**ASSESSMENT-3**

**CRYPTOGRAPHY ANALYSIS AND IMPLEMENTATION**

**ETHICAL HACKING EXTERNSHIP**

**NAME: KIRAN MANIKANDHAN**

**REG NO: 20BCE1786**

**VIT CHENNAI**

**Symmetric Encryption Algorithm: Blowfish**

Blowfish is a symmetric encryption algorithm designed by Bruce Schneier in 1993 as an alternative to [DES Encryption Technique](https://www.geeksforgeeks.org/data-encryption-standard-des-set-1/). It is significantly faster than DES and provides a good encryption rate with no effective [cryptanalysis technique](https://www.geeksforgeeks.org/introduction-to-crypto-terminologies/) found to date. It is one of the first, secure block cyphers not subject to any patents and hence freely available for anyone to use.

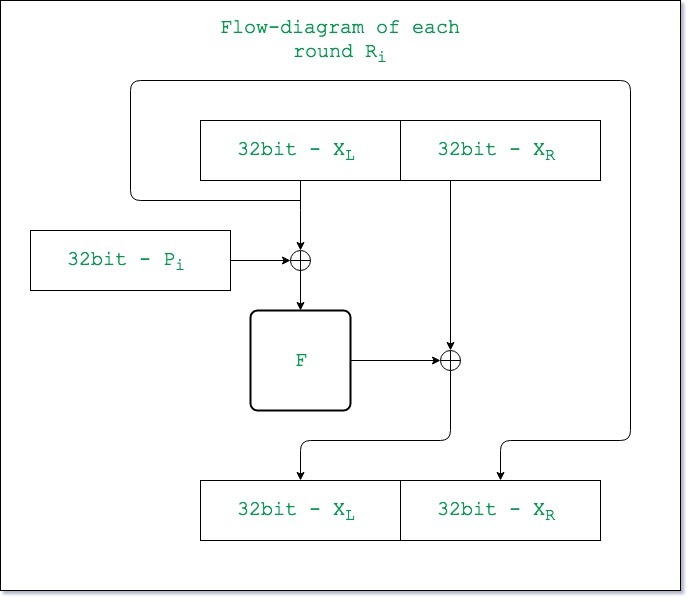
1. Block Size: 64-bits
2. Key Size: 32-bits to 448-bits variable size.
3. Number of subkeys: 18 [P-array]
4. Number of rounds: 16
5. Number of substitution boxes: 4 [each having 512 entries of 32-bits each].

**Working:**

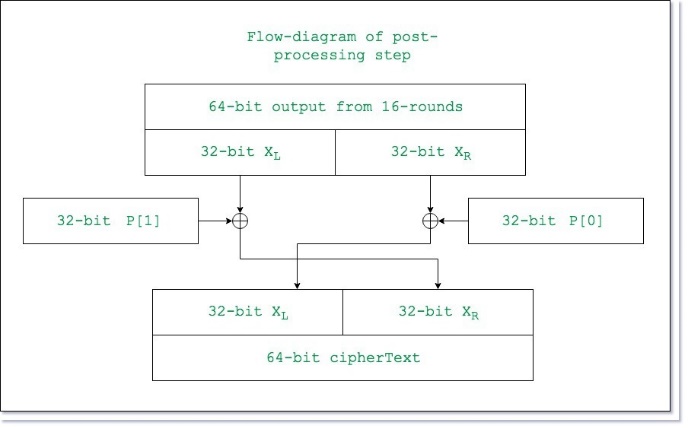
1. Generation of subkeys: 18 subkeys are stored in P-array in 32-bit format. Each subkey is initialized with digits of pi in hex representation and these keys are used for both encryption and decryption process. Each subkey is w.r.t input key as:

P[0] = P[0] xor 1st 32-bits of input key and so on.

1. Initialize Substitution boxes: 4 S-boxes are needed for both encryption and decryption process. It is initialized after P-array.
2. Encryption: It consists of two parts:
3. Rounds: Consists of 16 rounds and each round input is taken as plaintext from previous round.



1. Post-processing: The output is processed by dividing 64-bit outputs into 32-bits as follows:



**Advantages:**

* Fast, efficient, and secure.
* Since the key size is variable, it makes it more secure.
* Easy to implement and is compatible with many programming languages and platforms.

**Vulnerabilities:**

* Speed is affected when changing keys.
* Key schedule takes a long time.
* Small 64-bit block makes algorithm vulnerable to birthday attacks (type of brute-force attacks).

**Applications:**

* Password storage.
* Secure file transfer.
* Virtual Private Networks (VPNs).
* Email encryption.
* Password hashing and management.
* Packet encryption and random bit generation.

**Asymmetric Encryption Algorithm: RSA**

RSA (Rivest-Shamir-Adleman) algorithm is an asymmetric cryptography algorithm. Asymmetric means that it works on two different keys i.e., Public Key and Private Key. As the name describes that the Public Key is given to everyone and the Private key is kept private.

**Working:**

1. Generating public keys:

* Select two prime numbers (P and Q).
* First part of public key is n = P\*Q.
* Next, e an integer is taken which is not a factor of n and our public key is made of n and e.

1. Generating private keys:

* Calculate O(n) = (P-1) \*(Q-1).
* Now private key d is calculated as d = (k\*O(n) + 1).

**Advantages:**

* It is very secure and widely used for secure data transmission.
* RSA is used for public-key cryptography.
* RSA is used for secure key exchange.
* Speed of RSA is quite quick and effective.

**Vulnerabilities:**

* Slow processing speed especially when dealing with large amounts of data.
* Large key size requires more computational power and storage space.
* Limited use in some applications like decryption of large data or apps which require constant encryption.
* Vulnerable to side-channel attacks.
* Complex algorithm.
* Secure management of private key is necessary.
* Vulnerable to quantum computing.

**Applications:**

* Message encryption.
* Digital Signature.
* Secure communication protocols (like SSH and HTTPS).
* Encrypting email messages.
* VPNs and software protection.

**Hash Function: SHA-256**

SHA 256 is a part of the SHA 2 family of algorithms, where SHA stands for Secure Hash Algorithm. Published in 2001, it was a joint effort between the NSA and NIST to introduce a successor to the SHA 1 family, which was slowly losing strength against [brute force attacks.](https://www.simplilearn.com/tutorials/cryptography-tutorial/brute-force-attack) The significance of the 256 in the name stands for the final hash digest value, i.e. irrespective of the size of plaintext/cleartext, the hash value will always be 256 bits.

**Working:**

1. Padding bits: Some extra bits are added to message to length of exactly 64 bits.
2. Padding length: 64 bits of data is added to make final plaintext length a multiple of 512.
3. Initializing buffers: Eight buffers from a to h are initialized to be used in rounds. Also, 64 different keys are stored from K[0] to K[63].
4. Compression functions: Entire message is broken into blocks of length of 512 bits each. Each block is put through 64 rounds of operation.
5. Final output of each block (i.e., 512 bit hash) serves as the input for the next block and repeats until it reaches the last 512-bit block.

**Advantages:**

* Collision resistant: No two input values produce the same hash.
* Preimage resistance: Input cannot be recreated given a hash value.
* Deterministic: Output of hash is always the same if the input is same.
* Large Output: 256-bit output adds up to 2256 possibilities making it impossible to apply brute force solution to crack the hash.
* Avalanche effect: Small change in input results in a dramatic change in the output hash.

**Disadvantages:**

* SHA-256 is slower than its predecessors.
* Some software needs to be updated to support SHA-256 encryption.

**Applications:**

* Consensus mechanism in blockchain.
* Digital signatures verification.
* Password Hashing.
* SSL Handshake in browsing.
* Integrity checks.