CS5760: Cryptanalysis of DES and DES-like Iterated Cryptosystems

Gautam Singh

Indian Institute of Technology Hyderabad

February 3, 2025

Differential Cryptanalysis

1. Chosen plaintext attack.
2. Exploit XOR between plaintext pairs to find key bits.

Differential Cryptanalysis

1. Chosen plaintext attack.
2. Exploit XOR between plaintext pairs to find key bits.
3. Per DES round, XOR of respective inputs is:
   - *Linear* in expansion E to get S_E.
   - *Invariant* in key mixing with subkey S_K to get $S_I = S_E \oplus S_K$.
   - *Linear* in permutation P on S_O after S boxes.
   - *Invariant* in XOR operation connecting rounds.

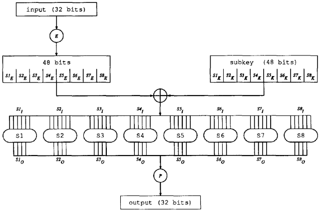

Figure 1: F function of DES.

Differential Cryptanalysis

1. Chosen plaintext attack.
2. Exploit XOR between plaintext pairs to find key bits.
3. Per DES round, XOR of respective inputs is:
    - *Linear* in expansion E to get S_E.
    - *Invariant* in key mixing with subkey S_K to get $S_I = S_E \oplus S_K$.
    - *Linear* in permutation P on S_O after S boxes.
    - *Invariant* in XOR operation connecting rounds.
4. S boxes are *nonlinear*. Probability analysis performed between input and output XOR.

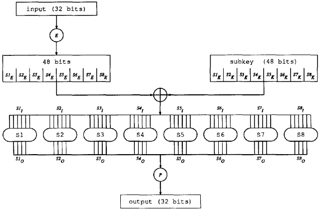

Figure 1: F function of DES.

Probability Analysis of S Boxes

1. Suppose $Si'_I = Si_I \oplus Si^*_I$ is the input XOR to the i^{th} S box, and Si'_O is the output XOR $(1 \leq i \leq 8)$.

Probability Analysis of S Boxes

1. Suppose $Si_I' = Si_I \oplus Si_I^*$ is the input XOR to the i^{th} S box, and Si_O' is the output XOR $(1 \leq i \leq 8)$.
2. We create a *pairs XOR distribution table* for each S box.
   - Each entry (Si_I', Si_O') equals the number of 6-bit key blocks Si_K for which $Si_I' \to Si_O'$.
   - 64-by-16 joint probability mass function.

Probability Analysis of S Boxes

1. Suppose $Si'_I = Si_I \oplus Si^*_I$ is the input XOR to the i^{th} S box, and Si'_O is the output XOR $(1 \le i \le 8)$.
2. We create a *pairs XOR distribution table* for each S box.
   - Each entry (Si'_I, Si'_O) equals the number of 6-bit key blocks Si_K for which $Si'_I \to Si'_O$.
   - 64-by-16 joint probability mass function.
3. This joint PMF can reduce the number of possible (sub)keys. Used to drive choice for the plaintext XOR.
   - $\approx 80\%$ entries are non-zero/possible for each S box (some have lesser percentages).
   - Given Si'_I and Si'_O, we can narrow down Si_K to a few possibilities.

Probability Analysis of S Boxes

1. Suppose $Si'_I = Si_I \oplus Si^*_I$ is the input XOR to the i^{th} S box, and Si'_O is the output XOR $(1 \leq i \leq 8)$.
2. We create a *pairs XOR distribution table* for each S box.
   - Each entry (Si'_I, Si'_O) equals the number of 6-bit key blocks Si_K for which $Si'_I \rightarrow Si'_O$.
   - 64-by-16 joint probability mass function.
3. This joint PMF can reduce the number of possible (sub)keys. Used to drive choice for the plaintext XOR.
   - $\approx 80\%$ entries are non-zero/possible for each S box (some have lesser percentages).
   - Given Si'_I and Si'_O, we can narrow down Si_K to a few possibilities.
4. i^{th} S box contributes probability p_i for $Si'_I \rightarrow Si'_O$.
   - For $X \rightarrow Y$ over a round, $P = \prod_i p_i$.
   - Over n rounds, $P = \prod_{i=1}^{n} P_i$.

Probability Analysis of S Boxes

1. Suppose $Si'_I = Si_I \oplus Si^*_I$ is the input XOR to the i^{th} S box, and Si'_O is the output XOR $(1 \leq i \leq 8)$.
2. We create a *pairs XOR distribution table* for each S box.
   - Each entry (Si'_I, Si'_O) equals the number of 6-bit key blocks Si_K for which $Si'_I \rightarrow Si'_O$.
   - 64-by-16 joint probability mass function.
3. This joint PMF can reduce the number of possible (sub)keys. Used to drive choice for the plaintext XOR.
   - $\approx 80\%$ entries are non-zero/possible for each S box (some have lesser percentages).
   - Given Si'_I and Si'_O, we can narrow down Si_K to a few possibilities.
4. i^{th} S box contributes probability p_i for $Si'_I \rightarrow Si'_O$.
   - For $X \rightarrow Y$ over a round, $P = \prod_i p_i$.
   - Over n rounds, $P = \prod_{i=1}^{n} P_i$.

Desirable for cryptanalysis: high P with large n.

Characteristic

Definition 1 (Characteristic)

An *n-round chracteristic* is a tuple $\Omega = (\Omega_P, \Omega_\Lambda, \Omega_T)$ where $\Omega_P = (L', R')$ and $\Omega_T = (l', r')$ are m bit numbers, $\Omega_\Lambda = (\Lambda_1, \dots, \Lambda_n)$, $\Lambda_i = (\lambda_I^i, \lambda_O^i)$ and $\lambda_I^i, \lambda_O^i, L', R', l', r'$ are $\frac{m}{2}$ bit numbers and m is the block size of the cryptosystem satisfying

$$\lambda_I^1 = R' \tag{1}$$

$$\lambda_I^2 = L' \oplus \lambda_O^1 \tag{2}$$

$$\lambda_I^n = r' \tag{3}$$

$$\lambda_I^{n-1} = l' \oplus \lambda_O^n \tag{4}$$

$$\forall\ 1 < i < n,\ \lambda_O^i = \lambda_I^{i-1} \oplus \lambda_I^{i+1} \tag{5}$$

Characteristic

Definition 2 (Right Pair)

A *right pair with respect to an n-round characteristic* $\Omega = (\Omega_P, \Omega_\Lambda, \Omega_T)$ *and an independent key* K is a pair for which $P' = \Omega_P$ and for each round i of the first n rounds of the encryption of the pair using K the input XOR of the i^{th} round equals λ_I^i and the output XOR of the F function equals λ_O^i. Pairs that do not satisfy these conditions are called *wrong pairs*.

Characteristic

Definition 2 (Right Pair)

A *right pair with respect to an n-round characteristic* $\Omega = (\Omega_P, \Omega_\Lambda, \Omega_T)$ *and an independent key* K is a pair for which $P' = \Omega_P$ and for each round i of the first n rounds of the encryption of the pair using K the input XOR of the i^{th} round equals λ_I^i and the output XOR of the F function equals λ_O^i. Pairs that do not satisfy these conditions are called *wrong pairs*.

Definition 3 (Probability of a Round of a Characteristic)

Round i of an n-round characteristic Ω has probability p_i^Ω if $\lambda_I^i \to \lambda_O^i$ with probability p_i^Ω by the F function.

Probability of a Characteristic

Definition 4 (Probability of a Characteristic)

An n-round characteristic Ω has probability p^{Ω} given by

$$p^{\Omega} = \prod_{i=1}^{n} p_i^{\Omega} \tag{6}$$

Probability of a Characteristic

Definition 4 (Probability of a Characteristic)

An n-round characteristic Ω has probability p^{Ω} given by

$$p^{\Omega} = \prod_{i=1}^{n} p_i^{\Omega} \tag{6}$$

Theorem 5 (Probability of a Characteristic and Right Pairs)

The formally defined probability of a characteristic $\Omega = (\Omega_P, \Omega_\Lambda, \Omega_T)$ is the probability that any fixed plaintext pair satisfying $P' = \Omega_P$ is a right pair when random independent keys are used.

Example of a Characteristic

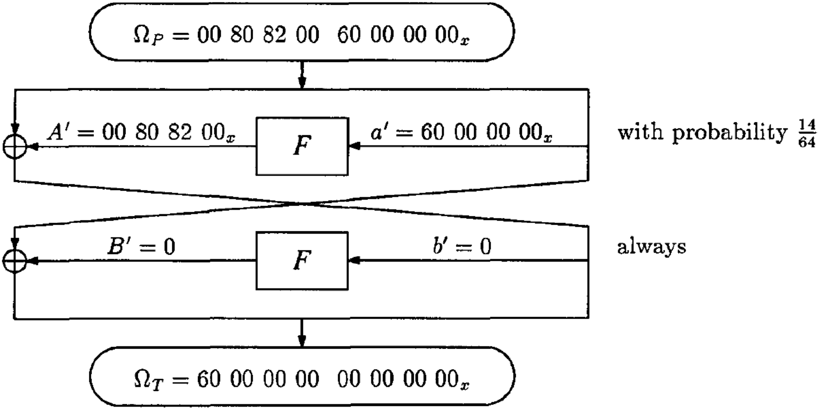

Figure 2: Example of a two-round characteristic with probability $\frac{14}{64}$.

Signal to Noise Ratio

1. Right pairs will always suggest the right key value. But right pairs occur with probability p^{Ω}.

Signal to Noise Ratio

1. Right pairs will always suggest the right key value. But right pairs occur with probability p^{Ω}.
2. On the other hand, wrong pairs suggest a randomly chosen key (not necessarily the right key in the worst case).

Signal to Noise Ratio

1. Right pairs will always suggest the right key value. But right pairs occur with probability p^{Ω}.
2. On the other hand, wrong pairs suggest a randomly chosen key (not necessarily the right key in the worst case).
3. Suitable counting approach on the key values will "spike" at the right key and have smaller but approximately equal counts at other keys.

Signal to Noise Ratio

1. Right pairs will always suggest the right key value. But right pairs occur with probability p^{Ω}.
2. On the other hand, wrong pairs suggest a randomly chosen key (not necessarily the right key in the worst case).
3. Suitable counting approach on the key values will "spike" at the right key and have smaller but approximately equal counts at other keys.
4. The key with the largest count is likely the actual key.

Signal to Noise Ratio

1. Right pairs will always suggest the right key value. But right pairs occur with probability p^{Ω}.
2. On the other hand, wrong pairs suggest a randomly chosen key (not necessarily the right key in the worst case).
3. Suitable counting approach on the key values will "spike" at the right key and have smaller but approximately equal counts at other keys.
4. The key with the largest count is likely the actual key.

Definition 6 (Signal-to-Noise Ratio)

The ratio between the number of right pairs and the average count of incorrect subkeys in a counting scheme is called the *signal to noise ratio of the counting scheme* and is denoted by S/N.

Computing the SNR

Consider the variables shown in Table 1.

Variable	Definition
p	Probability of the characteristic
m	Number of created pairs
α	Average count per analyzed pair
β	Fraction of analyzed pairs
k	Number of key bits counted on

Table 1: Table of variables to compute the SNR.

Computing the SNR

Consider the variables shown in Table 1.

Variable	Definition
p	Probability of the characteristic
m	Number of created pairs
α	Average count per analyzed pair
β	Fraction of analyzed pairs
k	Number of key bits counted on

Table 1: Table of variables to compute the SNR.

Then,

$$S/N = \frac{m \cdot p}{\frac{m \cdot \beta \cdot \alpha}{2^k}} = \frac{2^k \cdot p}{\alpha \cdot \beta} \tag{7}$$

Structures

1. Many attacks on DES use more than one characteristic.

Structures

1. Many attacks on DES use more than one characteristic.
2. Requirement to minimize the amount of plaintexts generated.

Structures

1. Many attacks on DES use more than one characteristic.
2. Requirement to minimize the amount of plaintexts generated.

Definition 7 (Quartet and Octet)

A *quartet* is a structure of four ciphertexts that simultaneously contains two ciphertext pairs of one characteristic and two ciphertext pairs of a second characteristic. An *octet* is a structure of eight ciphertexts that simultaneously contains four ciphertext pairs of each of three characteristics.

3. As an example, $(P, P \oplus \Omega_P^1, P \oplus \Omega_P^2, P \oplus \Omega_P^1 \oplus \Omega_P^2)$ is a quartet.

Structures

1. Many attacks on DES use more than one characteristic.
2. Requirement to minimize the amount of plaintexts generated.

Definition 7 (Quartet and Octet)

A *quartet* is a structure of four ciphertexts that simultaneously contains two ciphertext pairs of one characteristic and two ciphertext pairs of a second characteristic. An *octet* is a structure of eight ciphertexts that simultaneously contains four ciphertext pairs of each of three characteristics.

3. As an example, $(P, P \oplus \Omega_P^1, P \oplus \Omega_P^2, P \oplus \Omega_P^1 \oplus \Omega_P^2)$ is a quartet.
4. Quartets save $\frac{1}{2}$ of the data and octets save $\frac{2}{3}$ of the data.

DES Reduced to Four Rounds

1. Use two one-round characteristics, as shown in Figure 3.

DES Reduced to Four Rounds

1. Use two one-round characteristics, as shown in Figure 3.
2. Both characteristics have probability 1.

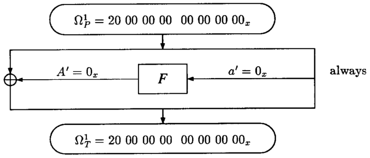

Figure 3: Characteristics used for cryptanalysis of DES reduced to four rounds.

DES Reduced to Four Rounds

1. Use two one-round characteristics, as shown in Figure 3.
2. Both characteristics have probability 1.
3. Example of a *3R-attack*. There are *three* extra rounds after the characteristic is applied.

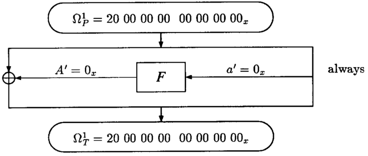

Figure 3: Characteristics used for cryptanalysis of DES reduced to four rounds.

DES Reduced to Four Rounds

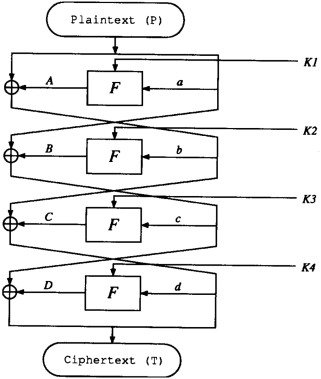

Figure 4: DES reduced to four rounds.

DES Reduced to Four Rounds

1. Using Ω^1, we have

$$c' = D' \oplus l' = a' \oplus B' \implies D' = B' \oplus l' \quad (8)$$

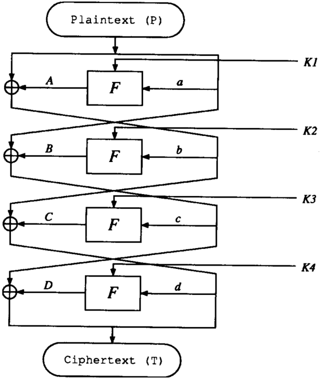

Figure 4: DES reduced to four rounds.

DES Reduced to Four Rounds

1. Using Ω^1, we have

$$c' = D' \oplus l' = a' \oplus B' \implies D' = B' \oplus l' \quad (8)$$

2. We have $a' = 0_x \implies A' = 0_x$ and $b' = A' \oplus L' = L'$.

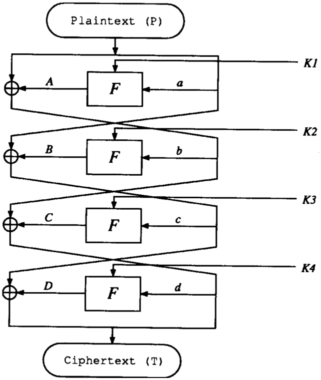

Figure 4: DES reduced to four rounds.

DES Reduced to Four Rounds

1. Using Ω^1, we have

$$c' = D' \oplus l' = a' \oplus B' \implies D' = B' \oplus l' \quad (8)$$

2. We have $a' = 0_x \implies A' = 0_x$ and $b' = A' \oplus L' = L'$.
   - In the second round S2, ..., S8 receive zero XOR input.

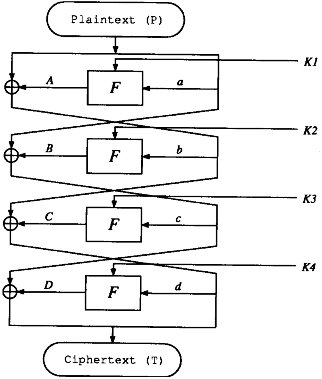

Figure 4: DES reduced to four rounds.

DES Reduced to Four Rounds

1. Using Ω^1, we have

$$c' = D' \oplus l' = a' \oplus B' \implies D' = B' \oplus l' \quad (8)$$

2. We have $a' = 0_x \implies A' = 0_x$ and $b' = A' \oplus L' = L'$.
   - In the second round S2, ..., S8 receive zero XOR input.
   - 28 bits of B' are zero and hence we can find *28 bits of* D'.

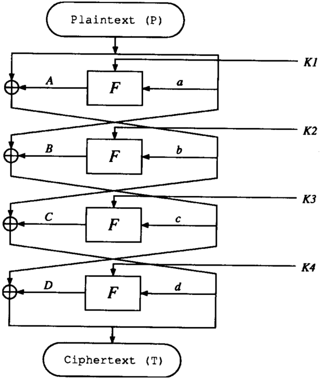

Figure 4: DES reduced to four rounds.

DES Reduced to Four Rounds

1. Using Ω^1, we have

$$c' = D' \oplus l' = a' \oplus B' \implies D' = B' \oplus l' \quad (8)$$

2. We have $a' = 0_x \implies A' = 0_x$ and $b' = A' \oplus L' = L'$.
   - In the second round S2, ..., S8 receive zero XOR input.
   - 28 bits of B' are zero and hence we can find *28 bits of D'*.
   - We already know $d' = r'$. So, we employ a counting approach to get $K4$.

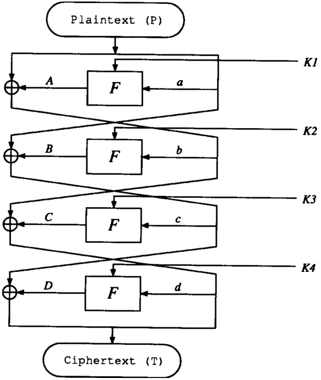

Figure 4: DES reduced to four rounds.

DES Reduced to Four Rounds

1. To get Si_{Kd} for $2 \leq i \leq 8$, we verify (9).

$$S(S_E \oplus S_K) \oplus S(S_E^* \oplus S_K) = S_O' \tag{9}$$

2. Only *one* plaintext pair is needed since characteristic probability is 1.
3. We recover $7 \times 6 = 42$ key bits of $K4$, which correspond to 42 bits of the master key.
4. Exhaustively search the other 14 key bits to get the entire master key.
5. We have used the key schedule to our advantage here? *What if all the keys were independent?*

DES Reduced to Four Rounds: Independent Subkeys

1. We now use Ω^2 to get the remaining 6 subkey bits of $K4$, as the input to S1 in the second round is now zero.

DES Reduced to Four Rounds: Independent Subkeys

1. We now use Ω^2 to get the remaining 6 subkey bits of $K4$, as the input to S1 in the second round is now zero.
2. We have $C' = b' \oplus d'$. Peeling off/decrypting one round will give us c' completely.

DES Reduced to Four Rounds: Independent Subkeys

1. We now use Ω^2 to get the remaining 6 subkey bits of $K4$, as the input to S1 in the second round is now zero.
2. We have $C' = b' \oplus d'$. Peeling off/decrypting one round will give us c' completely.
    - Since c' and C' are both completely known, $K3$ can be completely found using a similar counting argument.

DES Reduced to Four Rounds: Independent Subkeys

1. We now use Ω^2 to get the remaining 6 subkey bits of $K4$, as the input to S1 in the second round is now zero.
2. We have $C' = b' \oplus d'$. Peeling off/decrypting one round will give us c' completely.
   - Since c' and C' are both completely known, $K3$ can be completely found using a similar counting argument.
3. Since $a' = A' = 0_x$, all keys are equally likely. Other characteristics Ω^3 and Ω^4 are chosen such that
   - $S'_{Ea} \neq 0_x$ for all S boxes for both characteristics.
   - For every S box, the S'_{Ea} values differ between the characteristics.
   - Similar counting methods used to get $K1$ and $K2$.

DES Reduced to Four Rounds: Independent Subkeys

1. We now use Ω^2 to get the remaining 6 subkey bits of $K4$, as the input to S1 in the second round is now zero.
2. We have $C' = b' \oplus d'$. Peeling off/decrypting one round will give us c' completely.
   - Since c' and C' are both completely known, $K3$ can be completely found using a similar counting argument.
3. Since $a' = A' = 0_x$, all keys are equally likely. Other characteristics Ω^3 and Ω^4 are chosen such that
   - $S'_{Ea} \neq 0_x$ for all S boxes for both characteristics.
   - For every S box, the S'_{Ea} values differ between the characteristics.
   - Similar counting methods used to get $K1$ and $K2$.
4. 16 chosen plaintexts are needed for this attack.
   - 8 pairs of Ω^1 and Ω^2 each.
   - 4 pairs of Ω^3 and Ω^4 each.

To reduce the data needed, two octets are used.

DES Reduced to Six Rounds

1. Two three-round characteristics used, each with probability $\frac{1}{16}$.

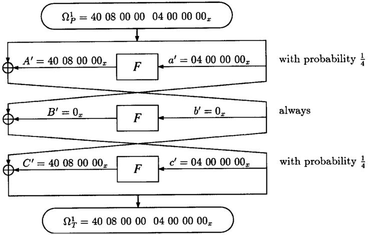

Figure 5: Characteristics used for cryptanalysis of DES reduced to 6 rounds.

DES Reduced to Six Rounds

1. Two three-round characteristics used, each with probability $\frac{1}{16}$.
2. We have,

$$e' = c' \oplus D' = F' \oplus l' \implies F' = c' \oplus D' \oplus l' \tag{10}$$

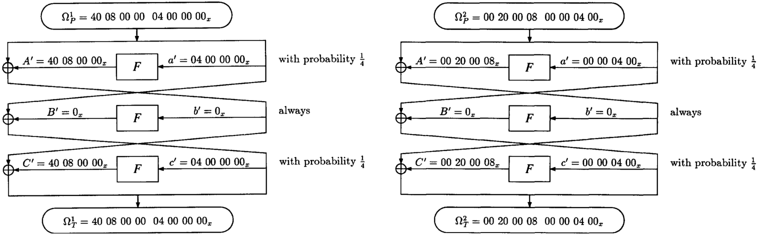

Figure 5: Characteristics used for cryptanalysis of DES reduced to 6 rounds.

DES Reduced to Six Rounds

1. In the fourth round,
    - with Ω^1, S2, S5, ..., S8 have zero input XORs.
    - with Ω^2, S1, S2, S4, S5 and S6 have zero input XORs.

DES Reduced to Six Rounds

1. In the fourth round,
   - with Ω^1, S2, S5, . . . , S8 have zero input XORs.
   - with Ω^2, S1, S2, S4, S5 and S6 have zero input XORs.
2. Combining both characteristics, 42 key bits of $K6$ can be found.

DES Reduced to Six Rounds

1. In the fourth round,
   - with Ω^1, S2, S5, ..., S8 have zero input XORs.
   - with Ω^2, S1, S2, S4, S5 and S6 have zero input XORs.
2. Combining both characteristics, 42 key bits of $K6$ can be found.
3. Counting on more bits gives high S/N at the cost of exponentially more memory.

DES Reduced to Six Rounds

1. In the fourth round,
    - with Ω^1, S2, S5, . . . , S8 have zero input XORs.
    - with Ω^2, S1, S2, S4, S5 and S6 have zero input XORs.
2. Combining both characteristics, 42 key bits of $K6$ can be found.
3. Counting on more bits gives high S/N at the cost of exponentially more memory.
4. Due to higher S/N, fewer plaintext pairs are analyzed. *This is exploited to get a faster counting algorithm*.

The Clique Method

1. Used to reduce memory when few plaintexts are used to count on more subkey bits.

The Clique Method

1. Used to reduce memory when few plaintexts are used to count on more subkey bits.
2. Create a graph where
   - Each plaintext pair is a vertex.
   - There is an edge between two vertices if corresponding pairs suggest the same key value for an S box.

The Clique Method

1. Used to reduce memory when few plaintexts are used to count on more subkey bits.
2. Create a graph where
   - Each plaintext pair is a vertex.
   - There is an edge between two vertices if corresponding pairs suggest the same key value for an S box.
3. The edges are labelled with five 64-bit masks (one mask per S box, one bit per suggested key value in the mask).
   - A pair suggests a key value if it passes the check in (9).

The Clique Method

1. Used to reduce memory when few plaintexts are used to count on more subkey bits.
2. Create a graph where
   - Each plaintext pair is a vertex.
   - There is an edge between two vertices if corresponding pairs suggest the same key value for an S box.
3. The edges are labelled with five 64-bit masks (one mask per S box, one bit per suggested key value in the mask).
   - A pair suggests a key value if it passes the check in (9).
4. Goal is to find the largest clique such that the bitwise AND of all masks in the subgraph induced by that clique is nonzero.

The Clique Method

1. Used to reduce memory when few plaintexts are used to count on more subkey bits.
2. Create a graph where
   - Each plaintext pair is a vertex.
   - There is an edge between two vertices if corresponding pairs suggest the same key value for an S box.
3. The edges are labelled with five 64-bit masks (one mask per S box, one bit per suggested key value in the mask).
   - A pair suggests a key value if it passes the check in (9).
4. Goal is to find the largest clique such that the bitwise AND of all masks in the subgraph induced by that clique is nonzero.
5. Apply this method for both Ω^1 and Ω^2, ensuring that the suggested keys at S2, S5 and S6 match. Otherwise, use more data.

Completing the Cryptanalysis

1. 42 key bits have been found, can exhaustively search remanining 14 bits.

Into S box number	e bits S_{Ee}	Key bits S_{Ke}
S1	+ + + + + +	**3** + . . + +
S2	+ + **3** + + +	+ **3** + **3** **3** **3**
S3	+ + + + + +	+ + + + + +
S4	+ + + + **3** +	+ + . . + +
S5	**3** + + + + +	+ + + . + +
S6	+ + + + **3** +	+ . + . + +
S7	**3** + + + + +	+ + + . + +
S8	+ + **3** + + +	+ + + + + +

Figure 6: Dependence of $K5$ on $K6$. '3' indicates dependence on $S3_{Kf}$, '.' indicates bits unused in K6 and '+' indicates dependence on known key bits of K6.

Completing the Cryptanalysis

1. 42 key bits have been found, can exhaustively search remaining 14 bits.
2. Speed up the search by finding remaining 6 key bits of $K6$ using Figure 6. Count using checks on S2, S3 and S8 of the fifth round.

Into S box number	e bits S_{Ee}	Key bits S_{Ke}
S1	+ + + + + +	**3** + . . + +
S2	+ + **3** + + +	+ **3** + **3** **3** **3**
S3	+ + + + + +	+ + + + + +
S4	+ + + + **3** +	+ + . . + +
S5	**3** + + + + +	+ + + . + +
S6	+ + + + **3** +	+ . + . + +
S7	**3** + + + + +	+ + + . + +
S8	+ + **3** + + +	+ + + + + +

Figure 6: Dependence of $K5$ on $K6$. ‘3’ indicates dependence on $S3_{Kf}$, ‘.’ indicates bits unused in K6 and ‘+’ indicates dependence on known key bits of K6.

Completing the Cryptanalysis

1. 42 key bits have been found, can exhaustively search remaining 14 bits.
2. Speed up the search by finding remaining 6 key bits of $K6$ using Figure 6. Count using checks on S2, S3 and S8 of the fifth round.
   - Exhaustively search remaining 8 bits.

Into S box number	e bits S_{Ee}	Key bits S_{Ke}
S1	+ + + + + +	**3** + . . + +
S2	+ + **3** + + +	+ **3** + **3** **3** **3**
S3	+ + + + + +	+ + + + + +
S4	+ + + + **3** +	+ + . . + +
S5	**3** + + + + +	+ + + . + +
S6	+ + + + **3** +	+ . + . + +
S7	**3** + + + + +	+ + + . + +
S8	+ + **3** + + +	+ + + + + +

Figure 6: Dependence of $K5$ on $K6$. '3' indicates dependence on $S3_{Kf}$, '.' indicates bits unused in K6 and '+' indicates dependence on known key bits of K6.

Completing the Cryptanalysis

1. 42 key bits have been found, can exhaustively search remaining 14 bits.
2. Speed up the search by finding remaining 6 key bits of $K6$ using Figure 6. Count using checks on S2, S3 and S8 of the fifth round.
   - Exhaustively search remaining 8 bits.
   - Discard wrong pairs by checking if they satisfy the characteristic and expected value of F'.

Into S box number	e bits S_{Ee}	Key bits S_{Ke}
S1	+ + + + + +	**3** + . . + +
S2	+ + **3** + + +	+ **3** + **3** **3** **3**
S3	+ + + + + +	+ + + + + +
S4	+ + + + **3** +	+ + . . + +
S5	**3** + + + + +	+ + + . + +
S6	+ + + + **3** +	+ . + . + +
S7	**3** + + + + +	+ + + . + +
S8	+ + **3** + + +	+ + + + + +

Figure 6: Dependence of $K5$ on $K6$. '3' indicates dependence on $S3_{Kf}$, '.' indicates bits unused in K6 and '+' indicates dependence on known key bits of K6.

Completing the Cryptanalysis

1. 42 key bits have been found, can exhaustively search remaining 14 bits.
2. Speed up the search by finding remaining 6 key bits of $K6$ using Figure 6. Count using checks on S2, S3 and S8 of the fifth round.
   - Exhaustively search remaining 8 bits.
   - Discard wrong pairs by checking if they satisfy the characteristic and expected value of F'.
   - Leaves $\frac{1}{16}$ of the pairs, boosts S/N.

Into S box number	e bits S_{Ee}	Key bits S_{Ke}
S1	+ + + + + +	3 + . . + +
S2	+ + 3 + + +	+ 3 + 3 3 3
S3	+ + + + + +	+ + + + + +
S4	+ + + + 3 +	+ + . . + +
S5	3 + + + + +	+ + + . + +
S6	+ + + + 3 +	+ . + . + +
S7	3 + + + + +	+ + + . + +
S8	+ + 3 + + +	+ + + + + +

Figure 6: Dependence of $K5$ on $K6$. '3' indicates dependence on $S3_{Kf}$, '.' indicates bits unused in K6 and '+' indicates dependence on known key bits of K6.

Data Requirements

1. The first phase has

$$S/N = \frac{2^{30} \cdot \frac{1}{16}}{4^5} = 2^{16}. \tag{11}$$

Only 7-8 pairs are needed for each characteristic. Since each characteristic has probability $\frac{1}{16}$, we require about 120 pairs of plaintexts.

Data Requirements

1. The first phase has

$$S/N = \frac{2^{30} \cdot \frac{1}{16}}{4^5} = 2^{16}. \tag{11}$$

Only 7-8 pairs are needed for each characteristic. Since each characteristic has probability $\frac{1}{16}$, we require about 120 pairs of plaintexts.

2. The second phase has

$$S/N = \frac{2^6 \cdot 1}{4} = 16. \tag{12}$$

Though S/N is lesser, we can use the 7-8 right pairs from the first part.

Data Requirements

1. The first phase has

$$S/N = \frac{2^{30} \cdot \frac{1}{16}}{4^5} = 2^{16}. \tag{11}$$

Only 7-8 pairs are needed for each characteristic. Since each characteristic has probability $\frac{1}{16}$, we require about 120 pairs of plaintexts.

2. The second phase has

$$S/N = \frac{2^6 \cdot 1}{4} = 16. \tag{12}$$

Though S/N is lesser, we can use the 7-8 right pairs from the first part.

3. We can reduce the data required by using quartets. In total, about 240 ciphertexts are needed.

DES Reduced to Eight Rounds

1. We use a 5-round characteristic with probability $\approx \frac{1}{10486}$.
2. From Figure 7, a right pair has $f' = d' \oplus E' = 40\ 5C\ 00\ 00_x$.
    - In the sixth round, S2, S5, ..., S8 have zero input XORs.
3. We have,

$$g' = e' \oplus F' = H' \oplus l' \tag{13}$$

$$\implies H' = e' \oplus F' \oplus l'. \tag{14}$$

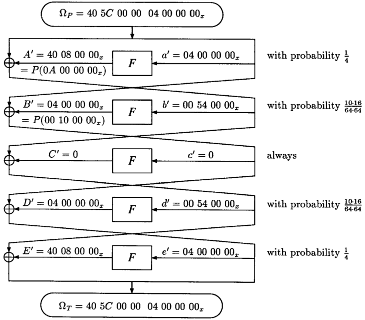

Figure 7: 5 round characteristic to cryptanalyze DES reduced to 8 rounds.

Improving the Signal to Noise Ratio

1. Signal to noise ratio for
   - $k = 30$ is $S/N = \frac{2^{30}}{4^5 \cdot 10486} \approx 100$. Requires 2^{30} counters.
   - $k = 24$ is $S/N = \frac{2^{24}}{4^4 \cdot 0.8 \cdot 10486} \approx 7.8$. Requires 2^{24} counters.
   - $k = 18$ is $S/N = \frac{2^{18}}{4^3 \cdot 0.8^2 \cdot 10486} \approx 0.6$. Requires 2^{18} counters.

Improving the Signal to Noise Ratio

1. Signal to noise ratio for
   - $k = 30$ is $S/N = \frac{2^{30}}{4^5 \cdot 10486} \approx 100$. Requires 2^{30} counters.
   - $k = 24$ is $S/N = \frac{2^{24}}{4^4 \cdot 0.8 \cdot 10486} \approx 7.8$. Requires 2^{24} counters.
   - $k = 18$ is $S/N = \frac{2^{18}}{4^3 \cdot 0.8^2 \cdot 10486} \approx 0.6$. Requires 2^{18} counters.

   Why the 0.8 in the denominator?

Improving the Signal to Noise Ratio

1. Signal to noise ratio for
   - $k = 30$ is $S/N = \frac{2^{30}}{4^5 \cdot 10486} \approx 100$. Requires 2^{30} counters.
   - $k = 24$ is $S/N = \frac{2^{24}}{4^4 \cdot 0.8 \cdot 10486} \approx 7.8$. Requires 2^{24} counters.
   - $k = 18$ is $S/N = \frac{2^{18}}{4^3 \cdot 0.8^2 \cdot 10486} \approx 0.6$. Requires 2^{18} counters.

   Why the 0.8 in the denominator?
2. If counting on fewer S boxes, choose the ones that *maximize* the characteristic probability, and check against the other S boxes.

Improving the Signal to Noise Ratio

1. Signal to noise ratio for
   - $k = 30$ is $S/N = \frac{2^{30}}{4^5 \cdot 10486} \approx 100$. Requires 2^{30} counters.
   - $k = 24$ is $S/N = \frac{2^{24}}{4^4 \cdot 0.8 \cdot 10486} \approx 7.8$. Requires 2^{24} counters.
   - $k = 18$ is $S/N = \frac{2^{18}}{4^3 \cdot 0.8^2 \cdot 10486} \approx 0.6$. Requires 2^{18} counters.

   Why the 0.8 in the denominator?
2. If counting on fewer S boxes, choose the ones that *maximize* the characteristic probability, and check against the other S boxes.
3. Notice that

$$e' = 04\ 00\ 00\ 00_x \rightarrow E' = P(0W\ 00\ 00\ 00_x) = X0\ 0Y\ Z0\ 00_x \quad (15)$$

where $W \in \{0, 1, 2, 3, 8, 9, A, B\}$, $X, Z \in \{0, 4\}$, $Y \in \{0, 8\}$.

Improving the Signal to Noise Ratio

1. Signal to noise ratio for
   - $k = 30$ is $S/N = \frac{2^{30}}{4^5 \cdot 10486} \approx 100$. Requires 2^{30} counters.
   - $k = 24$ is $S/N = \frac{2^{24}}{4^4 \cdot 0.8 \cdot 10486} \approx 7.8$. Requires 2^{24} counters.
   - $k = 18$ is $S/N = \frac{2^{18}}{4^3 \cdot 0.8^2 \cdot 10486} \approx 0.6$. Requires 2^{18} counters.

   Why the 0.8 in the denominator?
2. If counting on fewer S boxes, choose the ones that *maximize* the characteristic probability, and check against the other S boxes.
3. Notice that

$$e' = 04\ 00\ 00\ 00_x \to E' = P(0W\ 00\ 00\ 00_x) = X0\ 0Y\ Z0\ 00_x \quad (15)$$

   where $W \in \{0, 1, 2, 3, 8, 9, A, B\}$, $X, Z \in \{0, 4\}$, $Y \in \{0, 8\}$.
4. Thus, $f' = d' \oplus E' = X0\ 5V\ Z0\ 00_x$ where $V = Y \oplus 4$.
   - $Z = 0 \implies E' = 40\ 08\ 00\ 00_x$. This happens with probability $\frac{16}{64}$.
   - All other possiblities having $Z = 4$ happen with probability $\frac{20}{64}$.

Modifying the Characteristic

1. From (15), the modified probability of $e' \to E'$ is $\frac{16}{64} + 0.8\frac{20}{64} = \frac{1}{2}$.

Modifying the Characteristic

1. From (15), the modified probability of $e' \to E'$ is $\frac{16}{64} + 0.8\frac{20}{64} = \frac{1}{2}$.
2. *We have doubled the characteristic probability to* $\frac{1}{5243}$! Thus, for $k = 24, S/N \approx 15.6$ and for $k = 18, S/N \approx 1.2$.

Modifying the Characteristic

1. From (15), the modified probability of $e' \to E'$ is $\frac{16}{64} + 0.8\frac{20}{64} = \frac{1}{2}$.
2. *We have doubled the characteristic probability to* $\frac{1}{5243}$! Thus, for $k = 24, S/N \approx 15.6$ and for $k = 18, S/N \approx 1.2$.
    - Key bits of S5 NOT counted, but used for checking as in (9).

Modifying the Characteristic

1. From (15), the modified probability of $e' \to E'$ is $\frac{16}{64} + 0.8\frac{20}{64} = \frac{1}{2}$.
2. *We have doubled the characteristic probability to* $\frac{1}{5243}$! Thus, for $k = 24, S/N \approx 15.6$ and for $k = 18, S/N \approx 1.2$.
   - Key bits of S5 NOT counted, but used for checking as in (9).
3. For $k = 24$, we need 25000 pairs. For $k = 18$, we need 150000 pairs, where
   - Average count per key is $\frac{150000 \cdot 4^3 \cdot 0.8^2}{2^{18}} = 24$.
   - Right key is counted an additional $\frac{150000}{5243} = 29$ times, for a total count of $24 + 29 = 53$.

Modifying the Characteristic

1. From (15), the modified probability of $e' \to E'$ is $\frac{16}{64} + 0.8\frac{20}{64} = \frac{1}{2}$.
2. *We have doubled the characteristic probability to* $\frac{1}{5243}$! Thus, for $k = 24, S/N \approx 15.6$ and for $k = 18, S/N \approx 1.2$.
    - Key bits of S5 NOT counted, but used for checking as in (9).
3. For $k = 24$, we need 25000 pairs. For $k = 18$, we need 150000 pairs, where
    - Average count per key is $\frac{150000 \cdot 4^3 \cdot 0.8^2}{2^{18}} = 24$.
    - Right key is counted an additional $\frac{150000}{5243} = 29$ times, for a total count of $24 + 29 = 53$.
4. After counting on 18 key bits, we count on S2 and S5 of the last round using the 53 filtered pairs.
    - Almost all remaining pairs after both counting schemes should be right pairs (why?).

Modifying the Characteristic

1. From (15), the modified probability of $e' \to E'$ is $\frac{16}{64} + 0.8\frac{20}{64} = \frac{1}{2}$.
2. *We have doubled the characteristic probability to* $\frac{1}{5243}$! Thus, for $k = 24, S/N \approx 15.6$ and for $k = 18, S/N \approx 1.2$.
   - Key bits of S5 NOT counted, but used for checking as in (9).
3. For $k = 24$, we need 25000 pairs. For $k = 18$, we need 150000 pairs, where
   - Average count per key is $\frac{150000 \cdot 4^3 \cdot 0.8^2}{2^{18}} = 24$.
   - Right key is counted an additional $\frac{150000}{5243} = 29$ times, for a total count of $24 + 29 = 53$.
4. After counting on 18 key bits, we count on S2 and S5 of the last round using the 53 filtered pairs.
   - Almost all remaining pairs after both counting schemes should be right pairs (why?).
   - *Hint: What is the probability that a wrong pair survives both counting stages?*

Finding the Remaining Bits of $K8$

1. Since 20 bits of H and H^* are known we can get corresponding bits of g and g^*.

Finding the Remaining Bits of $K8$

1. Since 20 bits of H and H^* are known we can get corresponding bits of g and g^*.
2. Exhaustive search performed for the remaining 18 bits of $K8$. We know $G' = f' \oplus h'$.
   - Search for the 12 bits entering S1 and S4 by verifying $S3'_{Og}$.
   - Then exhaustively search for the other 6 bits using the relations in Figure 8.

Into S box number	g bits S_{Eg}	Key bits S_{Kg}
S1	+ 4 + + + +	3 + . . 4 +
S2	+ + 3 + + 1	1 3 4 3 3 3
S3	+ 1 4 + + +	+ 1 + 4 1 +
S4	+ + + + 3 1	1 1 . . 1 +
S5	3 1 + + 4 +	+ + + . + +
S6	4 + + 1 3 +	+ . + . + +
S7	3 + 4 + + +	+ + + . + +
S8	+ + 3 1 + 4	+ + + + + +

Figure 8: Dependence of $K7$ on $K8$.

Finding the Remaining Bits of $K8$

1. Since 20 bits of H and H^* are known we can get corresponding bits of g and g^*.
2. Exhaustive search performed for the remaining 18 bits of $K8$. We know $G' = f' \oplus h'$.
   - Search for the 12 bits entering S1 and S4 by verifying $S3'_{Og}$.
   - Then exhaustively search for the other 6 bits using the relations in Figure 8.
3. The last 8 bits can also be exhaustively searched using one ciphertext pair.

Into S box number	g bits S_{Eg}	Key bits S_{Kg}
S1	+ 4 + + + +	3 + . . 4 +
S2	+ + 3 + + 1	1 3 4 3 3 3
S3	+ 1 4 + + +	+ 1 + 4 1 +
S4	+ + + + 3 1	1 1 . . 1 +
S5	3 1 + + 4 +	+ + + . + +
S6	4 + + 1 3 +	+ . + . + +
S7	3 + 4 + + +	+ + + . + +
S8	+ + 3 1 + 4	+ + + + + +

Figure 8: Dependence of $K7$ on $K8$.

Memory Saving Techniques

1. We can discard wrong pairs as and when they are identified. Leaves $0.8^5 \approx \frac{1}{3}$ of all pairs.

Memory Saving Techniques

1. We can discard wrong pairs as and when they are identified. Leaves $0.8^5 \approx \frac{1}{3}$ of all pairs.
   - Still not enough for 18 bits (reduces to 50000 pairs).

Memory Saving Techniques

1. We can discard wrong pairs as and when they are identified. Leaves $0.8^5 \approx \frac{1}{3}$ of all pairs.
   - Still not enough for 18 bits (reduces to 50000 pairs).
2. Use a *heuristic* weighting function to discard even more wrong pairs.

Memory Saving Techniques

1. We can discard wrong pairs as and when they are identified. Leaves $0.8^5 \approx \frac{1}{3}$ of all pairs.
   - Still not enough for 18 bits (reduces to 50000 pairs).
2. Use a *heuristic* weighting function to discard even more wrong pairs.
   - Based on the idea that a right pair suggests more key values than a wrong pair.

Memory Saving Techniques

1. We can discard wrong pairs as and when they are identified. Leaves $0.8^5 \approx \frac{1}{3}$ of all pairs.
   - Still not enough for 18 bits (reduces to 50000 pairs).
2. Use a *heuristic* weighting function to discard even more wrong pairs.
   - Based on the idea that a right pair suggests more key values than a wrong pair.
   - Weighting function is product of key values suggested by the five countable S boxes of the last round.

Memory Saving Techniques

1. We can discard wrong pairs as and when they are identified. Leaves $0.8^5 \approx \frac{1}{3}$ of all pairs.
   - Still not enough for 18 bits (reduces to 50000 pairs).
2. Use a *heuristic* weighting function to discard even more wrong pairs.
   - Based on the idea that a right pair suggests more key values than a wrong pair.
   - Weighting function is product of key values suggested by the five countable S boxes of the last round.
   - Threshold chosen to discard most wrong pairs. Experimentally good threshold is 8192 which discards 97% of the wrong pairs.

Memory Saving Techniques

1. We can discard wrong pairs as and when they are identified. Leaves $0.8^5 \approx \frac{1}{3}$ of all pairs.
   - Still not enough for 18 bits (reduces to 50000 pairs).
2. Use a *heuristic* weighting function to discard even more wrong pairs.
   - Based on the idea that a right pair suggests more key values than a wrong pair.
   - Weighting function is product of key values suggested by the five countable S boxes of the last round.
   - Threshold chosen to discard most wrong pairs. Experimentally good threshold is 8192 which discards 97% of the wrong pairs.
3. The weighting function reduces number of analyzed pairs to 7500, leading to improvements in runtime.

Enhanced Characteristic's Probability

1. Use relations between input and output bits of the S boxes in the characteristic to refine choices for plaintexts.

Enhanced Characteristic's Probability

1. Use relations between input and output bits of the S boxes in the characteristic to refine choices for plaintexts.
2. Main idea:
   - Find relation between input bits for a high probability entry in the pairs XOR distribution table.
   - Find information about the key bits at those positions (this could be found earlier).
   - Choose plaintexts accordingly to boost characteristic probability and signal to noise ratio.

Extension to Nine Rounds

1. Characteristic shown in Figure 7 extended with extra round shown in Figure 9.

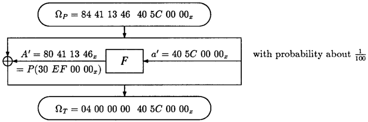

Figure 9: Extension of characteristic for cryptanalysis of DES reduced to 9 rounds.

Extension to Nine Rounds

1. Characteristic shown in Figure 7 extended with extra round shown in Figure 9.
2. Characteristic probability $\approx 10^{-6}$.
    - $S/N = \frac{2^{30}}{4^5 \cdot 10^6} \approx 1$.
    - About 30 million pairs and an array of 2^{30} counters needed.

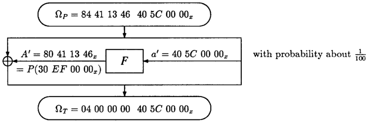

Figure 9: Extension of characteristic for cryptanalysis of DES reduced to 9 rounds.

Extension to Nine Rounds

1. Characteristic shown in Figure 7 extended with extra round shown in Figure 9.
2. Characteristic probability $\approx 10^{-6}$.
    - $S/N = \frac{2^{30}}{4^5 \cdot 10^6} \approx 1$.
    - About 30 million pairs and an array of 2^{30} counters needed.
3. This attack requires a lot of data and memory, hence it is unrealistic.

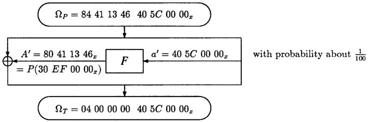

Figure 9: Extension of characteristic for cryptanalysis of DES reduced to 9 rounds.

Iterative Characteristics

1. Can concatenate with itself to create longer characteristics. Useful for arbitrary rounds.

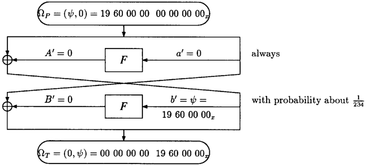

Figure 10: An iterative characteristic for DES.

Iterative Characteristics

1. Can concatenate with itself to create longer characteristics. Useful for arbitrary rounds.
2. Figure 10 shows a characteristic with optimal probability (why?).

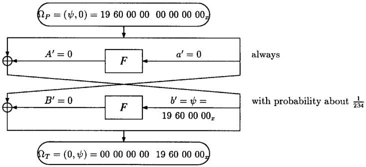

Figure 10: An iterative characteristic for DES.

Iterative Characteristics

1. Can concatenate with itself to create longer characteristics. Useful for arbitrary rounds.
2. Figure 10 shows a characteristic with optimal probability (why?).
   - Another value of $\psi = 1B\ 60\ 00\ 00_x$.

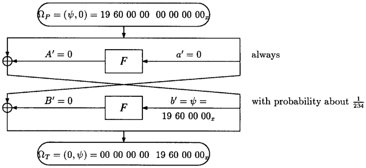

Figure 10: An iterative characteristic for DES.

Iterative Characteristics

1. Can concatenate with itself to create longer characteristics. Useful for arbitrary rounds.
2. Figure 10 shows a characteristic with optimal probability (why?).
   - Another value of $\psi = 1B\ 60\ 00\ 00_x$.
3. Add an extra round for "free" by concatenating just the first round again.

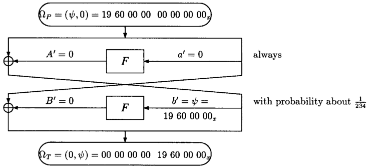

Figure 10: An iterative characteristic for DES.

Iterative Characteristics

1. Can concatenate with itself to create longer characteristics. Useful for arbitrary rounds.
2. Figure 10 shows a characteristic with optimal probability (why?).
   - Another value of $\psi = 1B\ 60\ 00\ 00_x$.
3. Add an extra round for "free" by concatenating just the first round again.
4. 15-round extension has probability 2^{-56}. *Just the iterative characteristic is not enough*!

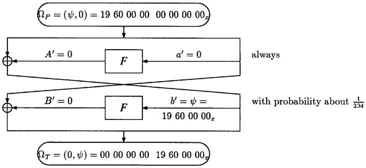

Figure 10: An iterative characteristic for DES.

3R-Attacks

1. Counting done on subkey bits of the last round that enter S boxes whose corresponding S boxes in the round which follows the last round of the characteristic have zero input XORs.

3R-Attacks

1. Counting done on subkey bits of the last round that enter S boxes whose corresponding S boxes in the round which follows the last round of the characteristic have zero input XORs.
   - In DES reduced to *four* rounds: "...zero input XORs in...the *second* round".

3R-Attacks

1. Counting done on subkey bits of the last round that enter S boxes whose corresponding S boxes in the round which follows the last round of the characteristic have zero input XORs.
   - In DES reduced to *four* rounds: "...zero input XORs in...the *second* round".
2. Not advisable for larger rounds due to small S/N.

3R-Attacks

1. Counting done on subkey bits of the last round that enter S boxes whose corresponding S boxes in the round which follows the last round of the characteristic have zero input XORs.
    - In DES reduced to *four* rounds: "...zero input XORs in...the *second* round".
2. Not advisable for larger rounds due to small S/N.
3. More powerful compared to 0R/1R/2R-attacks due to smaller characteristic length.
    - For fixed number of iterations in a cryptosystem, 3R-attacks are the most useful.

2R-Attacks

1. Count on all bits of the subkey of the last round (why?).

2R-Attacks

1. Count on all bits of the subkey of the last round (why?).
2. Wrong pairs discarded if input XORs of S boxes in the previous round may not cause expected output XORs.

2R-Attacks

1. Count on all bits of the subkey of the last round (why?).
2. Wrong pairs discarded if input XORs of S boxes in the previous round may not cause expected output XORs.
3. Example: DES reduced to 9 rounds.
   - 7-round iterative characteristic has probability $\approx 2^{-24}$.

2R-Attacks

1. Count on all bits of the subkey of the last round (why?).
2. Wrong pairs discarded if input XORs of S boxes in the previous round may not cause expected output XORs.
3. Example: DES reduced to 9 rounds.
   - 7-round iterative characteristic has probability $\approx 2^{-24}$.
   - Five S boxes in the eighth round have $S'_{Eh} = S'_{Ih} = 0$, thus $S'_{Oh} = 0$. This happens for wrong pairs with probability $\frac{1}{16}$. For other S boxes, this probability is 0.8.

2R-Attacks

1. Count on all bits of the subkey of the last round (why?).
2. Wrong pairs discarded if input XORs of S boxes in the previous round may not cause expected output XORs.
3. Example: DES reduced to 9 rounds.
   - 7-round iterative characteristic has probability $\approx 2^{-24}$.
   - Five S boxes in the eighth round have $S'_{Eh} = S'_{Ih} = 0$, thus $S'_{Oh} = 0$. This happens for wrong pairs with probability $\frac{1}{16}$. For other S boxes, this probability is 0.8.
   - Counting on 48 bits of $K9$ has $S/N = \frac{2^{48} \cdot 2^{-24}}{4^8 \cdot 0.8^3 \cdot (\frac{1}{16})^5} \approx 2^{29}$.

2R-Attacks

1. Count on all bits of the subkey of the last round (why?).
2. Wrong pairs discarded if input XORs of S boxes in the previous round may not cause expected output XORs.
3. Example: DES reduced to 9 rounds.
   - 7-round iterative characteristic has probability $\approx 2^{-24}$.
   - Five S boxes in the eighth round have $S'_{Eh} = S'_{Ih} = 0$, thus $S'_{Oh} = 0$. This happens for wrong pairs with probability $\frac{1}{16}$. For other S boxes, this probability is 0.8.
   - Counting on 48 bits of $K9$ has $S/N = \frac{2^{48} \cdot 2^{-24}}{4^8 \cdot 0.8^3 \cdot (\frac{1}{16})^5} \approx 2^{29}$.
   - Counting on 18 bits of $K9$ has $S/N = \frac{2^{18} \cdot 2^{-24}}{4^3 \cdot 0.8^5 \cdot 0.8^3 \cdot (\frac{1}{16})^5} \approx 2^{11}$.

2R-Attacks

1. Count on all bits of the subkey of the last round (why?).
2. Wrong pairs discarded if input XORs of S boxes in the previous round may not cause expected output XORs.
3. Example: DES reduced to 9 rounds.
   - 7-round iterative characteristic has probability $\approx 2^{-24}$.
   - Five S boxes in the eighth round have $S'_{Eh} = S'_{Ih} = 0$, thus $S'_{Oh} = 0$. This happens for wrong pairs with probability $\frac{1}{16}$. For other S boxes, this probability is 0.8.
   - Counting on 48 bits of $K9$ has $S/N = \frac{2^{48} \cdot 2^{-24}}{4^8 \cdot 0.8^3 \cdot (\frac{1}{16})^5} \approx 2^{29}$.
   - Counting on 18 bits of $K9$ has $S/N = \frac{2^{18} \cdot 2^{-24}}{4^3 \cdot 0.8^5 \cdot 0.8^3 \cdot (\frac{1}{16})^5} \approx 2^{11}$.
   - Total of 2^{26} pairs needed. Filtering on last two rounds leaves $0.8^3 \cdot (\frac{1}{16})^5 \cdot 0.8^8 \approx 2^{-24}$ of wrong pairs. *The clique method can be used since there are few pairs.*

1R-Attacks

1. Count on all bits of the subkey of the last round entering the S boxes with nonzero input XORs.

1R-Attacks

1. Count on all bits of the subkey of the last round entering the S boxes with nonzero input XORs.
2. Verify against r' itself and perform possibility checks on other S boxes in the last round.

1R-Attacks

1. Count on all bits of the subkey of the last round entering the S boxes with nonzero input XORs.
2. Verify against r' itself and perform possibility checks on other S boxes in the last round.
3. Example: DES reduced to 10 rounds.
   - 9-round iterative characteristic has probability $\approx 2^{-32}$.

1R-Attacks

1. Count on all bits of the subkey of the last round entering the S boxes with nonzero input XORs.
2. Verify against r' itself and perform possibility checks on other S boxes in the last round.
3. Example: DES reduced to 10 rounds.
   - 9-round iterative characteristic has probability $\approx 2^{-32}$.
   - Right pairs have $r' = \psi$ and 20 btits in l' going out of S4, ..., S8 are zero.

1R-Attacks

1. Count on all bits of the subkey of the last round entering the S boxes with nonzero input XORs.
2. Verify against r' itself and perform possibility checks on other S boxes in the last round.
3. Example: DES reduced to 10 rounds.
   - 9-round iterative characteristic has probability $\approx 2^{-32}$.
   - Right pairs have $r' = \psi$ and 20 btits in l' going out of S4, ..., S8 are zero.
   - Wrong pairs pass these checks with probability 2^{-52}. Thus, counting on 18 key bits has $S/N = \frac{2^{18} \cdot 2^{-32}}{4^3 \cdot 2^{-52}} = 2^3 2$. 2^{34} pairs are needed.

Complexity of Differential Cryptanalysis Attacks So Far

No. of rounds	No. pairs needed	No. pairs used	No. bits found	Characteristics		S/N		Comments
4	2^3	2^3	42	1	1	16	[6]	
6	2^7	2^7	30	3	1/16	2^{16}	*	
8	2^{15}	2^{13}	30	5	1/10,486	15.6	[24]	
8	2^{17}	2^{13}	30	5	1/10,486	1.2	[18]	
8	2^{20}	2^{19}	30	5	1/55,000	1.5	[24]	The iterative characteristic
9	2^{25}	2^{24}	30	6	1/1,000,000	1.0	[30]	Extension to six rounds
9	2^{26}	8	48	7	2^{-24}	2^{29}	*	
10	2^{34}	4	18	9	2^{-32}	2^{32}	*	
11	2^{35}	2^{11}	48	9	2^{-32}	2^{21}	*	
12	2^{42}	4	18	11	2^{-40}	2^{24}	*	
13	2^{43}	2^{19}	48	11	2^{-40}	4	[30]	
14	2^{50}	4	18	13	2^{-48}	2^{16}	*	
15	2^{51}	2^{27}	48	13	2^{-48}	2.5	[42]	Needs a huge memory. With less memory needs 2^{57} pairs
16	2^{57}	2^5	18	15	2^{-56}	2^8	*	Slower than exhaustive search

Figure 11: Summary of time and space complexity of differential cryptanalysis on DES.

Main Idea of the New Attack

1. The iterative characteristic by itself is not enough due to low probability.

Main Idea of the New Attack

1. The iterative characteristic by itself is not enough due to low probability.
2. We need to add an *extra* round at no additional cost.

Main Idea of the New Attack

1. The iterative characteristic by itself is not enough due to low probability.
2. We need to add an *extra* round at no additional cost.
3. A new round 1 created followed by 15-round 2R-attack to speed up cryptanalysis and reduce memory.

Main Idea of the New Attack

1. The iterative characteristic by itself is not enough due to low probability.
2. We need to add an *extra* round at no additional cost.
3. A new round 1 created followed by 15-round 2R-attack to speed up cryptanalysis and reduce memory.
4. This attack has two phases: *data collection* and *data analysis*.

Data Collection Phase

1. Want to generate plaintexts that are fed to 15-round attack after first round.

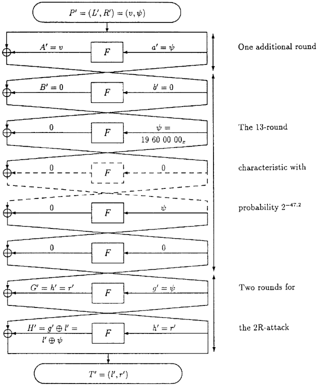

Figure 12: Modified 2R-attack on DES.

Data Collection Phase

1. Want to generate plaintexts that are fed to 15-round attack after first round.
2. Let $v_0, \ldots, v_{4095}$ be the 2^{12} 32-bit constants consisting of all possible values at the 12 output positions of S1, S2 and S3 after the first round, and zero elsewhere.

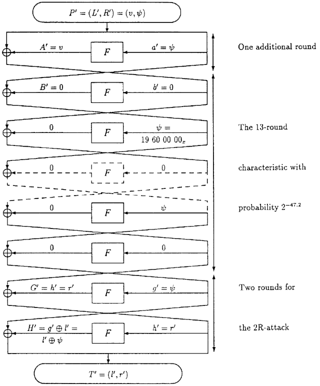

Figure 12: Modified 2R-attack on DES.

Data Collection Phase

1. For arbitrary 64-bit P, define

$$P_i \triangleq P \oplus (v_i, 0) \tag{16}$$

$$\bar{P}_i \triangleq (P \oplus (v_i, 0)) \oplus (0, \psi) \tag{17}$$

$$T_i \triangleq DES(P_i, K) \tag{18}$$

$$\bar{T}_i \triangleq DES(\bar{P}_i, K). \tag{19}$$

Data Collection Phase

1. For arbitrary 64-bit P, define

$$P_i \triangleq P \oplus (v_i, 0) \tag{16}$$

$$\bar{P}_i \triangleq (P \oplus (v_i, 0)) \oplus (0, \psi) \tag{17}$$

$$T_i \triangleq DES(P_i, K) \tag{18}$$

$$\bar{T}_i \triangleq DES(\bar{P}_i, K). \tag{19}$$

2. Observe that $P_i \oplus \bar{P}_j = (v_k, \psi)$. Each v_k occurs exactly 2^{12} times (why?).

Data Collection Phase

1. For arbitrary 64-bit P, define

$$P_i \triangleq P \oplus (v_i, 0) \tag{16}$$

$$\bar{P}_i \triangleq (P \oplus (v_i, 0)) \oplus (0, \psi) \tag{17}$$

$$T_i \triangleq DES(P_i, K) \tag{18}$$

$$\bar{T}_i \triangleq DES(\bar{P}_i, K). \tag{19}$$

2. Observe that $P_i \oplus \bar{P}_j = (v_k, \psi)$. Each v_k occurs exactly 2^{12} times (why?).
3. *We still don't know the best k for which $\psi \to v_k$.*

Data Collection Phase

1. For arbitrary 64-bit P, define

$$P_i \triangleq P \oplus (v_i, 0) \tag{16}$$

$$\bar{P}_i \triangleq (P \oplus (v_i, 0)) \oplus (0, \psi) \tag{17}$$

$$T_i \triangleq DES(P_i, K) \tag{18}$$

$$\bar{T}_i \triangleq DES(\bar{P}_i, K). \tag{19}$$

2. Observe that $P_i \oplus \bar{P}_j = (v_k, \psi)$. Each v_k occurs exactly 2^{12} times (why?).
3. *We still don't know the best k for which $\psi \rightarrow v_k$.*
   - Exhaustive search over the 2^{24} pairs is too slow.

Data Collection Phase

1. For arbitrary 64-bit P, define

$$P_i \triangleq P \oplus (v_i, 0) \tag{16}$$

$$\bar{P}_i \triangleq (P \oplus (v_i, 0)) \oplus (0, \psi) \tag{17}$$

$$T_i \triangleq DES(P_i, K) \tag{18}$$

$$\bar{T}_i \triangleq DES(\bar{P}_i, K). \tag{19}$$

2. Observe that $P_i \oplus \bar{P}_j = (v_k, \psi)$. Each v_k occurs exactly 2^{12} times (why?).
3. *We still don't know the best k for which $\psi \to v_k$.*
   - Exhaustive search over the 2^{24} pairs is too slow.
   - Exploit the cross-product structure to speed up the search.

Data Collection Phase

1. A right pair has zero output XOR at S4, . . . , S8 of the last round.

Data Collection Phase

1. A right pair has zero output XOR at S4, . . . , S8 of the last round.
2. Hash the 2^{13} plaintexts $P_i, \bar{P}_i$ based on these 2^{20} values. Only $2^{24} \cdot 2^{-20} = 2^4 = 16$ pairs will survive.

Data Collection Phase

1. A right pair has zero output XOR at S4, . . . , S8 of the last round.
2. Hash the 2^{13} plaintexts $P_i, \bar{P}_i$ based on these 2^{20} values. Only $2^{24} \cdot 2^{-20} = 2^4 = 16$ pairs will survive.
3. Additional S boxes can be tested in the first, fifteenth and sixteenth ronud to discard about 92.55% of surviving pairs, leaving about $16 \cdot 0.0745 = 1.19$ pairs per structure.

Data Collection Phase

1. A right pair has zero output XOR at S4, ..., S8 of the last round.
2. Hash the 2^{13} plaintexts $P_i, \bar{P}_i$ based on these 2^{20} values. Only $2^{24} \cdot 2^{-20} = 2^4 = 16$ pairs will survive.
3. Additional S boxes can be tested in the first, fifteenth and sixteenth ronud to discard about 92.55% of surviving pairs, leaving about $16 \cdot 0.0745 = 1.19$ pairs per structure.
   - Input XOR for right pairs in first and fifteenth rounds are fixed, so use number of non-zero entries of corresponding lines in the pairs XOR distribution table. Fraction of surviving pairs is $(\frac{14}{16} \cdot \frac{13}{16} \cdot \frac{15}{16})^2 \cdot 0.8^8 = 0.0745$.

Data Collection Phase

1. A right pair has zero output XOR at S4, . . . , S8 of the last round.
2. Hash the 2^{13} plaintexts $P_i, \bar{P}_i$ based on these 2^{20} values. Only $2^{24} \cdot 2^{-20} = 2^4 = 16$ pairs will survive.
3. Additional S boxes can be tested in the first, fifteenth and sixteenth ronud to discard about 92.55% of surviving pairs, leaving about $16 \cdot 0.0745 = 1.19$ pairs per structure.
   - Input XOR for right pairs in first and fifteenth rounds are fixed, so use number of non-zero entries of corresponding lines in the pairs XOR distribution table. Fraction of surviving pairs is $(\frac{14}{16} \cdot \frac{13}{16} \cdot \frac{15}{16})^2 \cdot 0.8^8 = 0.0745$.
4. Each structure has right pair with probability $2^{12} \cdot 2^{-47.2} = 2^{-35.2}$. Multiple structures needed.

Data Collection Phase

1. A right pair has zero output XOR at S4, . . . , S8 of the last round.
2. Hash the 2^{13} plaintexts $P_i, \bar{P}_i$ based on these 2^{20} values. Only $2^{24} \cdot 2^{-20} = 2^4 = 16$ pairs will survive.
3. Additional S boxes can be tested in the first, fifteenth and sixteenth ronud to discard about 92.55% of surviving pairs, leaving about $16 \cdot 0.0745 = 1.19$ pairs per structure.
   - Input XOR for right pairs in first and fifteenth rounds are fixed, so use number of non-zero entries of corresponding lines in the pairs XOR distribution table. Fraction of surviving pairs is $(\frac{14}{16} \cdot \frac{13}{16} \cdot \frac{15}{16})^2 \cdot 0.8^8 = 0.0745$.
4. Each structure has right pair with probability $2^{12} \cdot 2^{-47.2} = 2^{-35.2}$. Multiple structures needed.
5. *Input to data analsyis phase contains mix of right and wrong pairs.*

Data Analysis Phase

1. Uses *negligible* space. Fewer suggested key values can be tried immediately.

Data Analysis Phase

1. Uses *negligible* space. Fewer suggested key values can be tried immediately.
2. Due to key scheduling

Data Analysis Phase

1. Uses *negligible* space. Fewer suggested key values can be tried immediately.
2. Due to key scheduling
   - 28 bits of left key register are used as inputs to S1, S2 and S3 of first and fifteenth rounds, and S1, . . . , S4 of sixteenth round.

Data Analysis Phase

1. Uses *negligible* space. Fewer suggested key values can be tried immediately.
2. Due to key scheduling
    - 28 bits of left key register are used as inputs to S1, S2 and S3 of first and fifteenth rounds, and S1, . . . , S4 of sixteenth round.
    - 24 key bits of right key register are used in sixteenth round. Total of 28 + 24 = 52 key bits used.

Data Analysis Phase

1. Uses *negligible* space. Fewer suggested key values can be tried immediately.
2. Due to key scheduling
   - 28 bits of left key register are used as inputs to S1, S2 and S3 of first and fifteenth rounds, and S1, . . . , S4 of sixteenth round.
   - 24 key bits of right key register are used in sixteenth round. Total of 28 + 24 = 52 key bits used.
3. $\frac{2^{-32}}{0.8^8}$ keys survive by comparing output XOR to expected value.

Data Analysis Phase

1. Uses *negligible* space. Fewer suggested key values can be tried immediately.
2. Due to key scheduling
   - 28 bits of left key register are used as inputs to S1, S2 and S3 of first and fifteenth rounds, and S1, . . . , S4 of sixteenth round.
   - 24 key bits of right key register are used in sixteenth round. Total of 28 + 24 = 52 key bits used.
3. $\frac{2^{-32}}{0.8^8}$ keys survive by comparing output XOR to expected value.
4. $\frac{2^{-12}}{\frac{14}{16}\cdot\frac{13}{16}\cdot\frac{15}{16}}$ key values remain by comparing output XOR of three S boxes in the first and fifteenth round each.

Data Analysis Phase

1. Uses *negligible* space. Fewer suggested key values can be tried immediately.
2. Due to key scheduling
   - 28 bits of left key register are used as inputs to S1, S2 and S3 of first and fifteenth rounds, and S1, . . . , S4 of sixteenth round.
   - 24 key bits of right key register are used in sixteenth round. Total of 28 + 24 = 52 key bits used.
3. $\frac{2^{-32}}{0.8^8}$ keys survive by comparing output XOR to expected value.
4. $\frac{2^{-12}}{\frac{14}{16}\cdot\frac{13}{16}\cdot\frac{15}{16}}$ key values remain by comparing output XOR of three S boxes in the first and fifteenth round each.
5. Multiplying them together, each pair suggests 0.84 values for these 52 key bits. In total, each structure suggests $1.19 \cdot 0.84 \cdot 2^4 = 16$ values.

Data Analysis Phase

1. Verify each key by peeling up two rounds and checking against output of 13-round characteristic. Costs $16 \cdot \frac{2}{16} \cdot 2 = 4$ equivalent DES operations.

		K16									
		Left Key Register					Right Key Register				
		S1	S2	S3	S4	X	S5	S6	S7	S8	X
K1	S1		2	1	1	2					
	S2	2		1	2	1					
	S3	2			3	1					
	S4	2	3	1							
	X		1	3							
	S5							1	2	2	1
	S6						3		2	1	
	S7							2		2	2
	S8						2	3			1
	X						1		2	1	

Figure 13: Common bits between $K1$ and $K16$.

Data Analysis Phase

1. Verify each key by peeling up two rounds and checking against output of 13-round characteristic. Costs $16 \cdot \frac{2}{16} \cdot 2 = 4$ equivalent DES operations.
   - After this filtering, perform trial encryption to determine the right key.

		K16									
		Left Key Register					Right Key Register				
		S1	S2	S3	S4	X	S5	S6	S7	S8	X
K1	S1		2	1	1	2					
	S2	2		1	2	1					
	S3	2			3	1					
	S4	2	3	1							
	X		1	3							
	S5							1	2	2	1
	S6						3		2	1	
	S7							2		2	2
	S8						2	3			1
	X						1		2	1	

Figure 13: Common bits between $K1$ and $K16$.

Data Analysis Phase

1. Verify each key by peeling up two rounds and checking against output of 13-round characteristic. Costs $16 \cdot \frac{2}{16} \cdot 2 = 4$ equivalent DES operations.
    - After this filtering, perform trial encryption to determine the right key.
2. $S/N = \frac{2^{52} \cdot 2^{-47.2}}{\frac{1.19}{2^{12}} \cdot 0.84} = 2^{16.8}$. If this test succeeds, then we have found the right key with very high probability.

		K16									
		Left Key Register					Right Key Register				
		S1	S2	S3	S4	X	S5	S6	S7	S8	X
K1	S1		2	1	1	2					
	S2	2		1	2	1					
	S3	2			3	1					
	S4	2	3	1							
	X		1	3							
	S5							1	2	2	1
	S6						3		2	1	
	S7							2		2	2
	S8						2	3			1
	X						1		2	1	

Figure 13: Common bits between $K1$ and $K16$.

Data Analysis Phase

1. Verify each key by peeling up two rounds and checking against output of 13-round characteristic. Costs $16 \cdot \frac{2}{16} \cdot 2 = 4$ equivalent DES operations.
   - After this filtering, perform trial encryption to determine the right key.
2. $S/N = \frac{2^{52} \cdot 2^{-47.2}}{\frac{1.19}{2^{12}} \cdot 0.84} = 2^{16.8}$. If this test succeeds, then we have found the right key with very high probability.
3. Using common key bits, we can speed up the data analysis, as shown in Figure 13.

		K16									
		Left Key Register					Right Key Register				
		S1	S2	S3	S4	X	S5	S6	S7	S8	X
K1	S1		2	1	1	2					
	S2	2		1	2	1					
	S3	2			3	1					
	S4	2	3	1							
	X		1	3							
	S5							1	2	2	1
	S6						3		2	1	
	S7							2		2	2
	S8						2	3			1
	X						1		2	1	

Figure 13: Common bits between $K1$ and $K16$.

General Form of the Attack

Theorem 8

Given a characteristic with probability p and signal-to-noise ratio S/N for an iterated cryptosystem with k key bits, we can apply an attack which encrypts $\frac{2}{p}$ chosen plaintexts in the data collection phase and whose complexity is $\frac{2^k}{S/N}$ encryptions during the data analysis phase.

General Form of the Attack

Theorem 8

Given a characteristic with probability p and signal-to-noise ratio S/N for an iterated cryptosystem with k key bits, we can apply an attack which encrypts $\frac{2}{p}$ chosen plaintexts in the data collection phase and whose complexity is $\frac{2^k}{S/N}$ encryptions during the data analysis phase.

Appropriately chosen metastructures can reduce the number of plaintexts to $\frac{1}{p}$. Further, the effective time complexity can be reduced by a factor of $f \leq 1$ if a wrong key can be discarded by carrying out a fraction f of the rounds.

Results

Rounds	Chosen Plaintexts	Analyzed Plaintexts	Complexity of Analysis	Best Previous Time	Best Previous Space
8	2^{14}	4	2^{9}	2^{16}	2^{24}
9	2^{24}	2	2^{32}	2^{26}	2^{30}
10	2^{24}	2^{14}	2^{15}	2^{35}	–
11	2^{31}	2	2^{32}	2^{36}	–
12	2^{31}	2^{21}	2^{21}	2^{43}	–
13	2^{39}	2	2^{32}	2^{44}	2^{30}
14	2^{39}	2^{29}	2^{29}	2^{51}	–
15	2^{47}	2^{7}	2^{37}	2^{52}	2^{42}
16	2^{47}	2^{36}	2^{37}	2^{58}	–

Figure 14: Results of memoryless DES attack.