idea of product cipher, remember when we were looking at the details of substitution cipher and transposition cipher, we said that modern ciphers are based on product ciphers. In other words modern ciphers are going to use substitution and transposition in the cipher basically.

Claude Shannon and Substitution permutation cipher

In 1949 Claude Shannon introduced the idea of substitution and permutation ciphers in other words, all the modern ciphers are going to use this, we are going to call it S boxes. And for permutation we are going to call it P boxes. And substitution and permutation will provide confusion and diffusion of messages. And this I going to form basis for modern ciphers. In other words every modern block ciphers will have substitution and permutation, when you will see the AES algorithm ,you will see substitution box and permutation box. When you see the algorithm in details you are going to see the s box and the p box again. When you will be the blowfish again you will see the S box and permutation part of the algorithm. In other words ,all the algorithms that we are going to see will have the substitution part and the permutation part. That's why we call it product ciphers.

Confusion and Diffusion.

Ciphers needs to completely obscure the statistical properties of original message, imagine the substitution cipher we have to hide lot of things and frequencies and eventually when we realize that when we do substitution and transposition or transposition and substitution, things are going be much clear and same things is true over here. And what we have to do is we have to remove any security relationship between substitution and the transposition, if we can do that, then we have a good secure cipher. And one time pad does this and one time pad is the one of the major algorithms, lets see the details later on, as we discussed vernam cipher, and what we are doing is we have the message and plaintext and message comes like this, and we have the key and we are going to XOR the key like this and one time paths, and this is vernam cipher and it is one the key algorithm if the key is quite random. Moe practically, Shannon suggested combining elements to obtain diffusion and confusion. Diffusion dissipates statistical structure of plaintext over bulk of ciphertext. In other words we are going to remove the statistical relationship between ciphertext and plaintext. So ciphertext and plaintext relationship will be removed. In confusion it makes relationship between ciphertext and key as complex

as possible, in other words, it will be high if you have physical relationship between ciphertext and key ,recover the key and the algorithm will be broken. We have to hide two things, statistical relation ship between ciphertext and plaintext or statistical relationship between ciphertext and the secret key, to encrypt the messages. So diffusion and confusion will provide some kind of security when it comes to ciphertext and plaintext and ciphertext and key is concerned. So that is called confusion and diffusion basically.

Feister cipher structure

DES will be having a structure. And that structure is called feister cipher structure and you are going to compile that thing with the structure, you can give a different name and still you can give the name feister structure. Horst feister developed this cipher, this guy also invented DES algorithm , he invented based on the concept of invertible product cipher. Invertible means when you encrypt the data you can basically decrypt the data or recover the data back to the cipher basically. What we are going to do in this cipher is this we are going to have the message and we are going to split this message into blocks and each block will be split into two halves, right and left half and we are going to split into multiple rounds, and we are going to perform substitution on the left half of the data and abased on the round function we are going to perform ,we are going to perform right half of the data and the subkeys, remember that subkey function, of the round and the key is going to be the input to that round. In other words key and round will be the output in the right half of the data. And we are going to perform permutation. In this what we are doing is , I have a plaintext block and there are two W bitsand I am going to split these two blocks in to two halves, W bit right and the W bit left and the left half will be XORed with the, output of the round function and I am going to call it F. pay attention input to F is the subkey ,which is generated from the original key, and the right half of the data and that's going to be XORed . in other words, left half, is processed and this left half becomes right half and right half, not processed automatically becomes left half for the next round. And in the next round I am not doing anything's, it's the same thing, I am going to swap and I am going to swap it N times, and the actual result will be swapped and the actual result is one block cipher text. And this is feistel cipher. And if you have this kind of cipher and if you design this kind of cipher, we can guarantee that this cipher can be decrypted from which you can get plain text back from ciphertext and how you are going to do, I am going to explain.

Feister cipher design principles.

First of all we are going to check these principles, block size, because block size will be very important for us, if you are improving the block size then you are improving the security ,on the other side encryption is a cross process, by doing this you are slowing the encryption process. And speed will be very important especially in the today's world and the key size. Larger the key size, better the security, but slows the cipher, increasing number of rounds increases the security but slows the cipher, subkey generation, greater complexity, improves security but slows the cipher, round function, greater complexity but slows down the cipher. When you increase the key size then you are slowing the algorithm, but if you are getting complex security and complex round function ,but you are improving the security and slowing algorithm. But speed as I said is very important, for example if I have to login to

my bank account, I enter my password and I am waiting for 15 mins. This is not going to be expectable ,in other words you should find the balance. (…..) and speed will be actual challenge for us in the cryptography, you cannot say you have 1 bit security and you have more security and so on., if you are going to do that, that may take your password but it will take 1 week to encrypt or decrypt your password. And that s why we have to find a balance and that's why in today's world. Software that we ar using is very important for any crypto algorithms that we are using. Lot of people are doing research and spending lot o time just to find fast software, and encryption and decryption of crypto algorithms, these principles are very important, if you are designing a algorithm then these principles are very important. Block size and security, security and key size, block size and key size. Security and speed is concerned.

Feister Cipher Decryption

We did the encryption, we got the cipher ,how we are going to get the plaintext back, we said that feistel structure will guarantee us from getting plaintext from cipher text. This is the input to the cipher text and this is split into two halves, in this algorithm we have around 16 rounds and after 16 rounds as you can see you are getting the output of it. And I am using subkey 1 for the first round, subkey 2 for the second round and like that subkey 16 for the 16th round. And if you look at the decryption part again we are going to split into two halves left hand and right hand half, this is the first round for the decryption and the decryption key is the last subkey that I used in encryption will be the first subkey that I used in encryption. And in the second round, key 15 will go like this and in the last round, it will be K1 and other function F ,will not be changing at all. We are not changing anything in the algorithm we are jus changing subkey that we used in reverse order, and decryption. That is the whole idea behind feister, if you want to get plaintext back from the cipher text all you have to do is take subkey and you use for the encryption in the reverse order. And the result will be your plaintext ,you see how easy that is, if you have such numbers then you can do this algorithm and any algorithm basically and DES is one of them, DES Is going to use 64 blocks, and 16 rounds, and key size will be 64 or 56 depending on key. But the round key size are going to be 48 bits. Are we clear with this decryption process, the important thing is you are going to decrypt the process, from the cipher text by using subkeys in reverse order. Key 16 is going to be the first round key and key 15 is going to be the second round key and the last round key, for the decryption is going to be K1. Next we are going to talk in detail about DES algorithm.