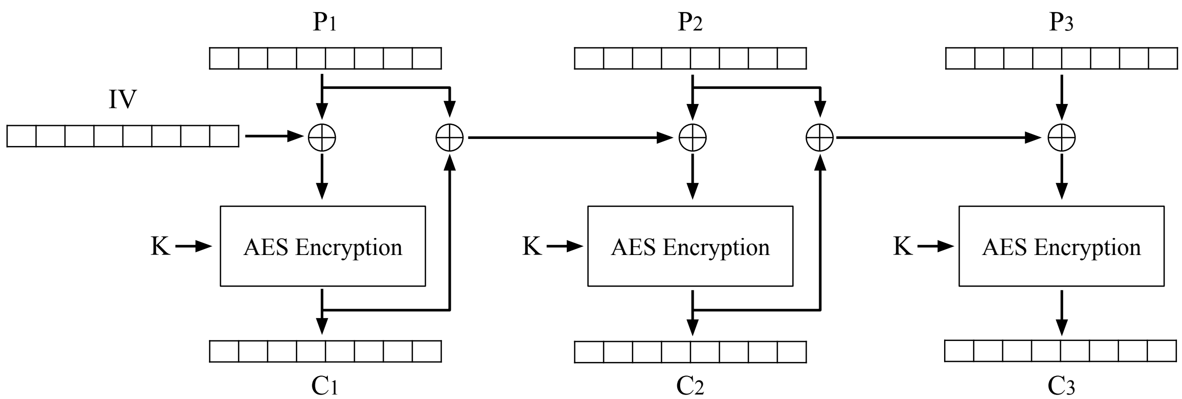
Peyrin Kao
Fall 2023

CS 161
Computer Security

Exam Prep 4

## Q1 *EvanBlock Cipher* **(24 points)**

EvanBot invents a new block cipher chaining mode called the EBC (EvanBlock Cipher). The encryption diagram is shown below:




Q1.1 (2 points) Write the encryption formula for $C_i$, where $i > 1$. You can use $E_K$ and $D_K$ to denote AES encryption and decryption respectively.

> **Solution:** $C_1 = E_K(P_1 \oplus IV)$
> $C_i = E_K(P_i \oplus P_{i-1} \oplus C_{i-1})$

Q1.2 (2 points) Write the decryption formula for $P_i$, where $i > 1$. You can use $E_K$ and $D_K$ to denote AES encryption and decryption respectively.

> **Solution:** $P_1 = D_K(C_1) \oplus IV$
> $P_i = D_K(C_i) \oplus P_{i-1} \oplus C_{i-1}$

Q1.3 (4 points) Select all true statements about this scheme.

■ It is IND-CPA secure if we use a random IV for every encryption.

☐ It is IND-CPA secure if we use a hard-coded, constant IV for every encryption.

☐ Encryption can be parallelized.

☐ Decryption can be parallelized.

☐ None of the above

**Solution:** This scheme actually exists in real life; it's called AES-PCBC, where PCBC stands for Propagating Cipher Block Chaining Mode. (The CBC here is the same as the CBC in AES-CBC.)

AES-PCBC is IND-CPA secure with random IVs. Intuitively, notice that AES-PCBC looks quite similar to AES-CBC, except we are sending both the ciphertext and plaintext into the next block cipher encryption, instead of just the ciphertext.

If we use the same IV for every encryption, AES-PCBC is deterministic, so it's not IND-CPA secure.

Encryption cannot be parallelized because you have to wait for the current block's ciphertext to be computed (which requires the current block cipher encryption to run) before you can pass the current block's ciphertext into the next block cipher encryption.

Decryption cannot be parallelized because you have to wait for the current block's plaintext to be computed (which requires the current block cipher decryption to run) before you can pass the current block's plaintext into the XOR that computes the next block's plaintext.

Q1.4 (4 points) Alice has a 4-block message $(P_1, P_2, P_3, P_4)$. She encrypts this message with the scheme and obtains the ciphertext $C = (IV, C_1, C_2, C_3, C_4)$.

Mallory tampers with this ciphertext by changing the $IV$ to 0. Bob receives the modified ciphertext $C' = (0, C_1, C_2, C_3, C_4)$.

What message will Bob compute when he decrypts the modified ciphertext $C'$?

$X$ represents some unpredictable "garbage" output of the AES block cipher.

○ $(P_1, P_2, P_3, P_4)$ ○ $(X, X, P_3, P_4)$ ● $(X, X, X, X)$

○ $(X, P_2, X, P_4)$ ○ $(X, P_2, P_3, P_4)$ ○ None of the above

**Solution:** Modifying any ciphertext block in AES-PCBC will cause itself and all future plaintext blocks to become garbage (hence the "propagate").

Alice has a 3-block message $(P_1, P_2, P_3)$. She encrypts this message with the scheme and obtains the ciphertext $C = (IV, C_1, C_2, C_3)$.

Mallory tampers with this ciphertext by swapping two blocks of ciphertext. Bob receives the modified ciphertext $C' = (IV, C_2, C_1, C_3)$.

When Bob decrypts the modified ciphertext $C'$, he obtains some modified plaintext $P' = (P'_1, P'_2, P'_3)$. In the next three subparts, write expressions for $P'_1$, $P'_2$, and $P'_3$.

Q1.5 (4 points) $P'_1$ is equal to these values, XORed together. Select as many options as you need.

For example, if you think $P'_1 = P_1 \oplus C_2$, then bubble in $P_1$ and $C_2$.

■ $P_1$ ■ $P_2$ □ $P_3$ ■ $IV$ ■ $C_1$ □ $C_2$ □ $C_3$

**Solution:**
We denote the "original" ciphertext blocks by $C_i$ and the modified ciphertext blocks by $C'_i$. For example, $C'_1 = C_2$ in our given scheme. This is likewise the case for plaintext blocks.

We have $C_1 = E_K(P_1 \oplus IV)$ and $C_2 = E_K(P_2 \oplus C_1 \oplus P_1)$ from the encryption/decryption formulas.

After swapping, when we decrypt $P_1$, we plug in $C_2$'s value for $C'_1$:

$$P'_1 = D_K(C'_1) \oplus IV$$
$$P'_1 = D_K(C_2) \oplus IV$$
$$P'_1 = D_K(E_K(P_2 \oplus C_1 \oplus P_1)) \oplus IV$$
$$P'_1 = P_2 \oplus C_1 \oplus P_1 \oplus IV$$

Q1.6 (4 points) $P'_2$ is equal to these values, XORed together. Select as many options as you need.

☐ $P_1$ ■ $P_2$ ☐ $P_3$ ☐ $IV$ ■ $C_1$ ■ $C_2$ ☐ $C_3$

**Solution:**

We have $C_1 = E_K(P_1 \oplus IV)$ and $C_2 = E_K(P_2 \oplus C_1 \oplus P_1)$.

We know from the previous subpart that $P'_1 = P_2 \oplus C_1 \oplus P_1 \oplus IV$. Key to this problem is that the decryption formulas will use the "new" values $P', C'$ for all values since that's what Bob receives/decrypts.

After swapping, when we decrypt $P_2$, we plug in $C_1$'s value:

$$
\begin{aligned}
P'_2 &= D_K(C'_2) \oplus P'_1 \oplus C'_1 \\
P'_2 &= D_K(C_1) \oplus P'_1 \oplus C'_1 \\
P'_2 &= D_K(E_K(P_1 \oplus IV)) \oplus P'_1 \oplus C'_1 \\
P'_2 &= (P_1 \oplus IV) \oplus P'_1 \oplus C'_1 \\
P'_2 &= (P_1 \oplus IV) \oplus (P_2 \oplus C_1 \oplus P_1 \oplus IV) \oplus C_2 \\
P'_2 &= P_2 \oplus C_1 \oplus C_2
\end{aligned}
$$

Q1.7 (4 points) $P'_3$ is equal to these values, XORed together. Select as many options as you need.

☐ $P_1$ ☐ $P_2$ ■ $P_3$ ☐ $IV$ ☐ $C_1$ ☐ $C_2$ ☐ $C_3$

**Solution:**

We know $P'_2 = P_2 \oplus C_1 \oplus C_2$ from the previous subpart and $C_3 = E_K(P_3 \oplus P_2 \oplus C_2)$.

Plug in decryption formula for $P_3$:

$$
\begin{aligned}
P'_3 &= D_K(C'_3) \oplus P'_2 \oplus C'_2 \\
P'_3 &= D_K(C_3) \oplus P'_2 \oplus C'_2 \\
P'_3 &= D_K(E_K(P_3 \oplus P_2 \oplus C_2)) \oplus P'_2 \oplus C'_2 \\
P'_3 &= (P_3 \oplus P_2 \oplus C_2) \oplus (P_2 \oplus C_1 \oplus C_2) \oplus C_1 \\
P'_3 &= P_3
\end{aligned}
$$

This turns out to be a unintended side effect of PCBC (and not a very good one).

## Q2 ***AES-GROOT*** **(30 points)**

Tony Stark develops a new block cipher mode of operation as follows:

$$
\begin{aligned}
C_0 &= IV \\
C_1 &= E_K(K) \oplus C_0 \oplus M_1 \\
C_i &= E_K(C_{i-1}) \oplus M_i \\
C &= C_0 \| C_1 \| \cdots \| C_n
\end{aligned}
$$

For all parts, assume that $IV$ is randomly generated per encryption unless otherwise stated.

Q2.1 (3 points) Write the decryption formula for $M_i$ using AES-GROOT.

**Solution:**

$$
\begin{aligned}
M_1 &= C_1 \oplus E_K(K) \oplus IV \\
M_i &= C_i \oplus E_K(C_{i-1})
\end{aligned}
$$

Q2.2 (3 points) AES-GROOT is not IND-CPA secure. Which of the following most accurately describes a way to break IND-CPA for this scheme?

- ● It is possible to compute a deterministic value from each ciphertext that is the same if the first blocks of the corresponding plaintexts are the same.
- ○ $C_1$ is deterministic. Two ciphertexts will have the same $C_1$ if the first blocks of the corresponding plaintexts are the same.
- ○ It is possible to learn the value of $K$, which can be used to decrypt the ciphertext.
- ○ It is possible to tamper with the value of $IV$ such that the decrypted plaintext block $M_1$ is mutated in a predictable manner.

**Solution:** The first block of ciphertext is, in fact, non-deterministic since it's XORed with a random IV. However, this doesn't provide any useful security since it's easy to just XOR out the IV and reveal the value of $E_K(K) \oplus M_1$, which is deterministic.

It is not possible to leak the value of $K$, and tampering with the $IV$ does break integrity, but this does not inherently violate IND-CPA (though it might break other threat models such as IND-CCA).

Q2.3 (5 points) AES-GROOT is vulnerable to plaintext recovery of the first block of plaintext. Given a ciphertext $C$ of an unknown plaintext $M$ and different plaintext-ciphertext pair $(M', C')$, provide a formula to recover $M_1$ in terms of $C_i$, $M'_i$, and $C'_i$ (for any $i$, e.g. $C_0$, $M'_2$, $C'_6$).

Recall that the $IV$ for some ciphertext $C$ can be referred to as $C_0$.

**Solution:** Like previously, we can XOR out the value of $C_0 = IV$, and, because we know the value of $C'_1$ and $M'_1$ in our plaintext-ciphertext pair, we can derive the value of $E_K(K) = C'_1 \oplus C'_0 \oplus M'_1$. Thus, to learn $M_1$, we compute

$$\begin{aligned} M_1 &= C_1 \oplus C_0 \oplus C'_1 \oplus C'_0 \oplus M'_1 \\ &= (E_K(K) \oplus C_0 \oplus M_1) \oplus C_0 \oplus (E_K(K) \oplus C'_0 \oplus M'_1) \oplus C'_0 \oplus M'_1 \\ &= M_1 \end{aligned}$$

If AES-GROOT is implemented with a fixed $IV = 0^b$ (a fixed block of $b$ 0's), the scheme is vulnerable to full plaintext recovery under the chosen-plaintext attack (CPA) model. Given a ciphertext $C$ of an unknown plaintext and different plaintext-ciphertext pair $(M', C')$, describe a method to recover plaintext block $M_4$.

Q2.4 (5 points) First, the adversary sends a value $M''$ to the challenger. Express your answer in terms of in terms of $C_i$, $M'_i$, and $C'_i$ (for any $i$).

**Solution:** We need to learn the value of $E_K(C_3)$ in order to recover the value of $M_4$. Since the $IV$ is fixed at $0^b$, we can send some message with $M''_1 = E_K(K) \oplus C_3$ and $M''_2 = 0^b$ ino rder to learn the $E_K(C_3)$. To do this, we first need to derive an expression for $E_K(K)$. Given $(M', C')$, we know that we can XOR out $M'_1$ from $C'_1$ to arrive at

$$\begin{aligned} E_K(K) &= C'_1 \oplus M'_1 \\ &= E_K(K) \oplus 0^b \oplus M'_1 \oplus M'_1 \\ &= E_K(K) \end{aligned}$$

Once we have this expression, we send

$$\begin{aligned} M''_1 &= C'_1 \oplus M'_1 \oplus C_3 \\ M''_2 &= 0^b \\ M'' &= M''_1 \| M''_2 \end{aligned}$$

The first block of the resulting ciphertext is $C''_1 = E_K(K) \oplus 0^b \oplus E_K(K) \oplus C_3 = C_3$. Because of this, the second resulting ciphertext block is $C''_2 = E_K(C_3) \oplus 0^b = E_K(C_3)$.

Q2.5 (5 points) The challenger sends back the encryption of $M''$ as $C''$. Write an expression for $M_4$ in terms of $C_i$, $M'_i$, $C'_i$, $M''_i$, and $C''_i$ (for any $i$).

**Solution:** Now that we have $C''_2 = E_K(C_3)$, we can simply XOR out that value from $C_4 = E_K(C_3) \oplus M_4$. The resulting expression is

$$\begin{aligned} M_4 &= C_4 \oplus C''_2 \\ &= E_K(C_3) \oplus M_4 \oplus E_K(C_3) \\ &= M_4 \end{aligned}$$

Q2.6 (4 points) Which of the following methods of choosing $IV$ allows an adversary under CPA to fully recover an arbitrary plaintext (not necessarily using your attack from above)? Select all that apply.

☐ $IV$ is randomly generated per encryption

■ $IV = 1^b$ (the bit 1 repeated $b$ times)

■ $IV$ is a counter starting at 0 and incremented per encryption

■ $IV$ is a counter starting at a randomly value chosen once during key generation and incremented per encryption

☐ None of the above

**Solution:** The above attack is possible with any method of choosing $IV$ that's predictable.

Q2.7 (2 points) Let $C$ be the encryption of some plaintext $M$. If Mallory flips with the last bit of $C_3$, which of the following blocks of plaintext no longer decrypt to its original value? Select all that apply.

☐ $M_1$

■ $M_3$

☐ None of the above

☐ $M_2$

■ $M_4$

**Solution:** We see $M_i$ depends on $C_i$ and $C_{i-1}$. That implies that a change in $C_3$ will result in a change of $M_3$ and $M_4$.

Q2.8 (3 points) Which of the following statements are true for AES-GROOT? Select all that apply.

- ☐ Encryption can be parallelized
- ■ Decryption can be parallelized
- ☐ AES-GROOT requires padding
- ☐ None of the above

**Solution:** Decryption can be parallelized because ciphertext decryption does not depend on another plaintext block. However, encryption depends on a previous ciphertext block, so it cannot be parallelized.

Padding is not required because the plaintext blocks are simply XORed with the encryption of the previous ciphertext block, like in CFB.