

# EE 720 Lecture Notes

Autumn 2023

Saravanan Vijayakumaran

September 15, 2023

# Table of contents

<b>Preface</b>	<b>1</b>
<b>About Me</b>	<b>2</b>
<b>1 Perfectly Secret Encryption</b>	<b>3</b>
1.1 Proof of Lemma 2.7 . . . . .	3
1.1.1 Forward Direction . . . . .	3
1.1.2 Reverse Direction . . . . .	5
<b>2 Private-Key Encryption</b>	<b>7</b>
2.1 Proof that EAV-Security Implies Definition 3.9 . . . . .	7
<b>3 CCA-Security and Authenticated Encryption</b>	<b>10</b>
3.1 CBC-Mode Encryption . . . . .	10
3.2 Padding Scheme from PKCS #7 Standard . . . . .	11
3.3 Decryption Oracle from Padding Errors . . . . .	11
3.4 Padding Oracle Attack Example . . . . .	11
3.4.1 Learning $b$ . . . . .	12
3.4.2 Learning $m_2$ . . . . .	12
3.4.3 Learning $m_1$ . . . . .	13

# Preface

These notes were created to support the course *EE720: An Introduction to Number Theory and Cryptography* at IIT Bombay. This course runs in the Electrical Engineering department and is offered to both undergraduate and postgraduate students.

I have taught EE720 three times (2018–2020) using the excellent textbook [Introduction to Modern Cryptography](#) by Jonathan Katz and Yehuda Lindell. Time constraints allowed me to cover only a small subset of the book’s content. I used other sources to teach topics from abstract algebra in more detail.

These notes are meant to **supplement** the book by Katz & Lindell. They will contain the following.

- Proofs of some results stated without proof in the book.
- Expanded coverage of some material relevant to the course.

*Saravanan Vijayakumaran*  
*July 2023*

## Warning

These notes are a work in progress and may contain errors. Please email the author at [sarva@ee.iitb.ac.in](mailto:sarva@ee.iitb.ac.in) to report any errors.

# About Me

My name is [Saravanan Vijayakumaran](#). I am an Associate Professor in the [Department of Electrical Engineering](#) at [IIT Bombay](#). I am currently (mid 2023) interested in cryptocurrency blockchains and applications of zero-knowledge proofs.

# Chapter 1

## Perfectly Secret Encryption

### 1.1 Proof of Lemma 2.7

**Lemma 1.1.** *Encryption scheme  $\Pi = (\text{Gen}, \text{Enc}, \text{Dec})$  is perfectly secret if and only if it is perfectly indistinguishable.*

#### 1.1.1 Forward Direction

- Suppose a scheme  $\Pi = (\text{Gen}, \text{Enc}, \text{Dec})$  is perfectly secret.
- Consider an adversary  $\mathcal{A}$  who picks two messages  $m_0, m_1$  and gives it to  $\Pi$ .
- $\Pi$  picks a bit  $b$  randomly and a key  $k$ . It gives  $c = \text{Enc}_k(m_b)$  to  $\mathcal{A}$ .
- $\mathcal{A}$  outputs  $b'$ .
- We want to prove that  $\Pr[b = b'] = \frac{1}{2}$ . Note abuse of notation; lower case letters for random variables.

##### 1.1.1.1 Deterministic Adversary

- If  $\mathcal{A}$  is deterministic, then there exists a partition  $\mathcal{C}_0, \mathcal{C}_1$  of  $\mathcal{C}$  such that  $\mathcal{C}_0 \cap \mathcal{C}_1 = \emptyset$  and

$$\mathcal{A}(c) = \begin{cases} 0 & \text{if } c \in \mathcal{C}_0, \\ 1 & \text{if } c \in \mathcal{C}_1. \end{cases}$$

Then we have

$$\begin{aligned}\Pr[b = b'] &= \frac{\Pr[\mathcal{A}(C) = 0 \mid b = 0] + \Pr[\mathcal{A}(C) = 1 \mid b = 1]}{2} \\ &= \frac{\Pr[C \in \mathcal{C}_0 \mid b = 0] + \Pr[C \in \mathcal{C}_1 \mid b = 1]}{2}\end{aligned}$$

Consider the first term in the numerator

$$\begin{aligned}\Pr[C \in \mathcal{C}_0 \mid b = 0] &= \Pr[C \in \mathcal{C}_0 \mid M = m_0] \\ &= \sum_{c \in \mathcal{C}_0} \Pr[C = c \mid M = m_0] = \sum_{c \in \mathcal{C}_0} \Pr[C = c] \\ &= \Pr[C \in \mathcal{C}_0]\end{aligned}$$

Similarly, the second term in the numerator is equal to  $\Pr[C \in \mathcal{C}_1]$ . Plugging these back into the previous equation, we get

$$\Pr[b = b'] = \frac{\Pr[C \in \mathcal{C}_0] + \Pr[C \in \mathcal{C}_1]}{2} = \frac{1}{2}.$$

#### 1.1.1.2 Probabilistic Adversary (not exclusive from previous case)

Suppose the adversary is probabilistic.

$$\Pr[b = b'] = \frac{\Pr[\mathcal{A}(C) = 0 \mid b = 0] + \Pr[\mathcal{A}(C) = 1 \mid b = 1]}{2}$$

Consider the first term in the numerator.

$$\begin{aligned}\Pr[\mathcal{A}(C) = 0 \mid b = 0] &= \sum_{c \in \mathcal{C}} \Pr[\mathcal{A}(C) = 0 \mid b = 0, C = c] \Pr[C = c \mid b = 0] \\ &= \sum_{c \in \mathcal{C}} \Pr[\mathcal{A}(C) = 0 \mid C = c] \Pr[C = c \mid b = 0] \\ &= \sum_{c \in \mathcal{C}} \Pr[\mathcal{A}(C) = 0 \mid C = c] \Pr[C = c] \\ &= \Pr[\mathcal{A}(C) = 0].\end{aligned}$$

where the second equality follows because the adversary's decision is independent of the message once we condition on the ciphertext, and the third equality follows from Lemma 2.5 and the assumption of perfect secrecy.

Similarly, we have  $\Pr[\mathcal{A}(C) = 1 \mid b = 1] = \Pr[\mathcal{A}(C) = 1]$ . Plugging these back into the previous equation, we get

$$\Pr[b = b'] = \frac{\Pr[\mathcal{A}(C) = 0] + \Pr[\mathcal{A}(C) = 1]}{2} = \frac{1}{2}.$$

### 1.1.2 Reverse Direction

Let  $A$  be perfect secrecy and  $B$  be perfect indistinguishability. We have already proved  $A \implies B$ .

To prove  $B \implies A$ , we can prove  $A^c \implies B^c$ .

Suppose  $\Pi$  is not perfectly secret. Then there exist  $m, m' \in \mathcal{M}$  and  $c_0 \in \mathcal{C}$  such that

$$\Pr[C = c_0 \mid M = m] \neq \Pr[C = c_0 \mid M = m'].$$

WLOG, assume

$$\Pr[C = c_0 \mid M = m] > \Pr[C = c_0 \mid M = m']$$

We need to construct an adversary who can exploit this fact. A natural choice for  $m$  and  $m'$  is  $m_0 = m$  and  $m_1 = m'$ .

What should the strategy be? Consider the following strategy

$$b' = \mathcal{A}(c) = \begin{cases} 0 & \text{if } c = c_0, \\ \text{random bit} & \text{if } c \neq c_0. \end{cases}$$

We have

$$\Pr[b = b'] = \frac{\Pr[\mathcal{A}(C) = 0 \mid b = 0] + \Pr[\mathcal{A}(C) = 1 \mid b = 1]}{2}.$$

Consider the first term in the numerator.

$$\begin{aligned} \Pr[\mathcal{A}(C) = 0 \mid b = 0] &= \Pr[\mathcal{A}(C) = 0 \mid b = 0, C = c_0] \Pr[C = c_0 \mid b = 0] \\ &\quad + \Pr[\mathcal{A}(C) = 0 \mid b = 0, C \neq c_0] \Pr[C \neq c_0 \mid b = 0] \\ &= 1 \cdot \Pr[C = c_0 \mid b = 0] + \frac{1}{2} \Pr[C \neq c_0 \mid b = 0]. \end{aligned}$$

Consider the second term in the numerator.

$$\begin{aligned}\Pr[\mathcal{A}(C) = 1 \mid b = 1] &= \Pr[\mathcal{A}(C) = 1 \mid b = 1, C = c_0] \Pr[C = c_0 \mid b = 1] \\ &\quad + \Pr[\mathcal{A}(C) = 1 \mid b = 1, C \neq c_0] \Pr[C \neq c_0 \mid b = 1] \\ &= \frac{1}{2} \Pr[C \neq c_0 \mid b = 1].\end{aligned}$$

Combining these two equations, we get

$$\begin{aligned}\Pr[b = b'] &= \frac{\Pr[C = c_0 \mid b = 0] + \frac{1}{2} \Pr[C \neq c_0 \mid b = 0] + \frac{1}{2} \Pr[C \neq c_0 \mid b = 1]}{2} \\ &> \frac{\frac{1}{2} \Pr[C = c_0 \mid b = 0] + \frac{1}{2} \Pr[C = c_0 \mid b = 1] + \Pr[C \neq c_0 \mid b = 0] + \Pr[C \neq c_0 \mid b = 1]}{2} \\ &= \frac{\Pr[C = c_0] + \Pr[C \neq c_0]}{2} = \frac{1}{2}.\end{aligned}$$



## Chapter 2

# Private-Key Encryption

### 2.1 Proof that EAV-Security Implies Definition 3.9

Suppose that a private-key encryption scheme  $\Pi = (\text{Gen}, \text{Enc}, \text{Dec})$  is EAV-secure.

Let  $\text{out}_{\mathcal{A}}(\text{PrivK}_{\mathcal{A},\Pi}^{\text{eav}}(n, b))$  denote the output  $b'$  of  $\mathcal{A}$  when  $m_b$  is encrypted

We want to prove that for all PPT adversaries  $\mathcal{A}$  there is a negligible function  $\text{negl}$  such that, for all  $n$ ,

$$\left| \Pr \left[ \text{out}_{\mathcal{A}} \left( \text{PrivK}_{\mathcal{A},\Pi}^{\text{eav}}(n, 0) \right) = 1 \right] - \Pr \left[ \text{out}_{\mathcal{A}} \left( \text{PrivK}_{\mathcal{A},\Pi}^{\text{eav}}(n, 1) \right) = 1 \right] \right| \leq \text{negl}(n). \quad (2.1)$$

Since  $\Pi$  is EAV-secure, we have  $\Pr \left[ \text{PrivK}_{\mathcal{A},\Pi}^{\text{eav}}(n) = 1 \right] \leq \frac{1}{2} + \text{negl}(n)$ . It follows that

$$\begin{aligned} & \Pr \left[ \text{PrivK}_{\mathcal{A},\Pi}^{\text{eav}}(n) = 1 \right] \\ &= \Pr[b' = b] \\ &= \frac{\Pr[b' = 0 \mid b = 0] + \Pr[b' = 1 \mid b = 1]}{2} \\ &= \frac{\Pr \left[ \text{out}_{\mathcal{A}} \left( \text{PrivK}_{\mathcal{A},\Pi}^{\text{eav}}(n, 0) \right) = 0 \right] + \Pr \left[ \text{out}_{\mathcal{A}} \left( \text{PrivK}_{\mathcal{A},\Pi}^{\text{eav}}(n, 1) \right) = 1 \right]}{2} \\ &= \frac{1 - \Pr \left[ \text{out}_{\mathcal{A}} \left( \text{PrivK}_{\mathcal{A},\Pi}^{\text{eav}}(n, 0) \right) = 1 \right] + \Pr \left[ \text{out}_{\mathcal{A}} \left( \text{PrivK}_{\mathcal{A},\Pi}^{\text{eav}}(n, 1) \right) = 1 \right]}{2} \\ &\leq \frac{1}{2} + \text{negl}(n), \end{aligned}$$

The final inequality gives us

$$\Pr \left[ \text{out}_{\mathcal{A}} \left( \text{PrivK}_{\mathcal{A}, \Pi}^{\text{eav}}(n, 1) \right) = 1 \right] - \Pr \left[ \text{out}_{\mathcal{A}} \left( \text{PrivK}_{\mathcal{A}, \Pi}^{\text{eav}}(n, 0) \right) = 1 \right] \leq \text{negl}(n).$$

But the result in Equation 2.1 has an absolute value on the left hand side. So if we can also show that

$$\Pr \left[ \text{out}_{\mathcal{A}} \left( \text{PrivK}_{\mathcal{A}, \Pi}^{\text{eav}}(n, 0) \right) = 1 \right] - \Pr \left[ \text{out}_{\mathcal{A}} \left( \text{PrivK}_{\mathcal{A}, \Pi}^{\text{eav}}(n, 1) \right) = 1 \right] \leq \text{negl}(n),$$

then we will be done.

**Claim:** If  $\Pi = (\text{Gen}, \text{Enc}, \text{Dec})$  is EAV-secure, then for any PPT adversary  $\mathcal{A}$  the following must hold:  $\Pr \left[ \text{PrivK}_{\mathcal{A}, \Pi}^{\text{eav}}(n) = 1 \right] \geq \frac{1}{2} - \text{negl}(n)$ .

In other words, it is not possible for a PPT adversary to fail with probability that is too far away from  $\frac{1}{2}$ . Let us rearrange the terms in our claim.

$$\begin{aligned} \Pr \left[ \text{PrivK}_{\mathcal{A}, \Pi}^{\text{eav}}(n) = 1 \right] &\geq \frac{1}{2} - \text{negl}(n) \\ \iff \frac{1}{2} - \Pr \left[ \text{PrivK}_{\mathcal{A}, \Pi}^{\text{eav}}(n) = 1 \right] &\leq \text{negl}(n) \end{aligned}$$

We will prove our claim by contradiction. Suppose that the function  $\frac{1}{2} - \Pr \left[ \text{PrivK}_{\mathcal{A}, \Pi}^{\text{eav}}(n) = 1 \right]$  is not negligible. Then there is an adversary  $\mathcal{A}'$  and a positive polynomial  $p(n)$  such that

$$\frac{1}{2} - \Pr \left[ \text{PrivK}_{\mathcal{A}', \Pi}^{\text{eav}}(n) = 1 \right] > \frac{1}{p(n)}$$

for infinitely many  $n$ . This implies that

$$\Pr[b' = b] = \Pr \left[ \text{PrivK}_{\mathcal{A}', \Pi}^{\text{eav}}(n) = 1 \right] < \frac{1}{2} - \frac{1}{p(n)}$$

for infinitely many  $n$ , where  $b'$  is the output of  $\mathcal{A}'$  and  $b$  is the bit chosen in the experiment.

Since  $\mathcal{A}'$  is good at failing in the experiment, flipping the output of  $\mathcal{A}'$  will give us an adversary that will succeed with a probability that is significantly better than  $\frac{1}{2}$ .

Consider an adversary  $\mathcal{A}''$  who uses  $\mathcal{A}'$  as a subroutine and outputs the opposite of whatever  $\mathcal{A}'$  outputs.

$$b'' = \mathcal{A}''(c) = 1 - \mathcal{A}'(c) = \begin{cases} 0 & \text{if } b' = 1, \\ 1 & \text{if } b' = 0. \end{cases}$$

If  $\mathcal{A}'$  is PPT, then  $\mathcal{A}''$  is also PPT. Then we have

$$\begin{aligned}\Pr[b'' = b] &= \Pr[b' \neq b] = 1 - \Pr[b' = b] \\ &= 1 - \Pr[b' = b] > 1 - \left(\frac{1}{2} - \frac{1}{p(n)}\right) \\ &= \frac{1}{2} + \frac{1}{p(n)}\end{aligned}$$

for infinitely many  $n$ . This is a contradiction. Hence our claim must be true.

We then have

$$\begin{aligned}\frac{1}{2} - \Pr[\text{PrivK}_{\mathcal{A}, \Pi}^{\text{eav}}(n) = 1] &\leq \text{negl}(n) \\ \iff \frac{1}{2} - \Pr[b' = b] &\leq \text{negl}(n) \\ \iff \frac{1 - \Pr[b' = 0 \mid b = 0] - \Pr[b' = 1 \mid b = 1]}{2} &\leq \text{negl}(n) \\ \iff \frac{\Pr[b' = 1 \mid b = 0] - \Pr[b' = 1 \mid b = 1]}{2} &\leq \text{negl}(n) \\ \iff \Pr[b' = 1 \mid b = 0] - \Pr[b' = 1 \mid b = 1] &\leq 2\text{negl}(n) \\ \iff \Pr[\text{out}_{\mathcal{A}}(\text{PrivK}_{\mathcal{A}, \Pi}^{\text{eav}}(n, 0)) = 1] - \Pr[\text{out}_{\mathcal{A}}(\text{PrivK}_{\mathcal{A}, \Pi}^{\text{eav}}(n, 1)) = 1] &\leq 2\text{negl}(n).\end{aligned}$$

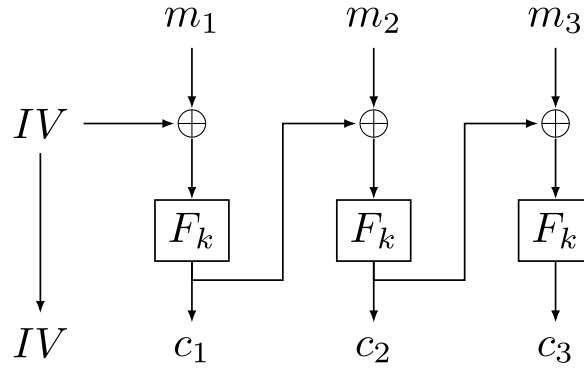
As  $2\text{negl}(n)$  is also negligible, we have the other inequality we need to prove that

$$\left| \Pr[\text{out}_{\mathcal{A}}(\text{PrivK}_{\mathcal{A}, \Pi}^{\text{eav}}(n, 0)) = 1] - \Pr[\text{out}_{\mathcal{A}}(\text{PrivK}_{\mathcal{A}, \Pi}^{\text{eav}}(n, 1)) = 1] \right| \leq \text{negl}(n).$$

## Chapter 3

# CCA-Security and Authenticated Encryption

### 3.1 CBC-Mode Encryption



Let  $\vec{m} = \langle m_1, m_2, \dots, m_l \rangle$  where  $m_i \in \{0, 1\}^n$ . Let  $F$  be a block cipher with block length  $n$ . To encrypt  $\vec{m}$ , the following steps are followed.

1. A uniform initialization vector (IV) of length  $n$  is chosen
2.  $c_0 = IV$ . For  $i = 1, \dots, l$ ,

$$c_i := F_k(c_{i-1} \oplus m_i).$$

To decrypt the ciphertext, for  $i = 1, 2, \dots, l$ ,

$$m_i := F_k^{-1}(c_i) \oplus c_{i-1}.$$

## 3.2 Padding Scheme from PKCS #7 Standard

In CBC-mode encryption, the message length is a multiple of the block length. If not, message needs to be padded.

Let  $m$  denote the original message. Assume that  $|m|$  is an integral number of bytes. Let  $L$  be the block length of  $F$  in bytes. We will assume that  $L < 256$ , so it can fit in a byte.

Let  $b$  denote the padding length in bytes. We will require that  $b$  is an integer from 1 to  $L$ , i.e.  $b = 0$  is not allowed.

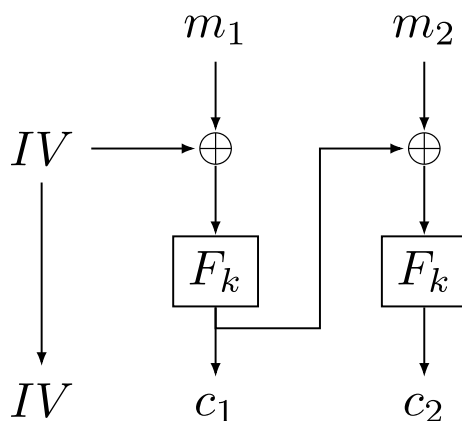
We append to the message the integer  $b$  (represented in 1 byte) repeated  $b$  times. For example, if 4 bytes are needed then 0x04040404 is appended. If  $|m|$  is already a multiple of  $L$ , then  $L$  bytes are appended each of which is equal to  $L$ .

To remove the padding, the receiver first applies the decryption algorithm on the ciphertext corresponding to encoded data. The value  $b$  of the last byte of the encoded data is read. The final  $b$  bytes of the encoded data are checked to be equal to  $b$ .

## 3.3 Decryption Oracle from Padding Errors

If padding is incorrect, the standard procedure is to return a “bad padding” error. This error message is a *partial decryption oracle*. Using these error messages, an attacker *can completely recover* the original message.

## 3.4 Padding Oracle Attack Example



Consider a 3-block ciphertext from CBC-mode encryption. The second plaintext block is obtained as

$$m_2 = F_k^{-1}(c_2) \oplus c_1.$$

Note that  $m_2$  ends in  $0xb0xb \dots 0xb$ .

Suppose an attacker changes  $c_1$  to  $c'_1$ , but keeps  $IV$  and  $c_2$  the same. Then  $m_2$  is changed to

$$m'_2 = F_k^{-1}(c_2) \oplus c'_1.$$

If the  $c_1$  and  $c'_1$  differ only in the  $i$ th byte, then  $m_2$  and  $m'_2$  also differ only in the  $i$ th byte.

### 3.4.1 Learning $b$

Suppose there is a server that receives ciphertexts and returns padding oracle error (if any). The attacker can learn  $b$  as follows.

- The attacker modifies the first byte of  $c_1$  to get  $c'_1$ , and sends  $(IV, c'_1, c_2)$  to the server
  - If error is returned, then  $b = L$
  - The server is reading all  $L$  bytes of  $m'_2$
- Otherwise, the attacker modifies the second byte of  $c_1$  and sends  $(IV, c'_1, c_2)$ . And so on.
- The first byte where decryption fails reveals  $b$

### 3.4.2 Learning $m_2$

Suppose  $b = L$ . Then the message block  $m_2$  consists of  $L$  bytes each equal to  $L$ . Only  $m_1$  remains to be recovered. The method for recovering  $m_1$  is described in the next section.

Consider the case when  $b < L$ . The attacker knows that the  $m_2$  ends with

$$0xM0xb \dots 0xb$$

where  $M$  is an unknown byte.

Define  $\Delta_i$  as

$$\begin{aligned} \Delta_i = & 0x00 \dots 0x00 \quad 0xi \quad \overbrace{0x(b+1) \dots 0x(b+1)}^{b \text{ times}} \\ & \oplus 0x00 \dots 0x00 \quad 0x00 \quad \overbrace{0xb \dots 0xb}^{b \text{ times}} \end{aligned}$$

$M$  is recovered as follows.

- Attacker sends  $(IV, c_1 \oplus \Delta_i, c_2)$  to the server
- The final  $b + 1$  bytes of the resulting encoded data are

$$0x(M \oplus i) \underbrace{0x(b+1) \cdots 0x(b+1)}_{b \text{ times}}$$

- Decryption fails unless  $0x(M \oplus i) = 0x(b+1)$
- Attacker needs to try at most 256 values for  $i$
- When decryption succeeds, attacker learns that

$$M = 0x(b+1) \oplus 0xi$$

If  $b = L - 1$ ,  $m_2$  has now been fully recovered and the attacker can proceed to recover  $m_1$ . Otherwise, the attacker knows that the  $m_2$  ends with

$$0xM0xa0xb \dots 0xb$$

for an unknown  $M$  and known  $a$ .

The attacker uses a technique similar to the one described above to recover the remaining bytes of  $m_2$ .

### 3.4.3 Learning $m_1$

In the normal scenario, message block  $m_1$  is decrypted as

$$m_1 = F_k^{-1}(c_1) \oplus IV.$$

Suppose an attacker changes  $IV$  to  $IV'$ , but keeps  $c_1$  the same. Then  $m_1$  is changed to

$$m'_1 = F_k^{-1}(c_1) \oplus IV'.$$

If the  $IV$  and  $IV'$  differ only in the  $i$ th byte, then  $m_1$  and  $m'_1$  also differ only in the  $i$ th byte.

For  $0 \leq i \leq 255$ , the attacker chooses  $\Delta_i$  defined as

$$\Delta_i = \underbrace{0\|0\| \cdots \|0\|}_{L-1 \text{ times}} \underbrace{\|i\|}_{1 \text{ byte}}$$

where  $i$  is the byte representation of the integer  $i$  and  $0$  is the byte representation of the integer  $0$ .

The attacker calculates  $IV^{(i)} = IV \oplus \Delta_i$  and sends  $(IV^{(i)}, c_1)$  to the padding oracle. Then  $m_1$  is changed to

$$m_1^{(i)} = F_k^{-1}(c_1) \oplus IV^{(i)} = F_k^{-1}(c_1) \oplus IV \oplus \Delta_i = m_1 \oplus \Delta_i.$$

The **key observation** is that there exists at least one  $i$  for which the padding oracle returns **ok**. And there may be more than one value of such an  $i$ .

Let  $m_{1,j}$  denote the  $j$ th byte of  $m_1$  for  $j \in \{1, 2, \dots, L\}$ .

- The value of  $i$  such that  $m_{1,L} \oplus i = 1$  will result in an **ok** response from the padding oracle irrespective of the values of the other bytes  $m_{1,1}, m_{1,2}, \dots, m_{1,L-1}$ . It corresponds to the case of 1 byte of padding.
- The value of  $i$  such that  $m_{1,L} \oplus i = 2$  will also result in an **ok** response if  $m_{1,L-1} = 2$ .
- The value of  $i$  such that  $m_{1,L} \oplus i = 3$  will also result in an **ok** response if  $m_{1,L-2} = m_{1,L-1} = 3$ .

To distinguish between these possible cases, the attacker first finds an  $i$  such that the padding oracle returns **ok**. At this point, the attacker knows that  $(IV^{(i)}, c_1)$  corresponds to a message  $m^{(i)}$  that ends with  $b$  bytes each equal to  $b$ , for some unknown  $b$ .

Then the attacker uses the technique in Section 3.4.1 to estimate the padding length  $b$  in the encoded data corresponding to  $(IV^{(i)}, c_1)$ . The details are left as an exercise.

Since  $m_{1,L} \oplus i = b$  and  $i$  is known, he can recover  $m_{1,L}$ . This also reveals that the values of  $b - 1$  bytes  $m_{1,L-b+1}, \dots, m_{1,L-2}, m_{1,L-1}$  are all equal to  $b$ .

If  $b = L$ , all the bytes of  $m_1$  have been recovered. Once again, consider the case when  $b < L$ . The attacker knows that the  $m_1^{(i)} = m_1 \oplus \Delta_i$  ends with

$$0xM0xb \dots 0xb$$

where  $M$  is an unknown byte. The attacker uses the technique in Section 3.4.2 to recover all the remaining bytes in  $m_1$ . The details are left as an exercise.