290201098

270201072

1. Block ciphers are chopped into fixed size of bit blocks before they are encrypted with the decided key, whereas stream ciphers encrypt data bit by bit. For example, since AES is a block ciphering method, bits are grouped, here it was blocks of 16 bytes. Most of the time the last block will not be exactly 16 bytes, then the message is padded. This feature of block ciphers makes them also slower than the stream ciphers because whole block must be collected before the encryption may start. Stream ciphers, on the other hand, can cipher bits as they come. Block encryptions provide high diffusion too, and not only confusion as stream ciphers do, for change of one bit affects many bits in the ciphertext, but this however may also result in higher error rates because a small error in the plaintext bit may affect blocks of information. Depending on one’s need, both can be chosen. For example, when one owns classified information and time is a small issue to have the necessary security, block ciphers can be chosen. When data must be sent fast, stream ciphers can be chosen. For example, while having a video conference, it would be a bad idea to accumulate the data chunks, encrypt them and send them. Then the receiver will receive the data after a long time which is not desirable.
2. We don’t need to keep the IV as a secret, but it would be ideal not to use the same IV with the same key while encrypting for it may leak information about the beginning of the plaintext if there is a repetition in the plaintext, an example may be an email which starts with certain lines such as TO: or FROM: (This issue is also existent in ECB mode.) IV should be also randomly generated and not predictable. Here in this homework, random bytes were generated for IV as well and they were kept as is in the .json file. This helps to create a completely different ciphertext even if the plaintext is the same.
3. While CBC mode is more secure than the ECB, these mods don’t provide the users authentication, but only confidentiality. Padding oracle attacks on the CBC may be used to guess the plaintext since the paddings are not authenticated in CBC mode by trial and error (*Sullivan, 2021*.) In CBC mode, every plaintext block is XOR’ed with a known cipher block by the attacker if they are listening to the communication, therefore data integrity is a problem and injections by the attacker should be detected. If one is confident that the ciphertext may not be altered on the way, CBC mode may still be viable (for static data for example.) To overcome the problem of authentication, AEAD modes were introduced to AES, which do not only provide confidentiality, but also integrity (*Block cipher mode of Operation, 2022*.) Being one of the AEAD modes, Galois / Counter Mode provides users authentication and integrity along with secrecy. Recently, it is preferable to use.

**References**

Sullivan, N. (2021, August 19). *Padding oracles and the decline of CBC-Mode Cipher Suites*. The Cloudflare Blog. Retrieved November 15, 2022, from <https://blog.cloudflare.com/padding-oracles-and-the-decline-of-cbc-mode-ciphersuites/>

Wikimedia Foundation. (2022, November 3). *Block cipher mode of Operation*. Wikipedia. Retrieved November 15, 2022, from <https://en.wikipedia.org/wiki/Block_cipher_mode_of_operation>