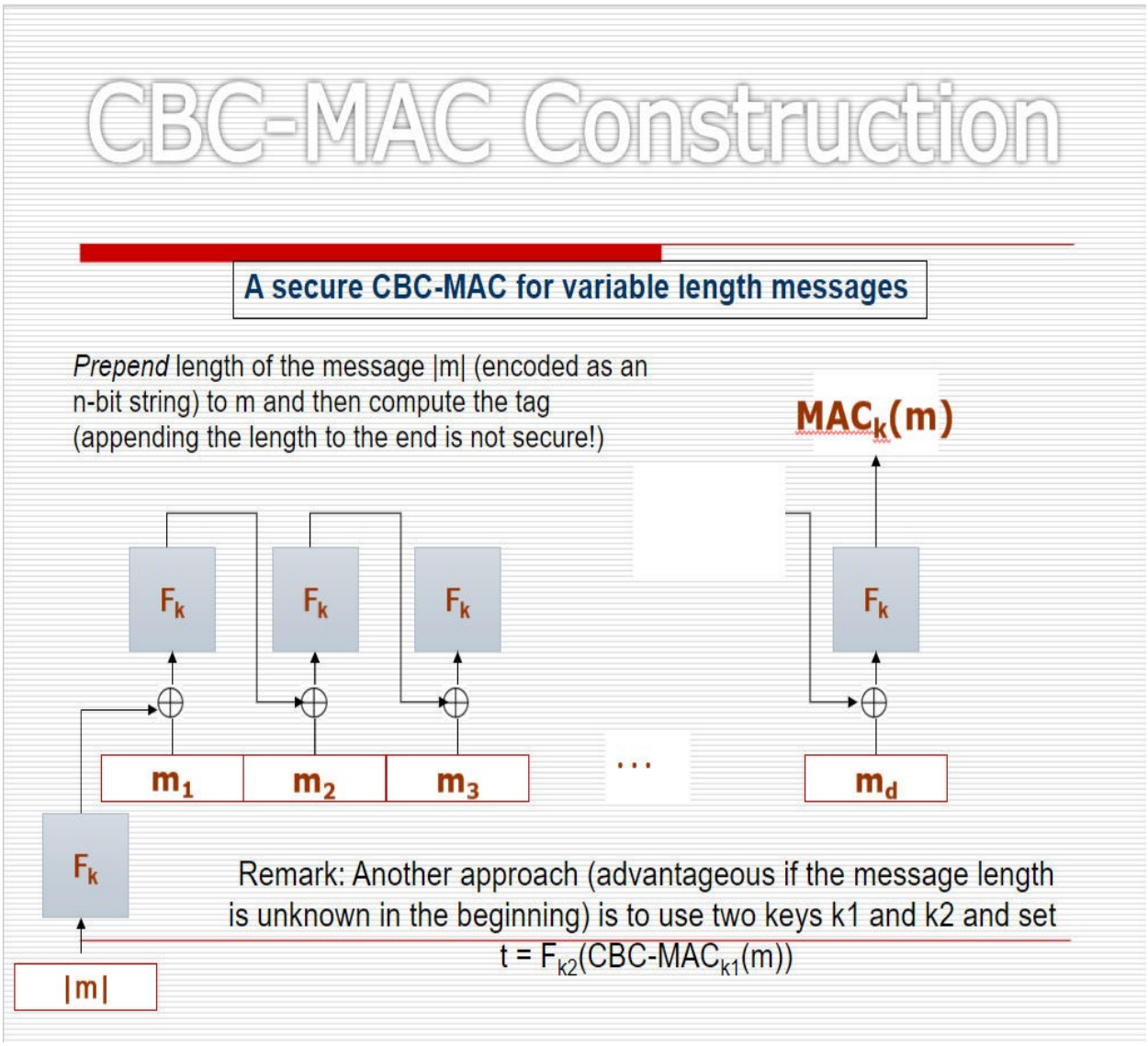
2022201041

Use the PRF to build a secure MAC

We will follow the below diagram for secure CBC-MAC construction for variable length

Messages.



Here, I’ve used previously built PRG and PRF to construct this CBC-MAC.

Working Flow:

1. First, we asked for input: prime(p), generator(g), key, data, block-size.

2. It will then check whether the data length is multiple of (block-size = key-size). If not,

then it will pad zeros after the data to make the data length multiple of the key length.

3. Then, it will divide the data into blocks of block-size.

4. Then it will follow the CBC construction[1].

5. In first step, the binary encoded data length will passed through the PRF. The obtained prf result will be then xored with the first message block and passed through the PRF again.

6. For the next steps, the obtained prf from the previous iteration and the corresponsing msg block will be xored and then passed through PRF

In this way, the last output from the PRF block will be used as MAC TAG.

To prove the security of CBC-MAC, we need to show that it satisfies the following security properties:

1.Existential unforgeability under chosen message attack: An attacker should not be able to forge a valid authentication tag for any message that they have not seen before.

2.Strong unforgeability under chosen message attack: An attacker should not be able to forge a valid authentication tag for any message, even if they have seen other messages and their authentication tags.

3.Message integrity: An attacker should not be able to modify a message without being detected by the receiver.

4.Message authenticity: A receiver should be able to verify that a received message was sent by the expected sender and has not been tampered with.

Existential unforgeability under chosen message attack can be proven by showing that, given a message m and its authentication tag t, an attacker cannot find a different message m' and its authentication tag t' such that t = t' and m ≠ m'. This property holds for CBC-MAC because it is based on a block cipher, which is assumed to be a secure pseudorandom permutation.

Strong unforgeability under chosen message attack can be proven by showing that, given authentication tags for many messages, an attacker cannot forge a new authentication tag for a message that they have not seen before. This property also holds for CBC-MAC because it is based on a secure pseudorandom permutation and uses the last block of the message as the IV, which ensures that each message has a unique encryption key.

Message integrity is achieved because any modification to the message will result in a different authentication tag, which the receiver will reject.

Message authenticity is achieved because the receiver can verify the authenticity of the message by computing the authentication tag using the same key and algorithm as the sender. If the computed tag matches the received tag, then the message is authentic.

In summary, CBC-MAC is a secure message authentication code that provides message integrity and authenticity. It is based on a secure pseudorandom permutation and uses the last block of the message as the IV, which ensures that each message has a unique encryption key. It satisfies the security properties of existential unforgeability under chosen message attack, strong unforgeability under chosen message attack, message integrity, and message authenticity.