Class 4

Block Cipher Modes

A block cipher mode is a mechanism to encrypt a plaintext **P** to a ciphertext **C** for a text and ciphertext with a variably length.

Block ciphers encrypt a fixed data block.

Padding

Required to uniformly format the length of the plaintext **P**. Some ciphers require that the length of the plaintext **P** be an exact multiple of the block size.

Padding must be reversible.

Padding from simply appending zeros (P||0) until achieving a suitable length NOT reversible.

Length of P plaintext string to be padded is < than the length of the padded plaintext.

Block Cipher Modes

Possible ways to pad a plaintext:

If **P** is plaintext and length of **P** (bytes).

Let ***b*** be the block of the cipher (bytes).

1.1 Append single byte with value 128

1.2 Append 0's as needed until the length is multiple of **b**

1.3 The number of zero bytes added is in the range of 0,…,*b*-1

Alternative Padding way:

2.1 Find the number of bytes require to pad (*n*) such:

1 ≤ *n* ≤ *b* *and n+l(P)* is a multiple of *b*

Pad the plaintext by appending n bytes, each with value *n*

Or,

3.1 Include **l(P)** at the beginning follow by **P** then pad to a block boundary.

Block Cipher Modes

After padding then:

$$P \rightarrow P_1 \ldots P_k$$

where the number of blocks *k*,

$$[l(P) + 1)/b]$$

Once the plaintext is decrypted the padding needs to be removed.

Verification of proper padding elimination so integration needs to be done, to avoid possible authentication issues.

Electronic Code Book (EBC)

Simplest method to encrypt a longer plaintext

$$C_i = E(K, P_i) \quad \text{for} \quad i = 1...k$$

NO SECURE

What if $P_1 = P_2$ then $C_1 = C_2$

Cipher Block Chaining (CBC)

Widely used:

$$C_i = E(K, P_i \oplus C_{i-1}) \quad \text{for} \quad i = 1...k$$

if $P_1 = P_2$ then $C_1 \neq C_2$

How about C_0 known as Initialization Vector (IV)?

Types of Initialization Vectors

Fixed IV

Not used, introduce similar problems as ECB

Counter IV

IV=0, IV=1,…., not good idea, why?

Radom IV

If a random IV is used, how the recipient of the message knows about it?

Solution is to:

Random block value C_0 for $i = 1...k$

$$C_i = E(K, P_i \oplus C_{i-1})$$

As long as:

$$P_1, \ldots, P_k \rightarrow C_0, \ldots, C_k$$

The disadvantage is that the cipher text is one block longer than a plaintext, decryption mechanism is given by:

$$P_i := D(K, C_i) \oplus C_{i-1}$$

Types of Initialization Vectors

Nonce-Generated IV

1. Unique number **NONCE** (Number used oNCE) assigned to each message to be encrypted.

Preparing the message to be send:

1. Message number assignation start 0.

2. Build the nonce using the message number generated. Has to be unique for *system* and should be as large as a single block of the block cipher.

3. Generated the IV by encrypting the nonce with the block cipher.

4. Using the IV and CBC mode encrypt the message.

5. Verification of the reconstruction of the cipher by aggregating information is needed such the receiver can decrypt the message. Lower number of 32-48 bits compared to a 128 bits if random IV is used. The nonce is secretly transmitted via a pre-established secure channel, nonce should be encrypted using an alternate key if there is a lack of it.

Output Feedback Mode (OFB)

Output feedback mode (OFB)

The block cipher is used to generate a *key stream*, a pseudo random string.

$$K_0 := IV$$
$$K_i := E(K_i, K_{i-1})$$
$$C_i := P_i \oplus K_i$$

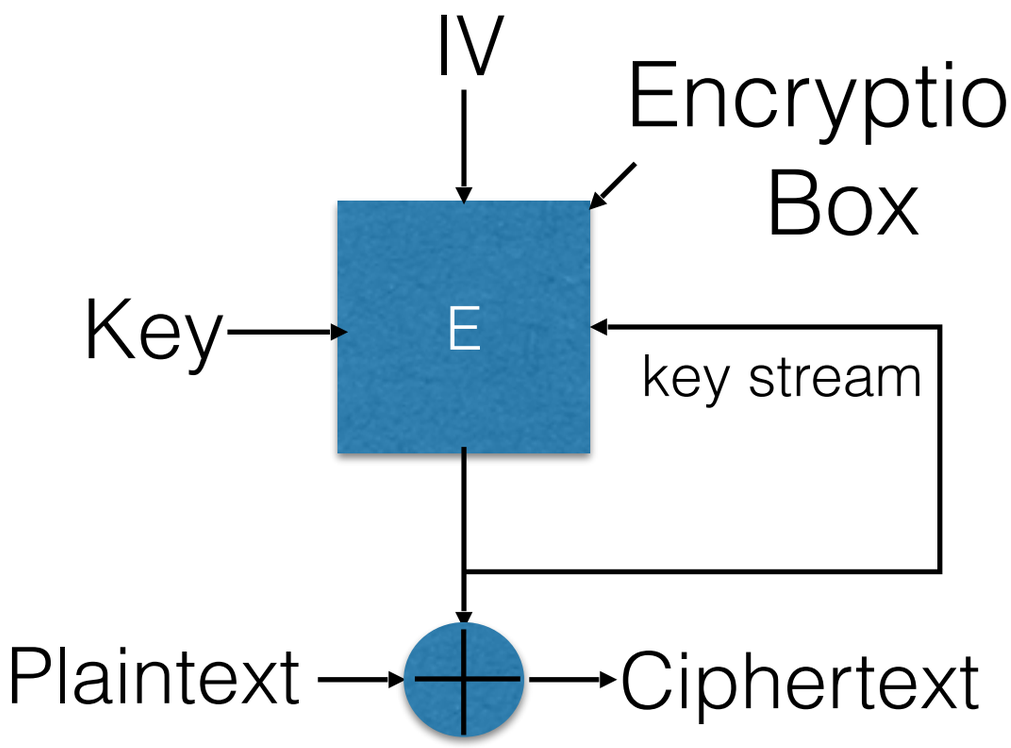
Stream cipher

$for\ i = 1,\ldots,k$

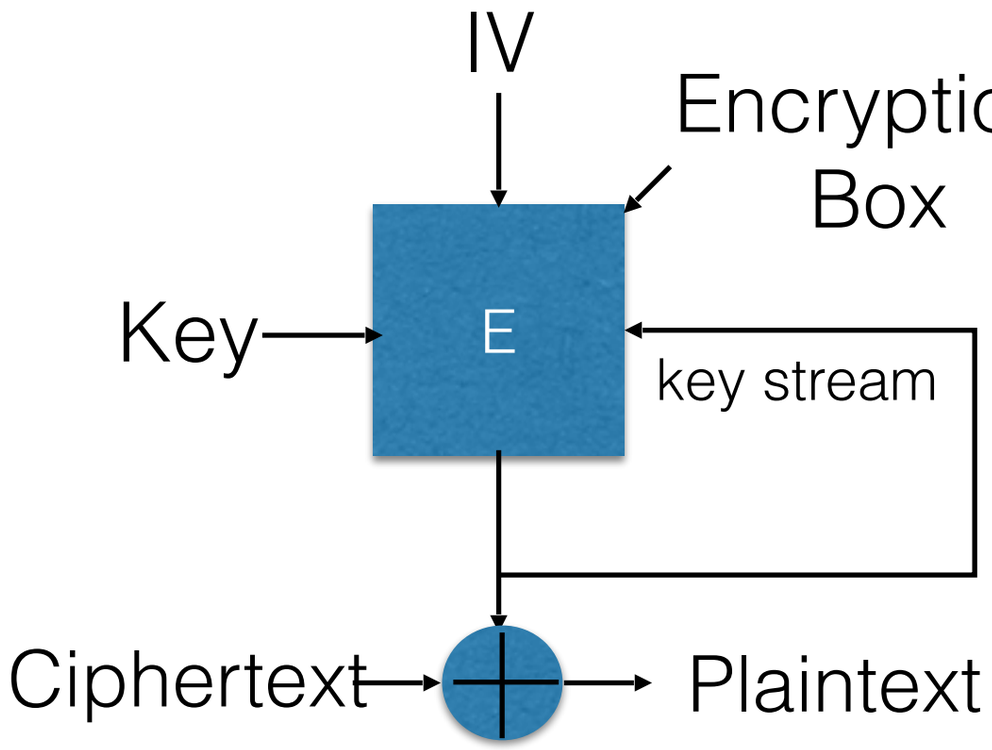
Ciphertext

Trade off between usability and careful implementation process

Output Feedback Mode (OFB)



Encryption



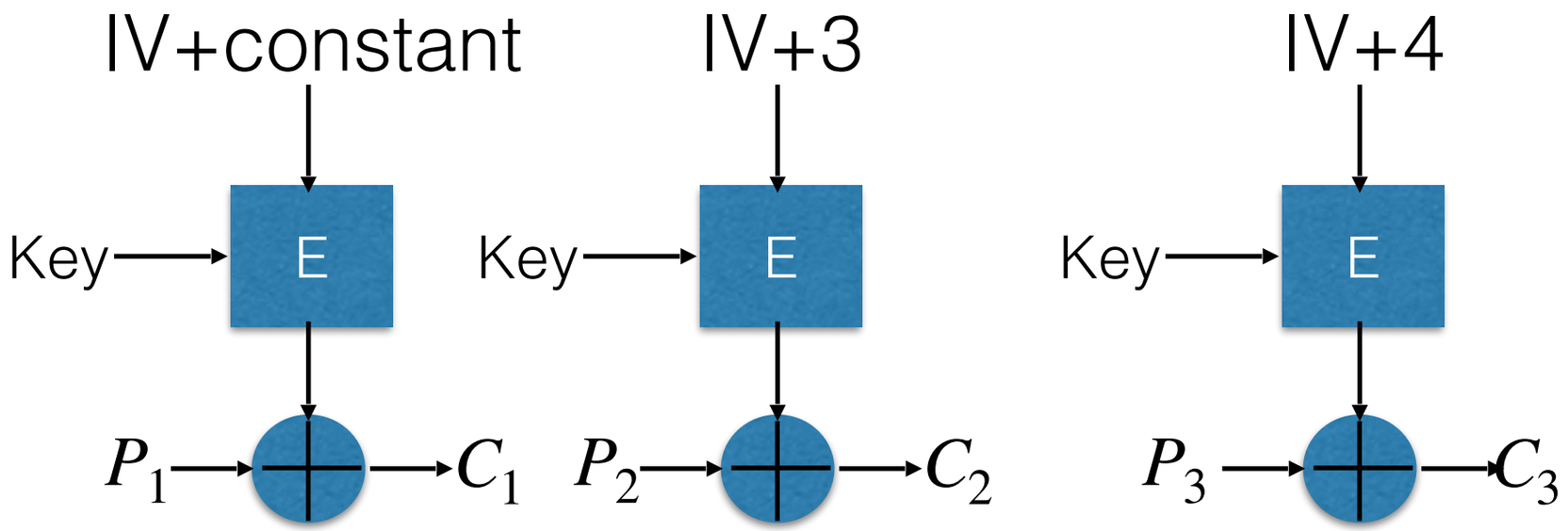
Decryption

Counter Mode (CTR)

Is a stream cipher mode

$$K_i := E(K, Nonce||i) \quad for\ i = 1,\ldots,k$$

$$C_i := P_i \oplus K_i$$



Counter Mode

For a standard setup could use:

A 48 bit message number

16 bits more for nonce data

64 bits for the counter *i*

System is limited to encrypt 2^{48} *messages using a single key*

each message is limited to 2^{68} *bytes*

Requirement to ensure that IV and nonce is unique

Which mode will you use?

CBC or CTR

CBC with random IV?

Nonce generation an important factor to chose the right mode.

All block cipher modes leak partial information.

Chances of a Collision

What is the likelihood of having a collision or two ciphertext with the same block?

If M is the total blocks encrypted, consider the total number of blocks pairs, something like

$$\frac{M(M-1)}{2}$$

The chance of each pair of being equal is

n is the block size of the block cipher 2^{-n}

$$\frac{M(M-1)}{2^{n+1}}$$

or

$$M = 2^{\frac{n}{2}}$$

Chances of a Collision

if n=128, when we could expect the first collision?

Estimate the amount size of the data set generated for this value of n.

Working around information leakage

CTR Vs CBC which one is more efficient in terms of limiting the amount of information leaked?