CT255
Introduction to Cyber-Security

Lecture 8

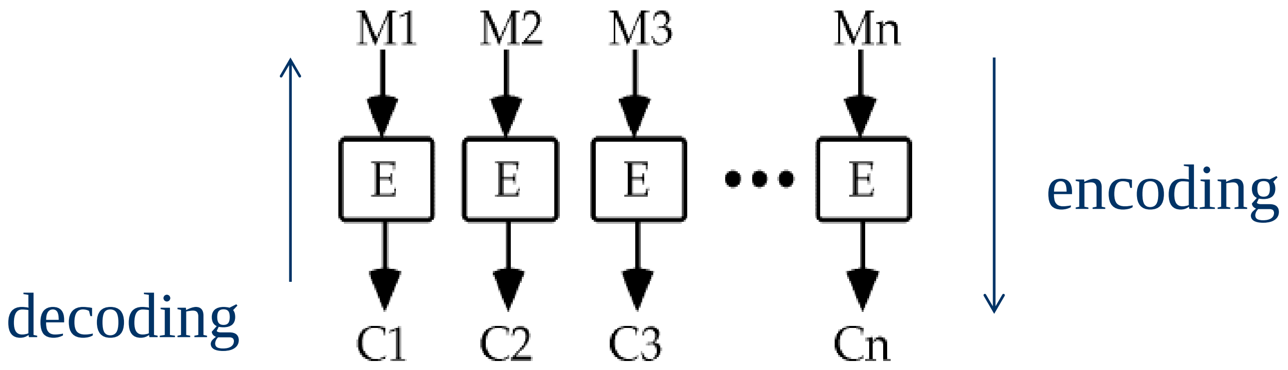
Block Ciphers and Stream Ciphers

Dr. Michael Schukat, 2019-2022

BLOCK CIPHERS

Encryption Algorithms based on Block Ciphers

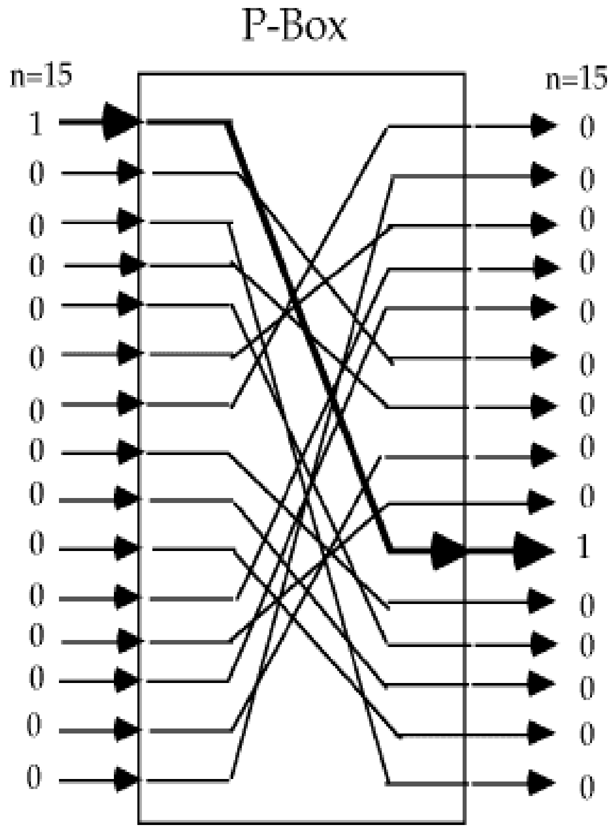
- In a block cipher the message is broken into blocks M1, M2, etc. of K bits length, each of which is then encrypted



  - Most ciphers we saw before process blocks of just one character
- Claude Shannon suggested to use the two primitive cryptographic operations as building blocks for such ciphers:
  - substitution
  - permutation

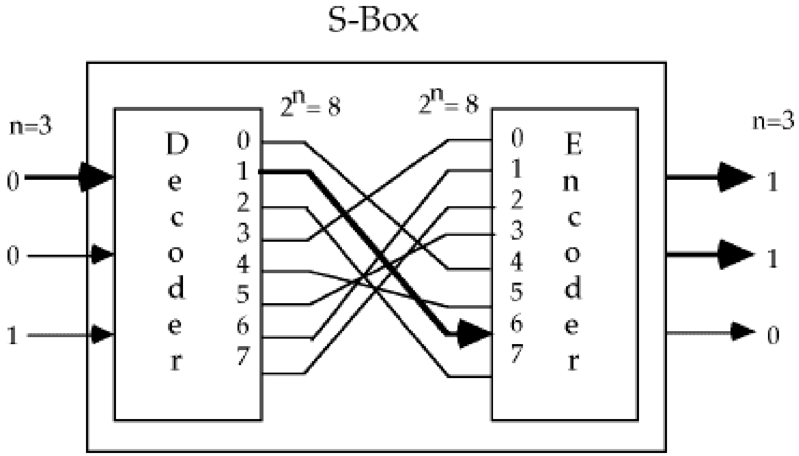
The Permutation Operation

- A binary word (i.e. block) has its bits reordered (permuted)
- The re-ordering forms the key
- Operation represented by a **P-box**
- The example allows for 15! = 1,307,674,368,000 combinations
- The key describes the combination used



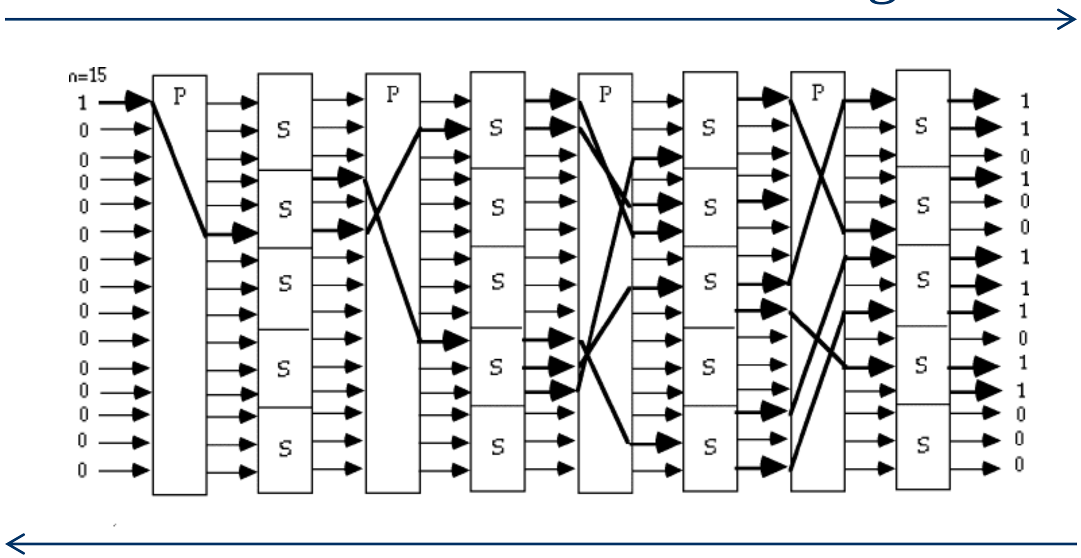
The Substitution Operation

- A binary word is replaced by some other binary word
- The whole substitution function forms the key
- Operation represented by an **S-box**
- The box below allows for 8! = 40320 combinations
- The key describes the combination used



Substitution-Permutation Network

encoding



decoding

- The key describes the internal wiring of all S-boxes and P-boxes
- The same key can be used for encoding and decoding, hence it is a **private key encryption algorithm**
- The direction of the process determines encoding / decoding

Confusion and Diffusion

- A cipher needs for obvious reasons to completely obscure statistical properties of original message
- Shannon introduced two terms to describe this:
  - **Diffusion** seeks to make the statistical relationship between the plaintext and ciphertext as complex as possible
  - **Confusion** seeks to make the relationship between the statistics of the ciphertext and the value of the encryption key as complex as possible
- Both thwart attempts to deduce the key used via a cryptanalysis (as seen before)

Confusion and Diffusion in Practice

- Example DES (→later): A swap of a single bit either in the key or in the plaintext result in a significant change in the ciphertext
- Note that DES encrypts a message over 16 iterations (rounds)

(a) Change in Plaintext		(b) Change in Key	
Round	Number of bits that differ	Round	Number of bits that differ
0	1	0	0
1	6	1	2
2	21	2	14
3	35	3	28
4	39	4	32
5	34	5	30
6	32	6	32
7	31	7	35
8	29	8	34
9	42	9	40
10	44	10	38
11	32	11	31
12	30	12	33
13	30	13	28
14	26	14	26
15	29	15	34
16	34	16	35

Ollscoil na Gaillimhe
University of Galway

Important Block Cipher Principle: Reversible Transformation

- Transformations must be reversible or non-singular, e.g.

Plaintext		Ciphertext
00	↔	11
01	↔	10
10	↔	00
11	↔	01

vs.

Plaintext		Ciphertext
00	↔	11
01	↔	10
10	↔	**00**
11	↔	**00**
	?	

- There must be a 1:1 association between a n-bit plaintext and an-bit ciphertext, otherwise mapping (encryption) is irreversible

Features of Private-Key Cryptography / Ciphers

- Traditional private/secret/single key cryptography uses one key, shared by only sender and receiver
- The algorithm / cipher itself is public, i.e. not a secret
- If the key is disclosed, communications are compromised
- The key is also **symmetric**, parties are equal
- Hence methods does not protect sender from receiver forging a message & claiming is sent by sender
- Examples include DES (Data Encryption Standard) and AES (Advanced Encryption Standard)

Examples AES

- Advanced Encryption Standard, successor of DES
- Modern block cipher with 128 bits block length
- Uses 128, 192 or 256 bit long keys
- The de-facto standard for secure encryption
- Widely used for
  - File / data encryption
  - Secure network (e.g. Internet) Communication

Why does Block and Key Length matter?

- Cryptographic algorithms with short block length can be tackled as seen with substitution cipher
- Large keys and long blocks prevent **brute-force attacks / searches**
  - Take the ciphertext and try all possible key combinations (or block permutations), until the decoded text makes sense

Ollscoil na Gaillimhe
University of Galway

Brute Force Search / Attacks

- A 56-bit key has a key space that contains 2^{56} keys
  - A prominent early day symmetric cipher called DES (Data Encryption Standard) used 56 bit keys… it is deemed unsafe since the 1990s
- A 128-bit key has 3.4E38 possible combinations
  - Generally accepted minimum key length today

Ollscoil na Gaillimhe
University of Galway

Brute Force Search

- Always possible to simply try every key
- Most basic attack, effort proportional to key size
- Assume that you either know or recognise plaintext

Key Size (bits)	Number of Alternative Keys	Time required at 1 decryption/µs	Time required at 10^6 decryptions/µs
32	$2^{32} = 4.3 \times 10^9$	2^{31} µs = 35.8 minutes	2.15 milliseconds
56	$2^{56} = 7.2 \times 10^{16}$	2^{55} µs = 1142 years	10.01 hours
128	$2^{128} = 3.4 \times 10^{38}$	2^{127} µs = 5.4×10^{24} years	5.4×10^{18} years
168	$2^{168} = 3.7 \times 10^{50}$	2^{167} µs = 5.9×10^{36} years	5.9×10^{30} years
26 characters (permutation)	$26! = 4 \times 10^{26}$	2×10^{26} µs = 6.4×10^{12} years	6.4×10^{6} years

The Feistel Cipher

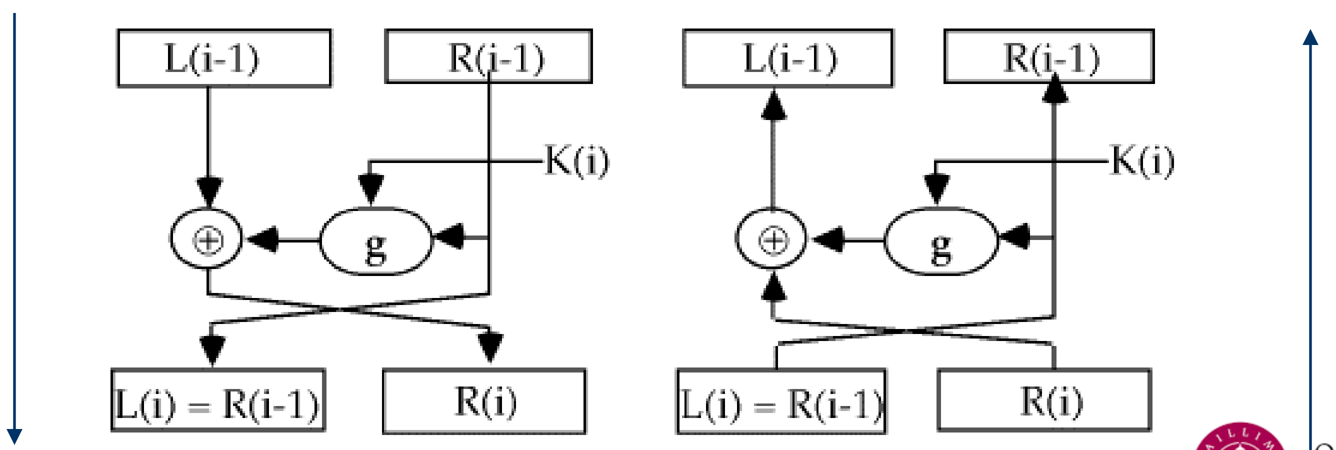
- In practice we need to be able to decrypt messages, as well as to encrypt them, hence either:
  - have to define inverses for each of the S & P-boxes, but this doubles the code/hardware needed, or
  - define a structure that is easy to reverse, so can use basically the same code or hardware for both encryption and decryption
- A **Feistel cipher** is such a structure
  - It is based on concept of the **invertible product cipher**
  - Most symmetric block ciphers are based on a Feistel Cipher structure

The Feistel Cipher

- Horst Feistel, working at IBM Thomas J Watson Research Labs, devised a suitable invertible cipher structure in early 70's
- One of Feistel's main contributions was the invention of a suitable structure which adapted Shannon's S-P network in an easily invertible structure
- Essentially the same hardware or software is used for both encryption and decryption, with just a slight change in how the keys are used

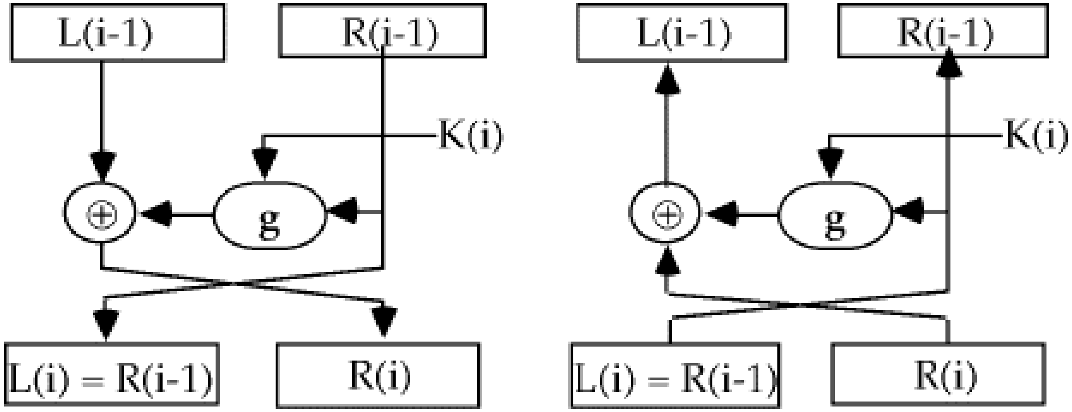
The Feistel Cipher – A Single Round

- The idea is to partition the input block into two halves, L(i−1) and R(i−1), and use only R(i−1) in the i^{th} round (part) of the cipher
- The function g incorporates one stage of the S-P network, controlled by part of the key K(i) known as the i^{th} subkey



The Feistel Cipher – A single Round

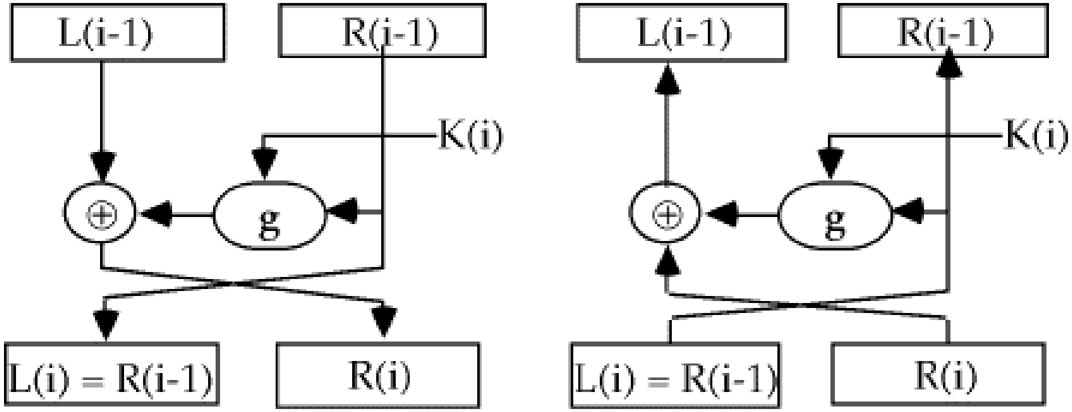
- A round of a Feistel cipher can be described functionally as:
  - `L(i) = R(i-1)`
  - `R(i) = L(i-1) EXOR g(K(i), R(i-1))`



Symmetry of Bitwise EXOR

- A EXOR B = C
  A EXOR C = B
  C EXOR B = A

	0	1
0	0	1
1	1	0



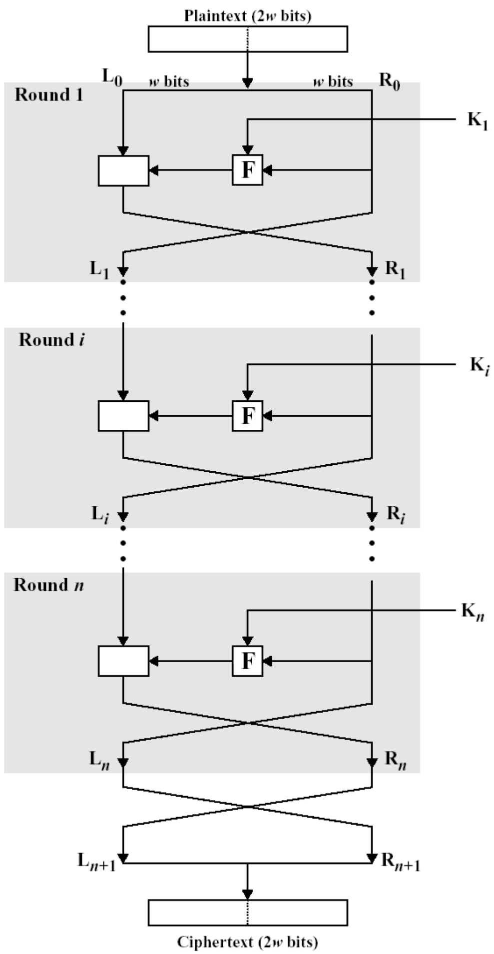
Ollscoil na Gaillimhe
University of Galway

Example

- Encoding of 01011110:
  - L(i − 1) = 0101 R(i − 1) = 1110
  - g(K(i), R(i−1)) = 1001 L(i) = 1110
  - R(i) = 0101 XOR 1001 = 1100
  - Therefore 01011110 becomes 11101100
- Decoding of 11101100:
  - L(i) = 1110 R(i) = 1100
  - g(K(i), R(i−1)) = 1001 R(i − 1) = 1110
  - L(i − 1) = 1100 XOR 1001 = 0101
  - Therefore 1110 1100 becomes 01011110

Ollscoil na Gaillimhe
University of Galway

A Feistel Network



- Perform multiple transformations (single rounds) sequentially, whereby output of i^{th} round becomes the input of the $(i+1)^{th}$ round
- Every round gets is own subkey, which is derived from master key
- Decryption process goes from bottom to top

Ollscoil na Gaillimhe
University of Galway

Feistel Cipher Design Elements

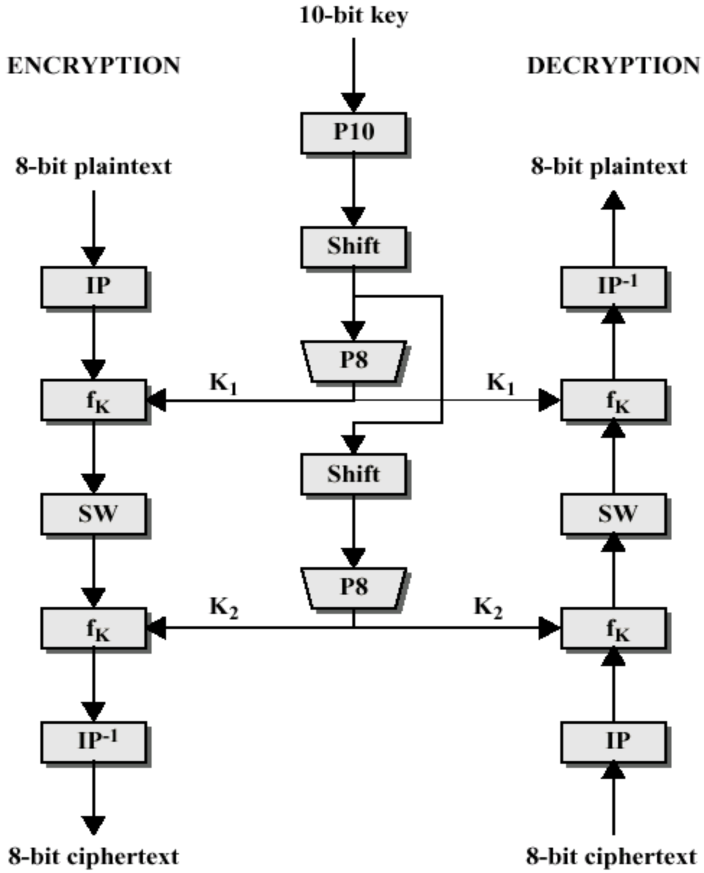
- Block size
- Key size
- Number of rounds
- Subkey generation algorithm
- Round function
- Fast software encryption/decryption

Simple Methods for Subkey Generation

- Multiple subkeys are based on a bigger master key
- Method 1:
  - MK: 0101000101000111110101001
  - SKs:0101000101000111110101001
- Method 2:
  - MK: 010100010100011
  - SKs:010100010100011

Example for private Key Block Cipher: Simple DES

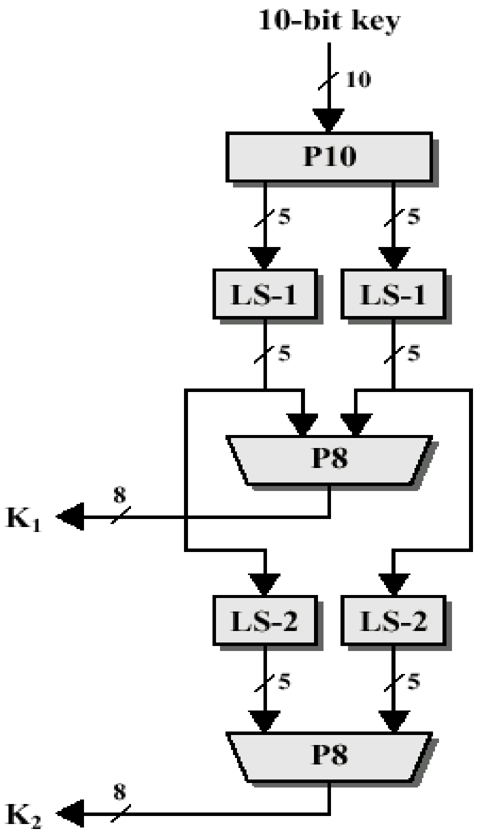
- An educational version of DES (Data Encryption Standard), the first widely used private key encryption algorithm:
  - 8 bit blocks and 10 bit keys
  - IP, IP^{-1} = (initial) permutation
  - P10 = 10 bit permutation
  - P8 = 8 bit permutation and selection.
  - SW = swap 2 halves



OLLSCOIL NA GAILLIMHE
UNIVERSITY OF GALWAY

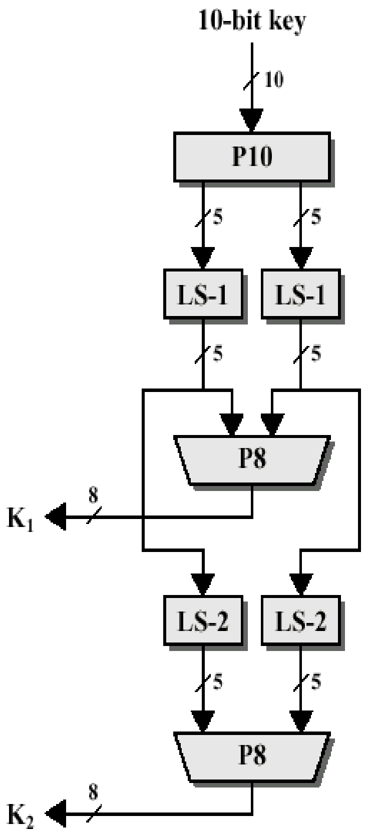
FYI: Simple DES – Key Generation

- P10: Permutation
  3 5 2 7 4 10 1 9 8 6
- LS-1: Left-shift 1
  Circular shift by 1 bit.
- P8: Permutation
  6 3 7 4 8 5 10 9
- LS-2: Left Shift 2
  Circular shift by 1 bit.
- P8: Permutation
  6 3 7 4 8 5 10 9



FYI: Example for Sub-Key Generation

- 10-bit key: 0110010110
- P10 permutation: 3 | 5 | 2 | 7 | 4 | 10 | 1 | 9 | 8 | 6
  10100 00111
- Circular left shift: 01001 01110
- P8 Permutation: 6 | 3 | 7 | 4 | 8 | 5 | 10 | 9
  K1: 00101101
- Circular left shift: 10010 11100
- P8 Permutation: 6 | 3 | 7 | 4 | 8 | 5 | 10 | 9
  K2: 10111000



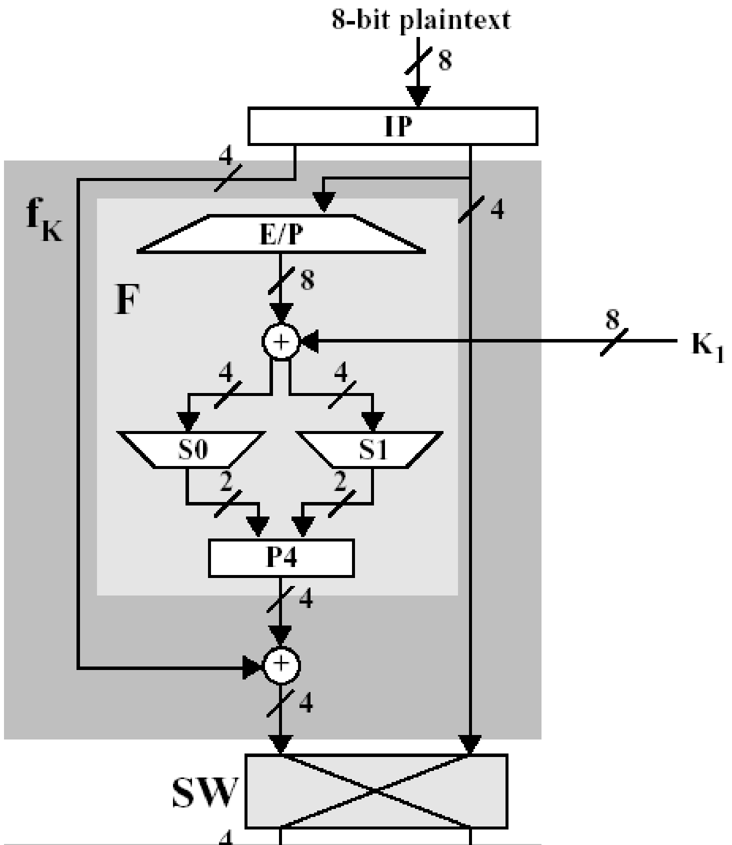
FYI: Structure of f_K

- E/P expansion permutation
  4 1 2 3 2 3 4 1
- 2 S-boxes S0 and S1

	0	**1**	**2**	**3**
0	1	0	3	2
1	3	2	1	0
2	0	2	1	3
3	3	1	3	2

	0	**1**	**2**	**3**
0	0	1	2	3
1	2	0	1	3
2	3	0	1	2
3	2	1	0	3

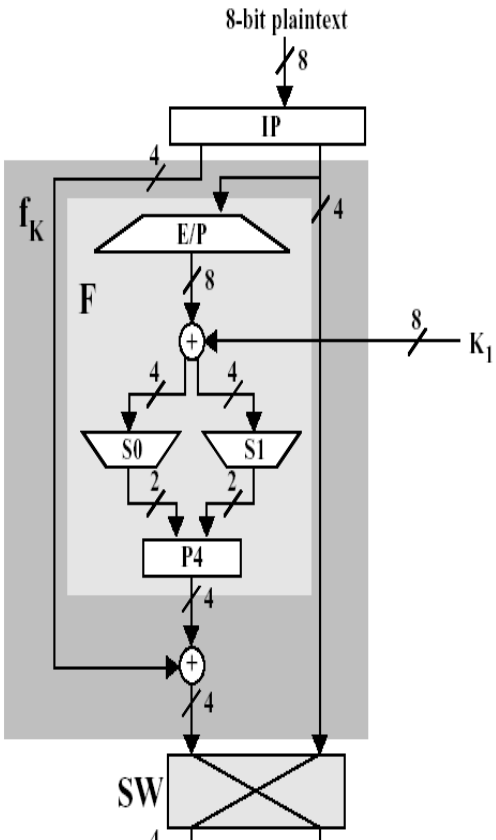
  The 1st and 4th input bits specify a row, the 2nd and 3rd input bits represent a column. The corresponding entry in a table represents the output
- P4 permutation 2 4 3 1



Ollscoil na Gaillimhe
University of Galway

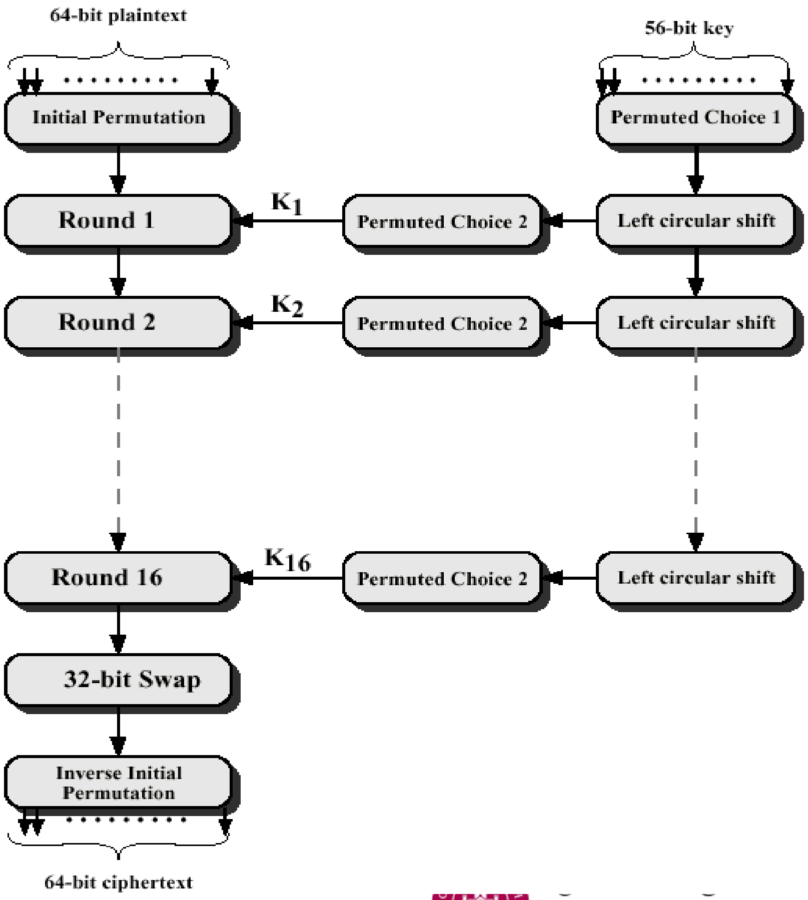
FYI: Example for f_K

- Input after IP: 01001101
- Left part: 0100
- E/P: 4|1|2|3|2|3|4|1
  11101011
- EX-OR K1: 00101101
  11000110
- S0 and S1: See previous page
  1011
- P4 permutation: 2|4|3|1
  0111
- EX-OR left part: 0100
  0011
- Concatenate right block: 1101
  00111101
- Swap: 11010011



DES

- 64 bit plain text
- 56 bit key and 48 bit sub-keys
- 16 rounds



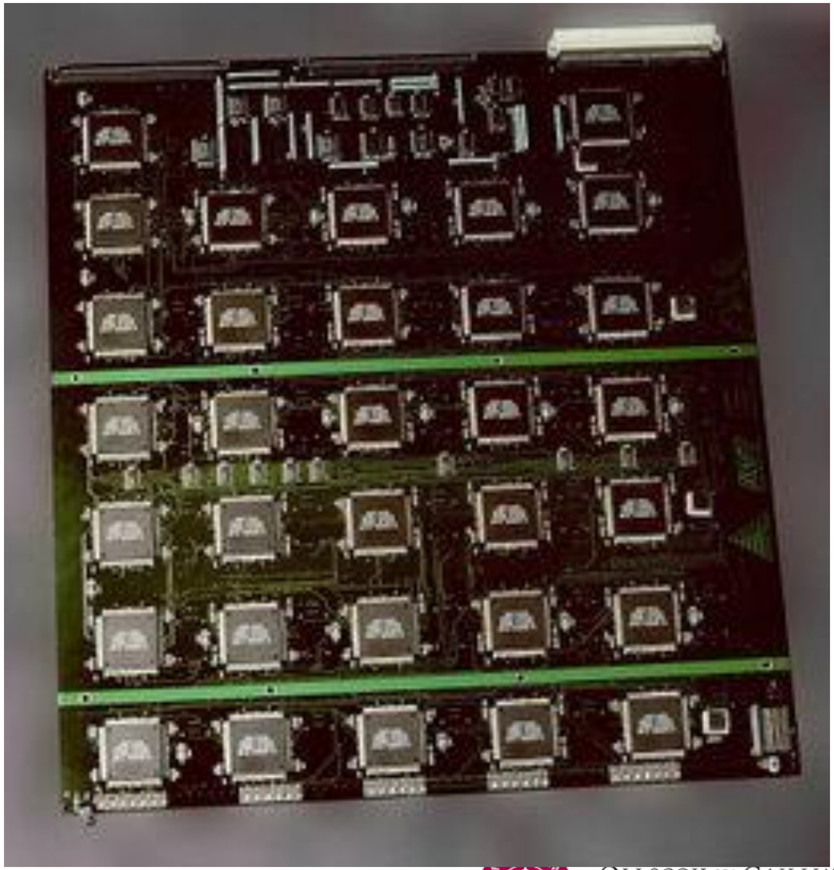
Strength of DES – Key Length?

- 56-bit keys have $2^{56} = 7.2 \times 10^{16}$ possible values
- Brute force search looks hard …
- But advances in 1990s have shown that it is possible:
  - In 1997 on Internet in a few months (using a PC cluster)
  - In 1998 on dedicated hardware in a few days
  - In 1999 above combined in 22 hrs!
- As a result, alternatives to DES had to be considered

Ollscoil na Gaillimhe
University of Galway

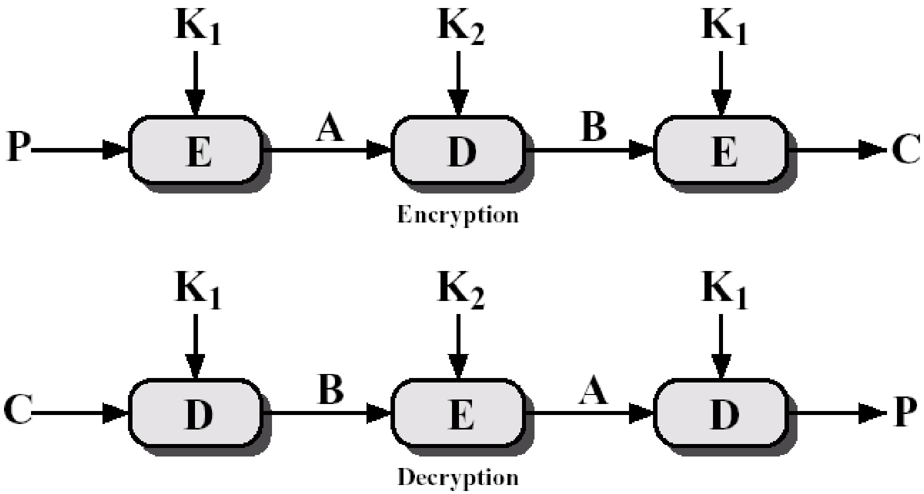
The DES Cracking Machine

- Developed by Electronic Frontier Foundation (EFF)
- Image shows a single circuit board.
- The entire machine consisted of 1,536 custom chips

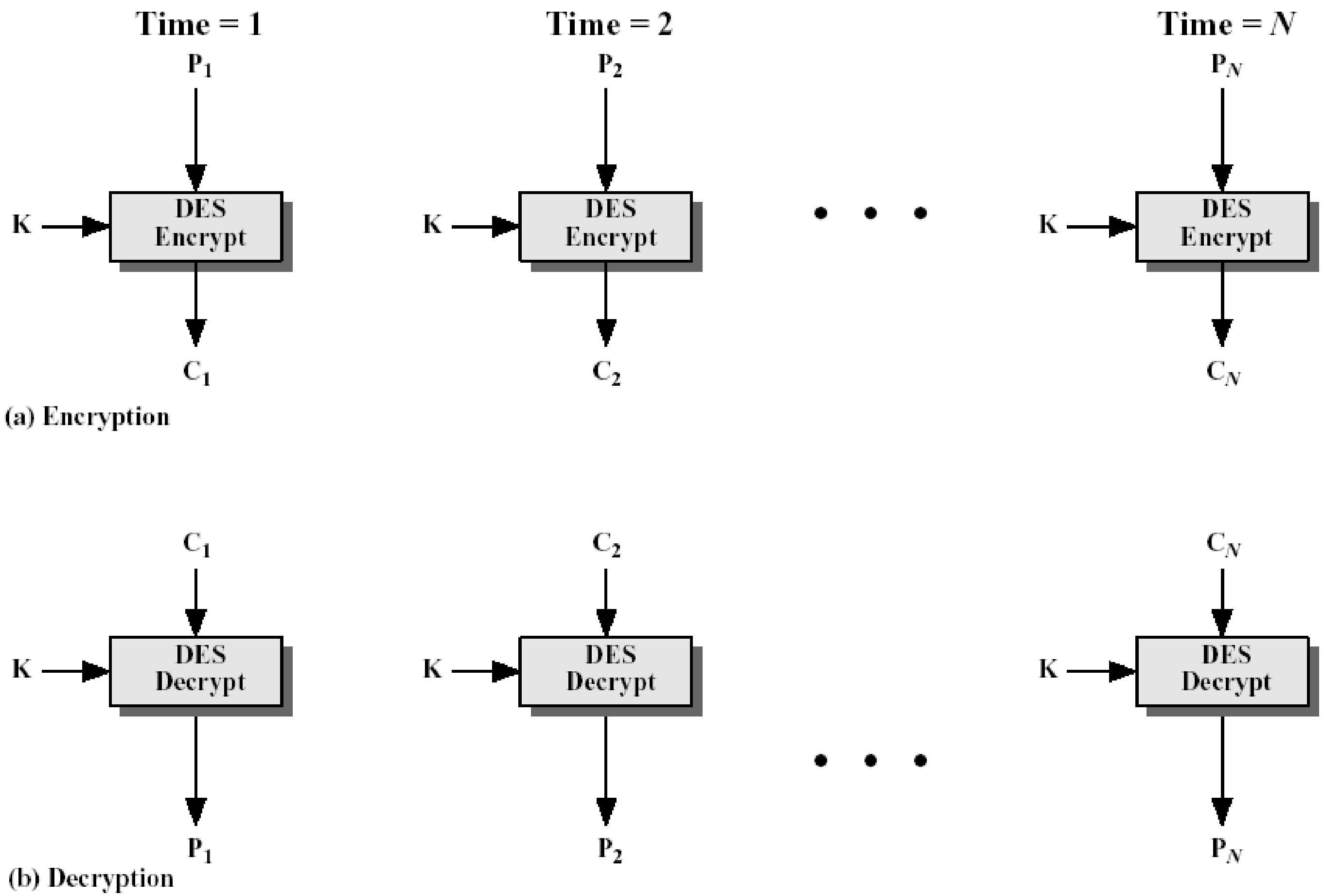


Triple DES

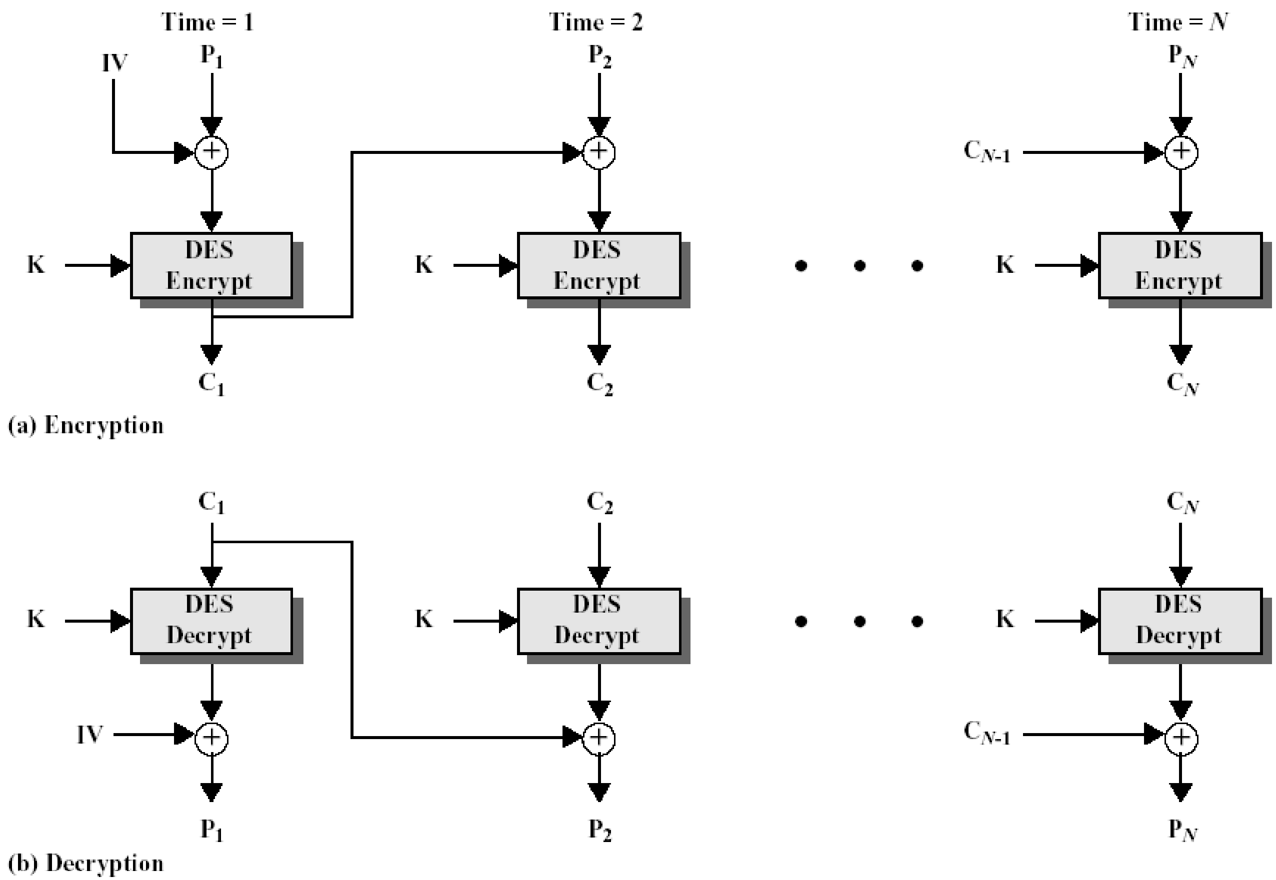
- Based on 2 (56-bit each) keys and three stages
- Symmetry preserved, therefore same concatenation is used for encoding and decoding



Modes of Operation: Electronic Codebook (EBC) Mode



Modes of Operation: Cipher Block Chaining (CBC) Mode



STREAM CIPHERS

Stream Ciphers

- So far we have examined block ciphers that process n-bytes at a time
- Stream ciphers in contrast process the message bit by bit (as a stream)
- They require a stream key K that is a pseudo-random sequence of 0s and 1s
- This bit-stream K is combined (EXORed) with the plaintext M bit by bit to generate the cipher text C:

$$C_i = M_i \text{ EXOR } K_i$$

- The randomness of the stream key completely destroys any statistically properties in the message
- The receiver generates the identical bit stream K and decodes the message C:

$$M_i = C_i \text{ EXOR } K_i$$

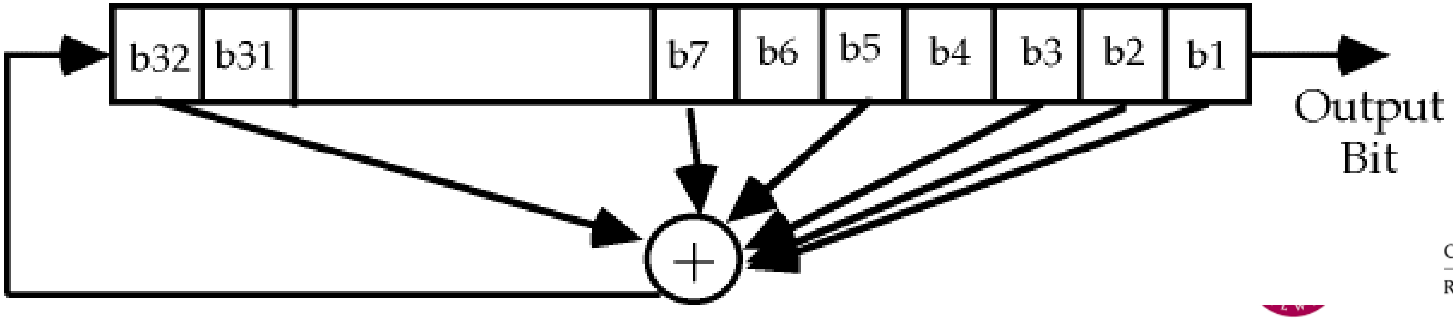
- Vernam Cipher or One-Time Pad is a famous stream cipher

Vernam Cipher

- Vernam cipher requires as many (random) key bits as message is long
  - Every message requires a new key, as reusing a stream key may allow an attacker to recover it!
- Such keys must be distributed securely between end-points
  - Very complicated, tedious and uneconomic, as a single stream key may consist of millions of bits
- For practical reasons stream ciphers based on pseudo-random generators (PRG) are used
  - PRGs are often based on Linear Feedback Shift Registers (LFSRs)
  - Only a seed value to initialise the PRG must be shared

Linear Feedback Shift Registers (LFSR)

- Consist a binary shift register of some length along with a linear feedback function that operates on some of those bits
- Each time a bit is needed, all bits are shifted right by one position
- The bit bumped out is the bit used as (pseudo-random) output from the LFSR
- A new bit is formed from the linear feedback function of some bits
- Correctly designed LFSRs generate a very long pseudo-random sequence before repeating
- LFSRs require an initialisation vector (i.e., seed) for their shift register



OIL NA GAILLIMHE
RSITY OF GALWAY

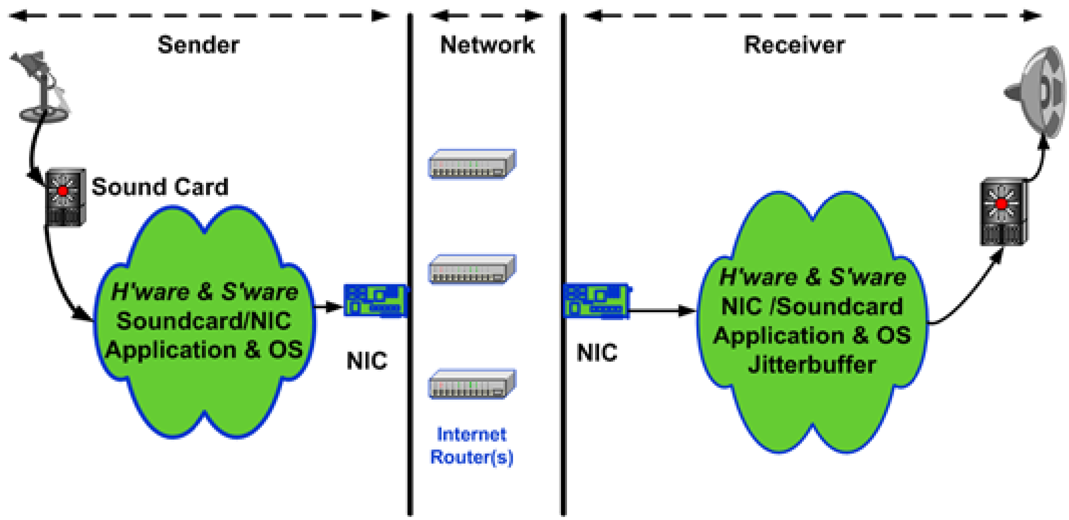
Example for an 8-Bit LFSR

- Initialisation vector: 10100110 (B_7 ••• B_0)
- Feedback Function: B_7 EXOR B_4 EXOR B_1
- Right shift after each cycle (B_0 shifted out)
- Iteration 0: 10100110
- Iteration 1: 01010011 >> 0
- Iteration 2: 00101001 >> 1
- Iteration 3: 00010100 >> 1
- Iteration 4: 10001010 >> 0
- … …

The feedback function returns a "1", if an odd number of inputs is set to "1"

Ollscoil na Gaillimhe
University of Galway

Example VoIP (Voice over the Internet Protocol)



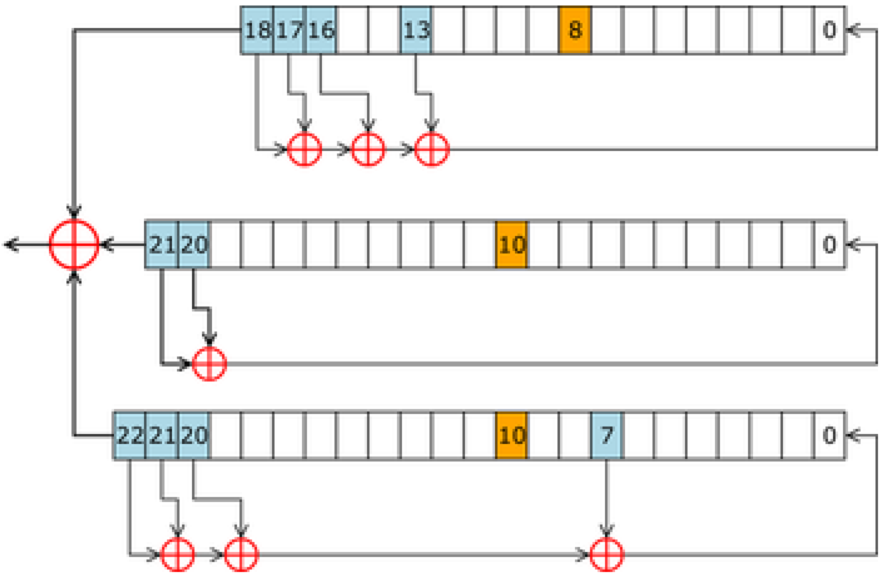
- The sender's voice is digitised and the resulting bit stream is encrypted using a stream cipher before being sent to the receiver over a network link
- Sender and receiver share the same seed value for their PRG

Stream Ciphers in Mobile Communication (early 2000s)

- Mobile phone conversations are sent as sequences of frames between both end points
  - Voice samples are collected and digitised by the mobile phone
- Every 4.6 milliseconds a 228-bits long frame consisting of digitised voice is processed and send out
- A5/1 is an LFSR-based algorithm that was used to produce 228 bits of key stream which is EXORed with the frame
- A5/1 is initialised using a 64-bit key

A5/1

- 3 independent LFSRs:
  - 19 bits
  - 22 bits
  - 23 bits
- The **majority bit** is the XORed output of all 3 LFSRs
- Each register is only shifted to the left, if their clocking bits (B8, B10, and B10 respectively) match the majority bit



A5/1

- A5/1 was originally introduced in 1987
- It was protected as a "trade secret“, but has subsequently been reverse engineered during the 90s
- As a result A5/2 was introduced, which has been broken as well
- A5/3 (KASUMI) was released in late 2002
  - Block-cipher based on Feistel network

RC4

- RC4 is a PRG designed by Ron Rivest of RSA Security in 1987
- RC4 was initially a trade secret, but 1994 a description of it was anonymously posted in the Internet
- It consists of a
  - key-scheduling algorithm (KSA) and a
  - pseudo-random generation algorithm (PRGA)

Ollscoil na Gaillimhe
University of Galway

RC4: The Key-Scheduling Algorithm (KSA)

- Requires a keyword (stored in `key[]`) with a specific `keylength`
- An 256 byte long permutation vector `S[]` is generated:

```
for i from 0 to 255
    S[i] := i;
j := 0;
for i from 0 to 255
    j := (j + S[i] + key[i mod keylength])
           mod 256;
    swap(S[i],S[j]);
```

RC4: The Pseudo-Random Generation Algorithm (PRGA)

- PRGA returns one byte at a time:

```
i := 0;
j := 0;
while GeneratingOutput:
    i := (i + 1) mod 256;
    j := (j + S[i]) mod 256;
    swap(S[i],S[j]);
    output S[(S[i] + S[j]) mod 256];
```

RC4

- Not an LFSR-based design, but rather a more general pseudo-random number generator design
- Can be efficiently implemented in software
- Broken and not used any more!

3 Security
- 3.1 Roos's biases and key reconstruction from permutation
- 3.2 Biased outputs of the RC4
- 3.3 Fluhrer, Mantin and Shamir attack
- 3.4 Klein's attack
- 3.5 Combinatorial problem
- 3.6 Royal Holloway attack
- 3.7 Bar-mitzvah attack
- 3.8 NOMORE attack