Lecture 4 – Blockciphers

COSC-260 Codes and Ciphers
Adam O’Neill

Adapted from http://cseweb.ucsd.edu/~mihir/cse107/
Setting the Stage

Perfect security $\Rightarrow$ keys as long as messages.
Setting the Stage

Perfect security => keys as long as messages.
Perfect security => keys as long as messages.

From now on we move to the setting of computationally-bounded adversaries.
Setting the Stage

Perfect security $\Rightarrow$ keys as long as messages.

From now on we move to the setting of computationally-bounded adversaries.
Setting the Stage

Perfect security => keys as long as messages.

From now on we move to the setting of computationally-bounded adversaries.

Today: first lower-level primitive, blockciphers.
What is a Blockcipher?

Let

\[ E: \{0,1\}^k \times \{0,1\}^l \rightarrow \{0,1\}^l \]

be a function taking a key \( K \) and input \( x \) to return output \( E(K, x) \). For each key \( K \) we let \( E_K: \{0,1\}^l \rightarrow \{0,1\}^l \) be the function defined by

\[ E_K(x) = E(K, x) \]

We say that \( E \) is a blockcipher if

- \( E_K(\cdot) \) is a permutation for every key \( K \).
- \( E_K, E_K^{-1} \) are efficiently computable.
**Blockcipher Examples**

Suppose \( k = l = 2 \).

\[
\begin{align*}
\begin{cases}
0 &\rightarrow 0 \\
0 &\rightarrow 0 \\
1 &\rightarrow 1 \\
1 &\rightarrow 1 \\
\end{cases}
\end{align*}
\]

\[E_k(x) = K \oplus x\]

\( k = l = 128 \)

In general, \( E_k(x) \) would take \( 128 \cdot 2^{128} \) bits to represent, which would not fit in a universe.
Blockcipher Usage

$K \oplus 0^t = K$

$E_k(x_1), \ldots, E_k(x_n)$

Encryption of block $x \in \{0,1\}^t$

Alice

Bob

Eve
History of DES

1972 – NBS (now NIST) asked for a block cipher for standardization

1974 – IBM designs Lucifer

Lucifer eventually evolved into DES.

Widely adopted as a standard including by ANSI and American Bankers association

Used in ATM machines

Replaced (by AES) in 2001.
DES Parameters

Key Length $k = 56$

Block length $\ell = 64$

So,

$\text{DES}: \{0, 1\}^{56} \times \{0, 1\}^{64} \rightarrow \{0, 1\}^{64}$

$\text{DES}^{-1}: \{0, 1\}^{56} \times \{0, 1\}^{64} \rightarrow \{0, 1\}^{64}$
function DES\(_K(M)\) // \(|K| = 56\) and \(|M| = 64\)
\((K_1, \ldots, K_{16}) \leftarrow \text{KeySchedule}(K)\) // \(|K_i| = 48\) for \(1 \leq i \leq 16\)

\(M \leftarrow \text{IP}(M)\)

Parse \(M\) as \(L_0 \parallel R_0\) // \(|L_0| = |R_0| = 32\)

for \(i = 1\) to \(16\) do

\(L_i \leftarrow R_{i-1}\); \(R_i \leftarrow f(K_i, R_{i-1}) \oplus L_{i-1}\)

\(C \leftarrow \text{IP}^{-1}(L_{16} \parallel R_{16})\)

return \(C\)

Round \(i\):

Feistel round

Invertible given \(K_i\):

\(f\)
Key-Recovery Security

Let $E: \{0, 1\}^k \times \{0, 1\}^\ell \to \{0, 1\}^\ell$ be a blockcipher. It is known to the adversary $A$.

**Def:** We say that $K' \in \{0, 1\}^k$ is consistent with $(M_1, C_1), \ldots, (M_q, C_q)$ if $E(K', M_i) = C_i$ for all $1 \leq i \leq q$. 
The Game

Let $E: \{0, 1\}^k \times \{0, 1\}^\ell \rightarrow \{0, 1\}^\ell$ be a blockcipher and $A$ an adversary.

**Game $KR_E$**

- **Initialize**
  
  $K \leftarrow \{0, 1\}^k; \ i \leftarrow 0$

- **Fn($M$)**
  
  $i \leftarrow i + 1; \ M_i \leftarrow M$
  
  $C_i \leftarrow E(K, M_i)$

- **Finalize($K'$)**
  
  $\text{win} \leftarrow \text{true}$
  
  For $j = 1, \ldots, i$ do
  
  - If $E(K', M_j) \neq C_j$ then $\text{win} \leftarrow \text{false}$
  
  If $M_j \in \{M_1, \ldots, M_{j-1}\}$ then $\text{win} \leftarrow \text{false}$

Return $\text{win}$

**Adv$_E^{\text{kr}}(A)$**

$\text{Adv}_E^{\text{kr}}(A) \overset{\text{def}}{=} \Pr[KR_E^A \Rightarrow \text{true}]$

$\overset{=}{=} \Pr[\text{Adversary returns a consistent key}]

\text{Output of the game is the output of } \text{Finalize.}
Exhaustive Key Search

Let $T_1, \ldots, T_{2^k}$ be a list of all $k$ bit keys and let $\langle i \rangle$ denote the $\ell$-bit binary representation of integer $i$. Let $1 \leq q \leq 2^\ell$ be a parameter.

We define a $q$-query EKS adversary as follows:

**Adversary EKS$_q$:**

- **Running-time:** $q + 2^k \cdot T_E$
  where $T_E$ is the time to execute $E$

  \[
  \text{For } i = 1 \text{ to } q \text{ do:} \quad C_i \leftarrow F_n (\langle i \rangle)
  \]

  \[
  \text{For } j = 1 \text{ to } 2^k \text{ do:} \quad \text{If } E_{T_j}(\langle i \rangle) = C_i \forall i \in [q] \text{ return } T_j
  \]
Exhaustive Key-Search on DES

DES can be computed at 1.6 Gbits/sec in hardware.

DES plaintext = 64 bits

Chip can perform \((1.6 \times 10^9)/64 = 2.5 \times 10^7\) DES computations per second

Expect \(A_{eks} (q = 1)\) to succeed in \(2^{55}\) DES computations, so it takes time

\[
\frac{2^{55}}{2.5 \times 10^7} \approx 1.4 \times 10^9 \text{ seconds} \\
\approx 45 \text{ years!}
\]

Key Complementation \(\Rightarrow 22.5\) years

But this is prohibitive. Does this mean DES is secure?
Observation: The $E$ computations can be performed in parallel!

In 1993, Wiener designed a dedicated DES-cracking machine:

- $1$ million
- $57$ chips, each with many, many DES processors
- Finds key in 3.5 hours
Increasing Key-Length

Can one use DES to design a new blockcipher with longer effective key-length?
2DES

Block cipher $2DES : \{0, 1\}^{112} \times \{0, 1\}^{64} \to \{0, 1\}^{64}$ is defined by

$$2DES_{K_1 K_2}(M) = DES_{K_2}(DES_{K_1}(M))$$
Meet-in-the-Middle Attack

\[ y \leftarrow F_n(0^{64}) \]

**ETCS:** Try all possible \(2^{12} \) keys.
(i.e. for each key \( k \) check if \( 2 \text{DES}_K(0^{64}) = y \).

Make a list of values \( \text{DES}_K(0^{64}) \)
for all possible \( k \in \{0,1\}^{56} \).

Make another list of values \( \text{DES}_K^{-1}(y) \)
for all possible \( k \in \{0,1\}^{56} \).
3DES

Block ciphers

3DES3 : \( \{0, 1\}^{168} \times \{0, 1\}^{64} \rightarrow \{0, 1\}^{64} \)
3DES2 : \( \{0, 1\}^{112} \times \{0, 1\}^{64} \rightarrow \{0, 1\}^{64} \)

are defined by

\[
3DES3_{K_1 \mid K_2 \mid K_3}(M) = DES_{K_3}(DES^{-1}_{K_2}(DES^{-1}_{K_1}(M)))
\]

\[
3DES2_{K_1 \mid K_2}(M) = DES_{K_2}(DES^{-1}_{K_1}(DES_{K_2}(M)))
\]

Meet-in-the-middle attack on 3DES3 reduces its “effective” key length to 112.
DESX

\[ DESX_{KK_1K_2}(M) = K_2 \oplus DES_K(K_1 \oplus M) \]

- Key length = 56 + 64 + 64 = 184
- “Effective” key length = 120 due to a \(2^{120}\) time meet-in-middle attack
Increasing Block-Length?

We will later see that we would also like a blockcipher with longer block-length.
Increasing Block-Length?

We will later see that we would also like a blockcipher with longer block-length.
Increasing Block-Length?

We will later see that we would also like a blockcipher with longer block-length.

This seems much harder to do using DES.
Increasing Block-Length?

We will later see that we would also like a blockcipher with longer block-length.

This seems much harder to do using DES.
Increasing Block-Length?

We will later see that we would also like a blockcipher with longer block-length.

This seems much harder to do using DES.

Motivated the search for a new blockcipher.
AES History

1998: NIST announces competition for a new block cipher

- key length 128
- block length 128
- faster than DES in software

Submissions from all over the world: MARS, Rijndael, Two-Fish, RC6, Serpent, Loki97, Cast-256, Frog, DFC, Magenta, E2, Crypton, HPC, Safer+, Deal

2001: NIST selects Rijndael to be AES.
AES Construction

function AES_\(K(M)\)
\((K_0, \ldots, K_{10}) \leftarrow \text{expand}(K)\)
\(s \leftarrow M \oplus K_0\)
for \(r = 1\) to 10 do
    \(s \leftarrow S(s)\)
    \(s \leftarrow \text{shift-cols}(s)\)
    if \(r \leq 9\) then \(s \leftarrow \text{mix-cols}(s)\) fi
    \(s \leftarrow s \oplus K_r\)
end for
return \(s\)

- Fewer tables than DES
- Finite field operations
Is Key-Recovery Security Enough?