Selected Areas in Cryptology
Cryptanalysis
Week 2

Marc Stevens

stevens@cwi.nl

https://homepages.cwi.nl/~stevens/mastermath/2021/

Block cipher Modes of Operation

Block cipher $Enc: \{0,1\}^k \times \{0,1\}^n \to \{0,1\}^n$

..only encrypts n-bit blocks as a whole

Mode of Operation:

- A scheme to encrypt arbitrary length plaintexts M using a block cipher and a secret key K
- Deterministic Mode: same key and plaintext always give same ciphertext
- Probabilistic Mode: same key and plaintext has many possible ciphertexts

First step: Padding

- Transform M into a sequence of blocks $M_1, \ldots, M_l$
- Must be unambiguous / injective:
  - If $M \neq M'$ then $(M_1, \ldots, M_l) \neq (M'_1, \ldots, M'_l)$
- E.g., first add a `1'-bit. Then as many `0'-bits to get a bit length multiple of n

Generic key recovery attack & distinguishing attack

- Exhaustive key search with cost $O(2^k)$

Distinguishing attacks

Distinguishing Game Probabilistic Modes

Attacker gets oracle access to either:

1. Mode Oracle $\mathcal{O}_{Mode}$ with randomly chosen key K

   On query M return $C \xleftarrow{r} ModeEnc_K(M)$

2. Random Oracle $\mathcal{O}_{rnd}$ keeps list of query answers $L = \{(M, C)\}$

   Let $L = \emptyset$

   On query M:

   1. Let $\boxed{C \xleftarrow{r} \{0,1\}^{|ModeEnc_*(M)|}} \setminus \{C' | (M', C') \in L\}$
   2. Update $L \coloneqq L \cup \{(M, C)\}$ and return C

   i.e., return a random bit string of the same length as Mode

Attacker must return 0 or 1.

Deterministic Modes:

On query M step 0: check if M has been queried already $\Rightarrow$ return same ciphertext

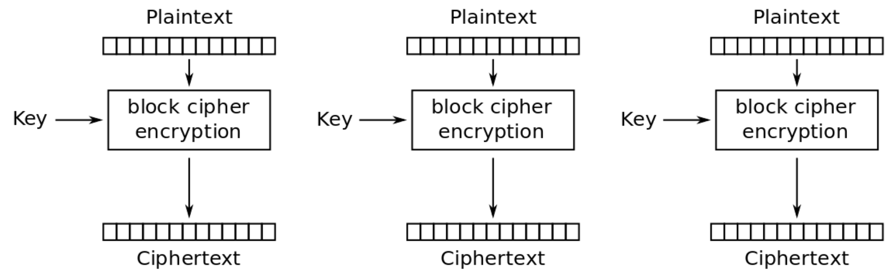
Electronic Code Book - ECB

The most simple (and problematic) deterministic mode:

Encrypt each block independently

$$C_i = Enc_K(M_i)$$

$$ECBEnc_K(M) \coloneqq C_1 || \ldots || C_l$$



Cost $O(1)$ distinguishing attack

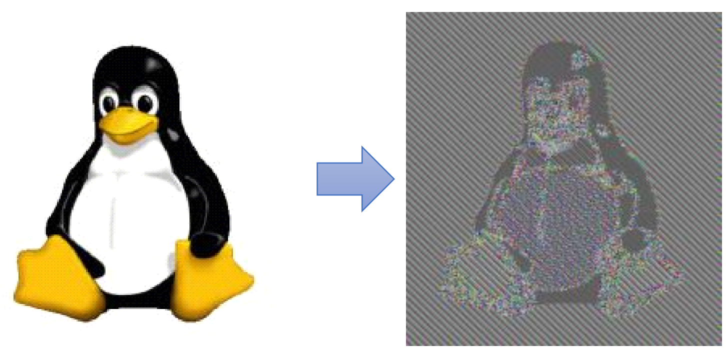
1. Query 2-block plaintext $M_1 = M_2$
2. Return 1 if $C_1 = C_2$, else 0

Probability $C_1 = C_2$ for $\mathcal{O}_{ECB}$: 1

Probability $C_1 = C_2$ for $\mathcal{O}_{rnd}$: 2^{-n}

Always leaks equivalent block structure

Limited plaintext secrecy...



Cipher Block Chaining - CBC

While ECB is a deterministic mode..

..CBC is a probabilistic mode

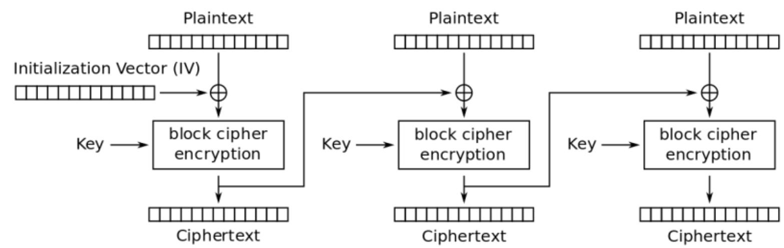
- By starting with an extra <u>random</u> Initialization Vector – IV block
- Two encryptions with the same plaintext and key are distinct

Encryption:

$$C_0 = IV \xleftarrow{r} \{0,1\}^n$$

$$C_i = Enc_K(C_{i-1} \oplus M_i)$$

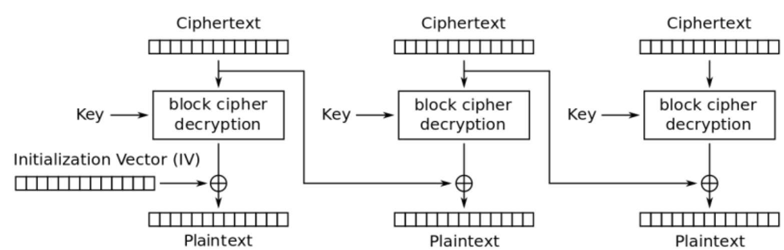
$$CBCEnc_K(M) \coloneqq C_0 || \dots || C_l$$



Cipher Block Chaining (CBC) mode encryption

Decryption:

$$M_i = Dec_K(C_i) \oplus C_{i-1}$$



Cipher Block Chaining (CBC) mode decryption

$O(2^{n/2})$-distinguishing attack

Distinguishing attack

Based on difference of behavior between:
a random permutation on $\mathcal{M}$: no collisions possible
a random function/sampling on $\mathcal{M}$: collisions occur
observable difference: first collision after $\approx \sqrt{|\mathcal{M}|}$ evaluations

Example against CBC, but works against more modes

Observation on $\mathcal{O}_{CBC}$: (with fixed secret key K)

Choose $B \in \{0,1\}^n$ and encrypt $M := B||\ldots||B$, i.e., l copies of B

Then $C_0 = IV$ (random), and $C_i = Enc_K(C_{i-1} \oplus B) =: F(C_{i-1})$

Note that F is a permutation, so $F(X) = F(Y) \Leftrightarrow X = Y$

Only two cases possible:

(1) No collision: all C_i are distinct

(2) A collision $C_i = C_j$, $i < j$ occurs:
$C_i = C_j$ implies both $C_{i-1} = C_{j-1}$ and $C_{i+1} = C_{j+1}$
Hence, a cycle: $C_{j-i} = C_0, C_{j-i+1} = C_1, \ldots$

$O\left(2^{n/2}\right)$-distinguishing attack

Observation on $\mathcal{O}_{rnd}$:

Let $C \stackrel{r}{\leftarrow} \{0,1\}^{|CBCEnc_*(M)|} \setminus \{C' | (M', C') \in L\}$

Same length, so $C = C_0 || \dots || C_l$, where $C_i \stackrel{r}{\leftarrow} \{0,1\}^n$

A collision among C_i occurs with high probability when $l \approx 2^{n/2}$

Birthday paradox:

Collection of $l + 1$ uniformly random samples C_i from a space of size N

Probability of unique samples (<u>no collision</u>!):

$$\Pr\left[C_i \neq C_j, 0 \leq i < j \leq l\right] = 1 \cdot \frac{N-1}{N} \cdots \frac{N-l}{N} = 1 \cdot \left(1 - \left(\frac{1}{N}\right)\right)\left(1 - \left(\frac{2}{N}\right)\right) \cdots \left(1 - \left(\frac{l}{N}\right)\right)$$

(multiply probabilities that C_j doesn't collide with $C_0, \dots, C_{j-1}$)

Use the approximation $e^x \approx 1 + x + O(x^2)$:

$$\Pr\left[C_i \neq C_j, 0 \leq i < j \leq l\right] \approx 1 \cdot e^{-\frac{1}{N}} \cdots e^{-\frac{l}{N}} = e^{-\frac{1+2+\cdots+l}{N}} = e^{-\frac{l(l+1)}{2N}}$$

For $l = \sqrt{N}$: $\Pr\left[C_i \neq C_j, 0 \leq i < j \leq l\right] \approx e^{-\frac{1}{2}} \approx 0.606531$

For $l = 4\sqrt{N}$: $\Pr\left[C_i \neq C_j, 0 \leq i < j \leq l\right] \approx e^{-8} \approx 0.000335$

For $l = 23$, $N = 365$: Pr ≈ 0.47 is counter-intuitive for people, hence "Birthday Paradox"

$O\left(2^{n/2}\right)$-distinguishing attack

Distinguisher A given oracle $\mathcal{O} \in \{\mathcal{O}_{CBC}, \mathcal{O}_{rnd}\}$

1. Choose $B \in \{0,1\}^n$ and let $l = 4 \cdot 2^{n/2}$
2. Let $M := M_1 || \ldots || M_l$, with $M_i = B$
3. Query $C \leftarrow \mathcal{O}(M)$
4. If no collision: $\forall 0 \leq i < j \leq l$: $C_i \neq C_j$ then return 0 (guess $\mathcal{O}_{CBC}$)
5. Otherwise, let $C_i = C_j$ with $i < j$
6. If cycle: $C_{j-i} = C_0$ and $C_{j-i+k} = C_k$ for all $0 \leq k \leq l + i - j$ then return 0
7. Else, return 1 (guess $\mathcal{O}_{rnd}$)

Analysis:

If $\mathcal{O} = \mathcal{O}_{CBC}$ then distinguisher returns 0 with probability 1

If $\mathcal{O} = \mathcal{O}_{rnd}$ then distinguisher returns 0 with probability ≈ 0.000335

Padding Oracle Attack

CBC requires padding:

$Pad(M) = (M_1, \ldots, M_l)$

Which must be unambiguous / injective:

If $M \neq M'$ then $Pad(M) \neq Pad(M')$

But is not necessarily surjective / always invertible.

What happens for a ciphertext C' resulting in $(M'_1, \ldots, M'_l)$ with invalid padding?

I.e., preimage space is empty: $Pad^{-1}(M'_1, \ldots, M'_l) = \emptyset$

An error message to sender?

Abort the connection?

Timing behavior difference?

i.e., respond very quickly with next message

If the attacker can reasonably observe distinction between valid and invalid padding then this is a Padding Oracle and may directly lead to attacks!

Padding Oracle Attack

Padding example: PKCS7 padding for byte strings

- Input message M of byte length m
- Needs to be padded to multiple of $n/8$, with at least 1 byte
- Let m' be the minimal such multiple: $m' \coloneqq (n/8)\lceil (m+1)/(n/8) \rceil$
- Amount of padding bytes $\mathrm{r} \coloneqq m' - m > 0$
- $Pad(M) = M||r||r|| \cdots ||r$ i.e., M padded with r bytes with value r

Definition Padding Oracle $\mathcal{O}_{pad}$

- Uniformly random chosen secret key $K \in \{0,1\}^k$
- On query input $C \in \{0,1\}^{(l+1)\cdot n}$ for $l \in \mathbb{N}$
  - Decrypts $M_i = Dec_K(C_i) \oplus C_{i-1}$ for $i = 1, \dots, l$
  - Let r be the last byte of M_l
  - If M_l ends with r bytes of value r then return *True*
    else return *False*

Padding Oracle Attack

Goal: Recover message M for ciphertext C using $\mathcal{O}_{pad}$

Idea of padding oracle attack

- Oracle output depends only on last block:
  - $M_l := Dec_K(C_l) \oplus C_{l-1}$
- Let C_l be fixed (with $l \geq 1$)
- Modifying $C'_{l-1} := C_{l-1} \oplus D$ implies $M'_i := M_i \oplus D$

- Oracle provides bit of information on <u>last unknown byte</u>
  - True in at most 2 cases:
    1. last byte has value $r = 1$
    2. last byte has value $r > 1$ and last r bytes also have value r
  - For now, let's ignore case 2 => exercise to figure out what happens

Padding Oracle Attack

Given C_{l-1}, C_l (from a real ciphertext)

Let $M_l \coloneqq Dec_K(C_l) \oplus C_{l-1}$

Consists of b $\coloneqq$ n/8 bytes: $M_l[1], \ldots, M_l[b]$

For $i = b, \ldots, 1$:

// Assume bytes $i+1, \ldots, b$ of M_l are known, learn byte i of M_l as follows

// Target padding of r bytes of value r, where byte i is first byte of padding

Let $r \coloneqq b - i + 1$

For $x = 0, \ldots, 255$:

Let $D \coloneqq 0||\cdots||0||x||(M_l[i+1] \oplus r)||\cdots||(M_l[b] \oplus r)$

Let $C'_{l-1} \coloneqq C_{l-1} \oplus D$ // results in $M'_l \coloneqq M_l \oplus D$

// Where $M'_l = \boxed{M_l[1]||\cdots||M_l[i-1]||}\boxed{(M_l[i] \oplus x)}\underbrace{\boxed{||r||\cdots||r}}_{r-1\text{ bytes}}$

If $\mathcal{O}_{pad}(C_0||\cdots||C_{l-2}||C'_{l-1}||C_l) = True$ then

Found $M_l[i] \coloneqq x \oplus r$

Continue with next i

Padding Oracle Attack

Learns value of last message block M_l

Assuming we don't end in case 2 for $i = b$:

2. last byte has value $r > 1$ and last r bytes also have value r

   Attack will fail (exercise: how and when?)

   Resolve by restart attack and avoid the bad value for the last byte

How can we learn the other message blocks?

Repeat the attack with $C \coloneqq C_0 || \cdots || C_{l-1}$ and learn M_{l-1}

Total attack cost:

At most 256 oracle calls per message byte

$O(|M|)$

Other Modes

Other Modes (see also lecture notes)

- ECB – Electronic Code Book
- CBC – Cipher Block Chaining
- CFB – Cipher Feedback
  - Probabilistic mode, no padding needed
  - Encryption: $C_0 = IV \stackrel{r}{\leftarrow} \{0,1\}^n, \quad C_i = M_i \oplus Enc_K(C_{i-1})$
  - Decryption: $M_i = C_i \oplus Enc_K(C_{i-1})$
- OFB – Output Feedback
  - Probabilistic mode, no padding needed
  - Encryption: $C_0 = O_0 = IV \stackrel{r}{\leftarrow} \{0,1\}^n, \quad O_i = Enc_K(O_{i-1}), \quad C_i = M_i \oplus O_i$
  - Decryption: $O_0 = IV, \quad O_i = Enc_K(O_{i-1}), \quad M_i = C_i \oplus O_i$
- CTR – Counter mode: c-bit counter, $(n-c)$-bit IV
  - Probabilistic mode, no padding needed
  - Encryption: $C_0 = IV \leftarrow \{0,1\}^{n-c}, \quad C_i = M_i \oplus Enc_K(IV||i)$
  - Decryption: $M_i = C_i \oplus Enc_K(IV||i)$

Other Modes

Mode	Gen Key Rec	Distinguisher	Padding Oracle?
ECB (determ)	$O(2^k)$	$O(1)$	N/A
CBC	$O(2^k)$	$O(2^{n/2})$	$O(\|M\|)$
CFB (no pad)	$O(2^k)$	$O(2^{n/2})$	.. Only if
OFB (no pad)	$O(2^k)$	$O(2^{n/2})$	.. padding
CTR (no pad)	$O(2^k)$	$O(\min(2^c, 2^{n/2}))$	.. is used