Breaking DES & Introducing AES

Luke Anderson

luke@lukeanderson.com.au

24th March 2016

University Of Sydney

Overview

Attacks on DES

DES Keys

Given one plaintext/ciphertext pair (m, c), there is a high probability that only one key will satisfy:

$$c = \mathsf{DES}(m, k)$$

Consider DES as a collection of permutations: $\pi(1)$ ...$\pi(2^{56})$

If π_i are independent permutations then $\forall(m, k)$:

$$\begin{aligned}
&\mathsf{Pr}[\exists k_1 \neq k : \mathsf{DES}(m, k_1) = DES(m, k)] \\
&= 256 \times 2^{-64} \\
&= 2^{-56} \\
&= 1.39 \times 10^{-17} \\
&= 0.00000000000000139\%
\end{aligned}$$

Thus, given one (m, c) pair, the key is (almost definitely) uniquely determined.
The problem is to find k.

Attacks on DES

Exhaustive Key Search

- Strong **n**-bit block cipher, **j**-bit key, the key can be recovered on average in 2^{j-1} operations, given a small number ($< (j+4)/n$) of plaintext/ciphertext pairs
- For **DES**, $\mathbf{j = 56}$, $\mathbf{n = 64}$ so exhaustive key search is expected to yield the key in 2^{55} operations.

Ciphertext-Only DES key search

- Example: DES is used to encrypt 8×8 ASCII characters ($=$ 64 bits) per block - one bit is a *parity* bit.
- Let's say we try decrypting - this will yeild all 8 correct parity bits with probability 2^{-8} ($\approx$ 0.4%)
- Thus with **t** blocks, we can safely say that it would have probability of 2^{-8t}
- So, using this - 2^{56} keys - probability of a correct key with all valid parity bits $= (1 - 2^{8t})$
- Therefore, **t** ~5-10 blocks are enough for $>$ 99.99999% sure.

Reducing Effort in Attacks on DES

DES is a Feistel Network, so:

Complementation Property

$$\mathsf{DES}(\neg m, \neg k) = \neg\mathsf{DES}(m, k)$$

So, using a *Chosen Plaintext Attack*:

if $c_1 = \mathsf{DES}(m, k)$ and $c_2 = \mathsf{DES}(\neg m, k)$ **then**
    if $\mathsf{DES}(m, k_1) \neq c_1$ OR c_2 **then**
        $k \neq k_1$ or $\neg k_2$
    end if
end if

Therefore, the search space is **HALVED!**

DES Enhancements

2DES

Double Encryption with DES (*2DES*)

2DES **IS BAD!**

$$\text{2DES}_{k_1,k_2}(m) = E_{k_1}(E_{k_2}(m))$$

Vulnerable to the **meet-in-the-middle** attack with known plaintext.

Example:
for a fixed message, **m**, create a table of all possible ciphertext with each 56-bit encryption keys:

$$E_k(m) \text{ for all } k \in \{0,1\}^{56}$$

Then, for $c = E_{k_1,k_2}(m)$, try to decrypt:

$$D_k(c) \text{ for all } k \in \{0,1\}^{56}$$

Until $D_k(c)$ appears in the table, since $D_{k_1}(c) = E_{k_2}(m)$.

2DES

What does this mean?

2DES can be broken in 2^{56} operations on average, using 2^{56} memory slots. (A time-space trade-off!).

This is not good when there should be 112-bits $(56 + 56)$ of key.

3DES

Two-key Triple DES (3DES) - DES 3 times, 2 keys. (*112 bits*)

$$3DES_{k_1,k_2}(m) = E_{k_1}(D_{k_2}(E_{k1}(m)))$$

The strength of DES/3DES is that it does not form a group!

$$DES_{k_1}(DES_{k_2}(m)) \neq DES_{k_3}(m)$$

Let's consider that *time-space trade-off* in 2DES

For time $\frac{2^{(56+64)}}{s}$ and space **s**, we can recover k_1 and k_2 in 2DES.

If $s > 28$ - we can do better than exhaustive search.

If you have three distinct keys - then it has 168 key bits.
(The effective key length $= 112$ bits becasue of "*meet-in-the-middle*")

If you use two keys $(\mathrm{k}_1 = \mathrm{k}_3, \mathrm{k}_2)$ then it has 112 key bits.
(The effective key length $= 80$ bits due to chosen/known plaintext attacks)

DESX

A modification of DES to avoid exhaustive key search is **DESX**.

$$\begin{aligned} \mathbf{k_1} &= \text{56bits (DES Key)} \\ \mathbf{k_2} &= \text{64bits (Whitening Key)} \\ \mathbf{k_3} &= h(k_2, k_3) \\ &= 64bits \end{aligned}$$

$$\mathbf{DESX}_{k_1,k_2,k_3}(m) = \mathbf{k_3} \oplus E_{k_1}(m \oplus \mathbf{k_2})$$

The **whitening key** gives greater resiliance to brute force attacks.

DESX

Given **j** plaintext / ciphertext pairs, the effective key size is greater or equal to:

$$\begin{aligned} |k| + n - 1 - \log(j) &= 56 + 64 - 1 - \log(j) \\ &= 119 - \log(j) \\ &\geqslant 100 bits \end{aligned}$$

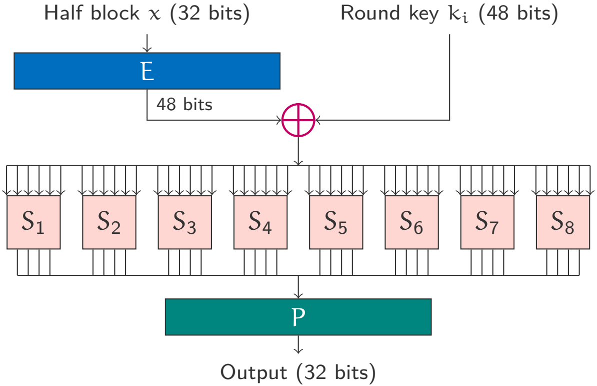
DES Round Function

The Function

$$\mathbf{F(x, k_i)} : \{0,1\}^{32} \times \{0,1\}^{48} \to \{0,1\}^{32}$$

Half block is reversibly expanded to 48 bits in the **Expander Function** (E).

S-Box collapses groups of 6 bits into groups of 4 bits.
(i.e. convert 48 bits back to 32 bits)



Cryptanalysis

Differential Cryptanalysis

Better-than-brute-force approach to attacking DES.

Utilises (plaintext, ciphertext) pairs with Chosen Plaintext Attack (CPA). Involves looking at the XOR of two texts.

We consider any **s-box** function $F(\mathbf{x}, \mathbf{k_i})$:

Define the difference measure (on input) as:

$$\begin{aligned} \Delta &= b_1 \oplus b_2 \\ &= (x_1 \oplus k_1) \oplus (x_2 \oplus k_i) \\ &= x_1 \oplus x_2 \end{aligned}$$

The input XOR $(b_1 \oplus b_2)$ does not depend on the key, but the output XOR $(e_1 \oplus e_2)$ does.

Differential Cryptanalysis

Now, define the set $\Delta(b)$ consisting of ordered pairs (b_1, b_2):

$$\Delta(b) = \{(b_1, b_2) \in \{0, 1\}^6 | b_1 \oplus b_2 = b\}$$

Where

$$|\Delta(b)| = 2^6 = 64$$

Example

If $\mathbf{b = 110100}$, then consider the first S-Box pairs to be:

$$\begin{array}{lcccc} \Delta(b) = & \{(000000, 110100), & (000001, 110101), & \ldots & (111111, 001011)\} \\ & \oplus & \oplus & & \oplus \\ & 110100 & 110100 & & 110100 \end{array}$$

If this is done for all 64 paris in $\Delta(b)$ then the distribution of output XORs $(e_1 \oplus e_2)$:

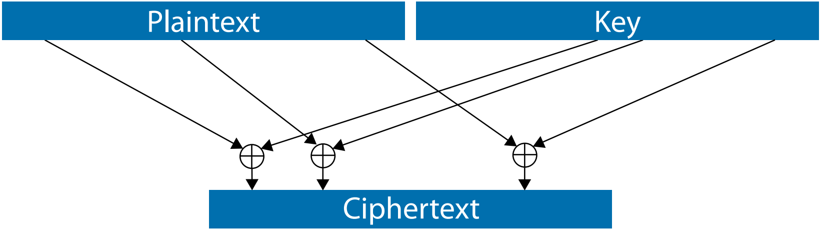
$(e_1 \oplus e_2)$	0000	0001	0010	0011	...	1111
	0	8	16	6		6

So, if $(b_1 \oplus b_2) = 110100$ and $(e_1 \oplus e_2) = 0001$, then (b_1, b_2) must be one of the eight possible pairs, $\therefore$ b_1 is one of 16 possible values.

Since x_1 is the known plaintext, the 6 bits of the key $\oplus$ $x_1 = b_1$ are one of the 16 possible values. This is repeated with different Δ to make deductions about the key!

Linear Cryptanalysis

Consider the ciphertext derived by combining certain bits from plaintext and key:



The cipher can easily be broken, for example:

$$\begin{aligned} c[1] &= p[4] \oplus p[17] \oplus k[5] \oplus k[3] \\ \text{i.e. } k[3] \oplus k[5] &= c[1] \oplus p[4] \oplus p[17] \end{aligned}$$

This is because the cipher is *linear*.

Notation:

$$p[i_1, ..., i_u] = p[i_1] \oplus p[i_2] \oplus \cdots \oplus p[i_u]$$

(the xor bits of the plaintext)

We also define:

$$\rho = Pr[p[i_1, ..., i_u] \oplus c[j_1, ..., j_v] = k[s_1, ..., s_w]]$$

Now, if $|\rho - 0.5|$ is large, then we can guess $k[s_1, ..., s_w]$

Optimally, for a break, $|\rho - 0.5| = 0.5 (\rho = 0 \text{ or } 1)$
(A *perfect* cipher would have $\rho = 0.5$)

Algorithm to recover key bits

Given **R** plaintext, ciphertext pairs (Note: **R** is large):

if $\rho > 0.5$ **then**
$k[s_1, ..., s_w] = \text{majority}\{p[i_1, ..., i_u]\} \oplus c[j_1, ..., j_v]\}$
over all plaintext, ciphertext pairs
else if $\rho < 0.5$ **then**
$k[s_1, ..., s_w] = \text{minority} = 1 \oplus \text{majority}$
end if

Fact

If given $R \geqslant (\rho - 0.5)^{-2}$ then the correct value of $k[s_1, ..., s_w]$ is obtained with probability $> 97.7\%$.

Linear Cryptanalysis of DES

In 1993, **Matsui** made an observation about DES.

Approximates the 5th S-Box as a linear function:

$$\begin{aligned} \rho_5 &= Pr[x[4] = S5(x)[0,1,2,3]] \\ &= \frac{12}{64} \\ &= 0.19 \end{aligned}$$

Note: $x \in \{0,1\}^6$

What does this mean?
For the i^{th} DES round:

$$\begin{aligned} Pr_i[R_i[15] \oplus F(R_i, k_i)[7,18,24,29] &= k_i[22]] \\ &= \rho_5 \\ &= 0.19 \end{aligned}$$

Where the bits have been chosen to undo the permutation.

Attack on 3DES

From the first round, we write:

$$\Pr[r_1[7,18,24,29] \oplus l_0[7,18,24,29] \oplus r_0[15] = k_0[22]] = \rho_5$$

From the last round, we now write:

$$\Pr[r_1[7,18,24,29] \oplus c_r[7,18,24,29] \oplus c_l[15] = k_0[22]] = \rho_5$$

Which is then XORed to give:

$$\begin{aligned}\Pr[r_1[7,18,24,29] \oplus c_l[7,18,24,29] \oplus c_l[15] \oplus r_0[15] = k_0[22] \oplus k_2[22]] \\ = \rho_5 \cdot \rho_5 + (1-\rho_5) \\ = 0.7\end{aligned}$$

Then we can find $k_0[22] \oplus k_2[22]$ using $R = (0.7 - 0.5)^{-2} = 25$ plaintext/ciphertext pairs.

DES Strength against attacks

Attack vs Complexity

Attack	Messages		Requirements	
	Known	Chosen	Storage	Processing
Exhaustive Precomputation	-	1	2^{56}	1
Exhaustive Search	1	-	Neg.	2^{55}
Linear Cryptanalysis	2^{43}(85%)	-	Texts	2^{43}
	2^{38}(10%)	-	Texts	2^{50}
Differential Cryptanalysis	-	2^{47}	Texts	2^{47}
	2^{55}	-	Texts	2^{55}

Replacing DES

Replacing DES

US Government wanted DES used as a “**standard**”

RSA Security wanted to demonstrate that DES *sucked*.. it was weak because of the key length.

Timeline:

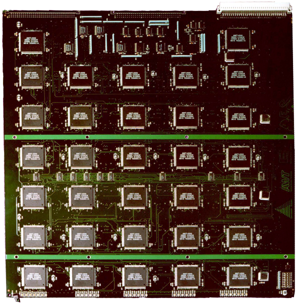
- (**1997**) First DES challenge solved in **96** days using distributed computing (*idle CPU*)
  Second DES challenge solved in **41** days using *distributed.net* (*idle CPU*)
- (**1998**) EFF created **Deep Crack** for $250K - decrypted in **56 hours**
- (**1999**) Deep Crack + Distributed.net decrypted DES in **22h 15min**

EFFs DES Cracker

Whitfield Diffie and **Martin Hellman** (Stanford Uni.) estimated that a machine fast enough to test that many keys in a day would cost about $20 million in 1976. (Minimal cost for NSA or governments...)

Composed of 1856 custom **ASIC DES** chips, 90 billion ($\approx 2^{36}$) keys per second!

Entire key space in **9 days**!
(On average, key found in half that time!)



(**2006**) COPACOBANA ($10k) recover DES key in $\approx$ 6.4 days
(**2008**) Reduced to less than one day using 128 off the shelf **FPGA**s

Introducing AES

Advanced Encryption Standard (AES)

In 1997 NIST announced that a competition would be held to choose a new cipher to replace the outdated DES cipher, this to be was named the Advanced Encryption Standard – AES.

Of the contenders, they chose Rijndael as the new AES.

- Block cipher
- 128 bit blocks
- 128/192/256 bit keys
- Criteria:
    - Strength $\geqslant$ 3DES, but much better efficiency
    - Flexible - can be implemented in software, hardware or smartcards
    - Simple and Elegant
- Royalty-free worldwide
- Security for over 30 years
- May protect sensitive data for over 100 years
- Public confidence in the cipher

AES Candidates

15 submissions from the international field.

A number of strong finalists:

Name	Type	Rounds	Rel. Speed (cycles)	Gates
Twofish	Feistel	16	1254	23k
Serpent	SP-network	32	1800	70k
Mars	Type-3 Feistel	32	1600	70k
Rijndael	SP-network	10, 12, 14	1276	-
RC6	Feistel	20	1436	-

AES

Rijndael (pronounced [rɛinda:l] "rain-dahl") announced October 2000

- Operates on 128 bit blocks
- Key length is variable: 128, 192 or 256 bits
- It is an SP-network (substitution-permutation network)
- Uses a single S-box which acts on a byte input to give a byte output (a 256 byte lookup table):

  $$S(x) = M(x^{-1}) + b \text{ over } \mathrm{GF}(2^8)$$

  Where M is a predefined matrix, b is a constant and GF is chosen Galois field (nonlinearity comes from $x \mapsto x^{-1}$).
- Construction gives tight differential and linear bounds

AES Overview - Rounds

The number of rounds are variable:

- 10 rounds – 128 bit keys
- 12 rounds – 192 bit keys
- 14 rounds – 256 bit keys

Rounds have a 50% margin of safety based on current known attacks. Potential attacks (which require an *enormous* number of plaintext/ciphertext pairs) are possible on:

- Only 6 rounds for 128 bit keys
- Only 7 rounds for 192 bit keys
- Only 9 rounds for 256 bit keys

Safety against possible attacks believed to currently be $\approx 100\%$

Stick Figure guide to AES