ECE 746: Lecture 1

Cryptographic Standard Contests

Advanced Encryption Standard Contest
Why a new standard?

1. Old standard insecure against brute-force attacks

2. Straightforward fixes lead to inefficient implementations
   - Triple DES

3. New trends in fast software encryption
   - Use of basic instructions of the microprocessor

4. New ways of assessing cipher strength
   - Differential cryptanalysis
   - Linear cryptanalysis

Why a contest?

- Focus the effort of cryptographic community
- Small number of specialists in the open research
- Stimulate the research on methods of constructing secure ciphers
- Avoid backdoor theories
- Speed-up the acceptance of the standard
External format of the AES algorithm

plaintext block

AES

128 bits

128, 192, 256 bits

ciphertext block

key

Rules of the contest

Each team submits

- Detailed cipher description
- Justification of design decisions
- Tentative results of cryptanalysis

Source code in C

Source code in Java

Test vectors
### AES Contest Effort

**June 1998**

15 Candidates
- USA
- Canada
- Belgium
- France
- Germany
- Norway
- UK
- Israel
- Korea
- Japan
- Australia
- Costa Rica

**Round 1**
- Security
- Software efficiency

**August 1999**

5 final candidates
- Mars
- RC6
- Rijndael
- Serpent
- Twofish

**Round 2**
- Security
- Hardware efficiency

**October 2000**

1 winner: Rijndael
- Belgium

### AES contest - First Round

**15 June 1998**
November 1997

- Deadline for submitting candidates
- 21 submissions,
- 15 fulfilled all requirements

**August 1998**

- 1st AES Conference in Ventura, CA
- Presentation of candidates

**March 1999**

- 2nd AES Conference in Rome, Italy
- Review of results of the First Round analysis

**August 1999**

- NIST announces five final candidates
### AES: Candidate algorithms

<table>
<thead>
<tr>
<th>North America (8)</th>
<th>Europe (4)</th>
<th>Asia (2)</th>
<th>First round June 1998-August 1999</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Canada:</strong></td>
<td><strong>Germany:</strong></td>
<td><strong>Korea:</strong></td>
<td><strong>Security</strong></td>
</tr>
<tr>
<td>CAST-256</td>
<td>Magenta</td>
<td>Crypton</td>
<td>Resistance to known attacks,</td>
</tr>
<tr>
<td>Deal</td>
<td></td>
<td></td>
<td>randomness tests</td>
</tr>
<tr>
<td><strong>USA:</strong></td>
<td><strong>Belgium:</strong></td>
<td><strong>Japan:</strong></td>
<td><strong>Software implementations</strong></td>
</tr>
<tr>
<td>Mars</td>
<td>Rijndael</td>
<td>E2</td>
<td>PC</td>
</tr>
<tr>
<td>RC6</td>
<td></td>
<td></td>
<td>Smart cards</td>
</tr>
<tr>
<td>Twofish</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Safer+</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>HPC</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Costa Rica:</strong></td>
<td><strong>France:</strong></td>
<td><strong>Australia (1)</strong></td>
<td></td>
</tr>
<tr>
<td>Frog</td>
<td>DFC</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Israel, UK, Norway:</strong></td>
<td><strong>Australia:</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Frog</td>
<td>Serpent</td>
<td>LOKI97</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Survey filled by 104 participants of the Second AES Conference in Rome, March 1999

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td>Rijndael</td>
</tr>
<tr>
<td>2.</td>
<td>RC6</td>
</tr>
<tr>
<td>3.</td>
<td>Twofish</td>
</tr>
<tr>
<td>4.</td>
<td>Mars</td>
</tr>
<tr>
<td>5.</td>
<td>Serpent</td>
</tr>
<tr>
<td>6.</td>
<td>E2</td>
</tr>
<tr>
<td>7.</td>
<td>CAST-256</td>
</tr>
<tr>
<td>8.</td>
<td>Safer+</td>
</tr>
<tr>
<td>9.</td>
<td>DFC</td>
</tr>
<tr>
<td>10.</td>
<td>Crypton</td>
</tr>
<tr>
<td>11.</td>
<td>DEAL</td>
</tr>
<tr>
<td>12.</td>
<td>HPC</td>
</tr>
<tr>
<td>13.</td>
<td>Magenta</td>
</tr>
<tr>
<td>14.</td>
<td>Loki97</td>
</tr>
<tr>
<td>15.</td>
<td>Frog</td>
</tr>
</tbody>
</table>

Overwhelming YES
Mild YES
Middle-of-the-Road
Mild NO
Overwhelming NO

AES Finalists (1)

USA

Mars - IBM
C. Burwick, D. Coppersmith, E. D’Avignon,
R. Gennaro, S. Halevi, C. Jutla, S. M. Matyas,
L. O’Connor, M. Peyravian, D. Safford,
N. Zunic

RC6 - RSA Data Security, Inc.
R. Rivest - MIT
M. Robshaw, R. Sidney, Y. L. Yin - RSA

Twofish - Counterpane Systems
B. Schneier, J. Kelsey, C. Hall, N. Ferguson
- Counterpane, D. Whiting - Hi fn,
D. Wagner - Berkeley
AES Finalists (2)

Europe

Rijndael - J. Daemen, V. Rijmen
Katholieke Universiteit Leuven
Belgium

E. Biham - Technion, Israel
L. Knudsen, University of Bergen, Norway

Second round  August 1999-August 2000

Security
Resistance to new attacks

Hardware implementations
FPGA
ASIC
### AES contest: Second Round

<table>
<thead>
<tr>
<th>Date</th>
<th>Event</th>
</tr>
</thead>
<tbody>
<tr>
<td>13-14 April 2000</td>
<td>3rd AES Conference in New York</td>
</tr>
<tr>
<td>15 May 2000</td>
<td>End of the comment period for Round II</td>
</tr>
<tr>
<td>2 October 2000</td>
<td>Winner announced</td>
</tr>
<tr>
<td>November 2001</td>
<td>FIPS-197: AES announced</td>
</tr>
<tr>
<td>May 2002</td>
<td>Standard becomes effective</td>
</tr>
</tbody>
</table>

### How NIST has made a final decision?

**BASIC CRITERIA**

- security
- software efficiency
- hardware efficiency
- flexibility
Security

Security: Theoretical attacks better than exhaustive key search

<table>
<thead>
<tr>
<th>Algorithm</th>
<th># of rounds in the attack/total # of rounds</th>
</tr>
</thead>
<tbody>
<tr>
<td>Serpent</td>
<td>9/23/32 without 16 mixing rounds</td>
</tr>
<tr>
<td>Twofish</td>
<td>6/10/16</td>
</tr>
<tr>
<td>Mars</td>
<td>11/5/16 without 16 mixing rounds</td>
</tr>
<tr>
<td>Rijndael</td>
<td>7/3/10</td>
</tr>
<tr>
<td>RC6</td>
<td>15/5/20</td>
</tr>
</tbody>
</table>
### Security: Theoretical attacks better than exhaustive key search

<table>
<thead>
<tr>
<th>Cipher</th>
<th>Team 1</th>
<th>Team 2</th>
<th>Ratio</th>
</tr>
</thead>
<tbody>
<tr>
<td>Serpent</td>
<td>28%</td>
<td>72%</td>
<td>28%</td>
</tr>
<tr>
<td>Twofish</td>
<td>38%</td>
<td>62%</td>
<td>38%</td>
</tr>
<tr>
<td>Mars</td>
<td>69%</td>
<td>31%</td>
<td>69%</td>
</tr>
<tr>
<td>Rijndael</td>
<td>70%</td>
<td>30%</td>
<td>70%</td>
</tr>
<tr>
<td>RC6</td>
<td>75%</td>
<td>25%</td>
<td>75%</td>
</tr>
</tbody>
</table>

# of rounds in the attack/total # of rounds - 100%

### Security: Authors of attacks

**Team**
- **Twofish**: Kelsey, Kohno, Schneier, Ferguson, Stay, Wagner, Whiting
- **Serpent**: Knudsen, Meier
- **Other groups**: Lucks, U. Mannheim, Gilbert, Minier, France Telecom, Gilbert, Handschuh, Joux, Vaudenay, France Telecom

**Attacked cipher**
- MARS
- Serpent
- Rijndael
- RC6
- Twofish
Efficiency - What’s more important: software or hardware?
Historical view

<table>
<thead>
<tr>
<th>Time</th>
<th>Secret-key ciphers</th>
<th>Hash functions</th>
</tr>
</thead>
<tbody>
<tr>
<td>1970</td>
<td>DES – optimized for <strong>hardware</strong></td>
<td>DES-based hash functions – optimized for <strong>hardware</strong></td>
</tr>
<tr>
<td>1980</td>
<td>Fast Software Encryption: ciphers optimized for <strong>software</strong>: e.g., RC5, Blowfish, RC4</td>
<td>MD4-family optimized primarily for <strong>software</strong></td>
</tr>
<tr>
<td>1990</td>
<td>AES – optimized for <strong>software and hardware</strong></td>
<td></td>
</tr>
<tr>
<td>2000</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Software or hardware?**

**SOFTWARE**
- **security of data during transmission**
- **flexibility** (new cryptoalgorithms, protection against new attacks)
- **speed**
- **low cost**

**HARDWARE**
- **random key generation**
- **access control to keys**
- **tamper resistance** (viruses, internal attacks)
- **low cost**
Efficiency indicators

Primary efficiency indicators

Software
- Speed
- Memory

Hardware
- Speed
- Area
  - Power consumption
Efficiency parameters

Latency

Throughput = Speed

Encryption/decryption

\[ M_i \]

Time to encrypt/decrypt a single block of data

\[ C_i \]

Encryption/decryption

\[ M_{i+2} \]
\[ M_{i+1} \]
\[ M_i \]

\[ C_{i+2} \]
\[ C_{i+1} \]
\[ C_i \]

Number of bits encrypted/decrypted in a unit of time

Throughput = \frac{\text{Block size} \cdot \text{Number of blocks processed simultaneously}}{\text{Latency}}
Efficiency in software: Code submitted by authors
200 MHz Pentium Pro, Borland C++

Mean encryption & decryption speed in software
Code developed by Brian Gladman, UK
200 MHz Pentium Pro, Microsoft Visual C++, ver. 6
**Efficiency in software: NIST tests**

*450 MHz Pentium II, DJGPP gcc*

<table>
<thead>
<tr>
<th>Speed [Mbits/s]</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mars</td>
</tr>
<tr>
<td>128-bit key</td>
</tr>
</tbody>
</table>

**Encryption time in clock cycles on various platforms**

*Twofish team: Bruce Schneier & Doug Whiting*

better
### Efficiency in software: Ranking of encryption speeds for various platforms

<table>
<thead>
<tr>
<th></th>
<th>Intel</th>
<th>Alpha</th>
<th>Sun-Sparc</th>
<th>H-P</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mars</td>
<td>4 4 2 4 3 2</td>
<td>3 4 4 3 3</td>
<td>3 3</td>
<td></td>
</tr>
<tr>
<td>RC6</td>
<td>1 3 1 1 4 1</td>
<td>4 3 3 5 4</td>
<td>4</td>
<td></td>
</tr>
<tr>
<td>Twofish</td>
<td>2 1 3 2 1 4</td>
<td>2 2 2 2 2</td>
<td>2 2</td>
<td></td>
</tr>
<tr>
<td>Rijndael</td>
<td>3 2 4 3 2 3</td>
<td>1 1 1 1 1</td>
<td>1 1</td>
<td></td>
</tr>
<tr>
<td>Serpent</td>
<td>5 5 5 5 5 5</td>
<td>5 5 5 4 5</td>
<td>5</td>
<td></td>
</tr>
</tbody>
</table>

### NIST Report: Software Efficiency

#### Encryption and Decryption Speed

<table>
<thead>
<tr>
<th></th>
<th>32-bit processors</th>
<th>64-bit processors</th>
<th>DSPs</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>high</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>RC6</td>
<td></td>
<td>Rijndael Twofish</td>
<td></td>
</tr>
<tr>
<td><strong>medium</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Rijndael</td>
<td>Mars Twofish</td>
<td>Mars RC6</td>
<td></td>
</tr>
<tr>
<td><strong>low</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Serpent</td>
<td></td>
<td>Serpent</td>
<td></td>
</tr>
</tbody>
</table>
NIST Report: Software Efficiency
Encryption and decryption speed in software on smart cards

8-bit processors | 32-bit processors
--- | ---
high | Rijndael
medium | RC6, Mars, Twofish
low | Serpent

Efficiency in software: Key setup, author codes
200 MHz Pentium Pro, Borland C++

Time [clock cycles]
- Rijndael
- RC6
- Mars
- Serpent
- Twofish

- 128-bit key
- 192-bit key
- 256-bit key

better
Key set-up time in software
Code developed by Brian Gladman, UK

200 MHz Pentium Pro, Microsoft Visual C++, ver. 6

NIST Report: Software Efficiency
Key scheduling

<table>
<thead>
<tr>
<th></th>
<th>32-bit processors</th>
<th>64-bit processors</th>
<th>DSPs</th>
</tr>
</thead>
<tbody>
<tr>
<td>high</td>
<td>Rijndael</td>
<td>Rijndael</td>
<td>Rijndael Serpent</td>
</tr>
<tr>
<td>medium</td>
<td>Mars RC6</td>
<td>RC6 Serpent</td>
<td>Mars RC6</td>
</tr>
<tr>
<td>low</td>
<td>Serpent Twofish</td>
<td>Mars Twofish</td>
<td>Twofish</td>
</tr>
</tbody>
</table>
NIST Report: Software Efficiency
Key scheduling on smart cards

<table>
<thead>
<tr>
<th>8-bit processors</th>
<th>high</th>
<th>medium</th>
<th>low</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Rijndael</td>
<td>Mars</td>
<td>RC6</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Twofish</td>
<td>Serpent</td>
</tr>
</tbody>
</table>

Efficiency in software

Strong dependence on:

1. Instruction set architecture (e.g., variable rotations)
2. Programming language (assembler, C, Java)
3. Compiler
4. Programming style
**Efficiency in software: Conclusions**

**Encryption/decryption**
- Strong variation of results
- Serpent the worst for majority of platforms

**Key setup**
- Moderate variation of results
- Rijndael and RC6 the best for majority of platforms
- Twofish and Serpent the worst for majority of platforms

**Efficiency in hardware**
Primary ways of implementing cryptography in hardware

<table>
<thead>
<tr>
<th>ASIC</th>
<th>FPGA</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Application Specific Integrated Circuit</strong></td>
<td><strong>Field Programmable Gate Array</strong></td>
</tr>
<tr>
<td>• designs must be sent for expensive and time consuming fabrication in semiconductor foundry</td>
<td>• bought off the shelf and reconfigured by designers themselves</td>
</tr>
<tr>
<td>• designed all the way from behavioral description to physical layout</td>
<td>• no physical layout design; design ends with a bitstream used to configure a device</td>
</tr>
</tbody>
</table>

Which way to go?

<table>
<thead>
<tr>
<th>ASICs</th>
<th>FPGAs</th>
</tr>
</thead>
<tbody>
<tr>
<td>High performance</td>
<td>Off-the-shelf</td>
</tr>
<tr>
<td>Low power</td>
<td>Low development costs</td>
</tr>
<tr>
<td>Low cost (but only in high volumes)</td>
<td>Short time to the market</td>
</tr>
<tr>
<td>Reconfigurability</td>
<td></td>
</tr>
</tbody>
</table>
Capabilities of reconfiguration (1)

External ROM and microprocessor enables changing the FPGA function in several milliseconds.

Various algorithms

- FPGA: Triple DES
  - 5-15 ms

- FPGA: AES
  - 5-15 ms

- FPGA: Skipjack

Top level block diagram

- Control unit
- Input interface
- Encryption/decryption
- Key scheduling
- Memory of internal keys
- Output interface
- Output
Typical Flow Diagram of a Secret-Key Block Cipher

Initial transformation

i:=1

Cipher Round

i:=i+1

i<#rounds?

Final transformation

Round Key[0]

Round Key[i]

Round Key[#rounds+1]

#rounds times
Basic iterative architecture

Efficiency in hardware: FPGA Virtex 1000: Speed

Throughput [Mbit/s]
Efficiency in hardware: FPGA Virtex 1000: Area

Area [CLB slices]

- George Mason University
- University of Southern California
- Worcester Polytechnic Institute

ASIC implementations: NSA group

- 128-bit key scheduling
- 3-in-1 (128, 192, 256 bit) key scheduling

128-bit key scheduling:
- Rijndael: 606
- Serpent I: 443

3-in-1 (128, 192, 256 bit) key scheduling:
- Rijndael: 202
- Serpent I: 202
- Twofish: 105
- RC6: 103
- Mars: 57
NIST Report + GMU Report: Hardware Efficiency
Feedback cipher modes: CBC, CFB

<table>
<thead>
<tr>
<th>Speed</th>
<th>Small</th>
<th>Medium</th>
<th>Large</th>
</tr>
</thead>
<tbody>
<tr>
<td>High</td>
<td>Rijndael</td>
<td>Serpent</td>
<td></td>
</tr>
<tr>
<td>Medium</td>
<td>Twofish</td>
<td>RC6</td>
<td></td>
</tr>
<tr>
<td>Low</td>
<td>MARS</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Feedback cipher modes - CBC

\[ C_i = AES(M_i \oplus IV) \]
\[ C_i = AES(M_i \oplus C_{i-1}) \quad \text{for } i=2..N \]
Non-feedback Counter Mode - CTR

\[ C_i = M_i \oplus AES(IV+i) \quad \text{for } i=0..N \]

Increasing speed by parallel processing
Increasing speed using pipelining

Speed = \frac{\text{block size}}{\text{target_clock_period}}

Pipelined operation of the encryption unit
Encryption in non-feedback modes (ECB, counter) decryption in all modes

Assuming clock period = 50 MHz

Area [CLB slices]  Speed [Mbit/s]

Rijndael  6.4 Gbit/s
Serpent  RC6  Mars
Twofish

NIST Report + GMU Report: Hardware Efficiency
Non-feedback cipher modes: ECB, CTR

Speed
High  Rijndael
       Serpent
       Twofish
       RC6
Mars
Medium
Low

Area
Small  Medium  Large
Flexibility

Flexibility: Criteria

- Additional key-sizes and block-sizes
- Ability to function efficiently and securely in a wide variety of platforms and applications
  - low-end smartcards, wireless - memory requirements
  - IPSec, ATM - key setup time in hardware
  - B-ISDN, satellite communication - encryption speed
Survey filled by 167 participants of the Third AES Conference, April 2000

Ranking by participants of the AES3 Conference

Positive votes – negative votes
### Leading candidates (1)

**Rijndael**

<table>
<thead>
<tr>
<th>+</th>
<th>–</th>
</tr>
</thead>
<tbody>
<tr>
<td>• fastest in hardware</td>
<td>• security margin</td>
</tr>
<tr>
<td>• close to the fastest in software</td>
<td></td>
</tr>
<tr>
<td>• very high flexibility</td>
<td></td>
</tr>
</tbody>
</table>

**novel ideas**

### Leading candidates (2)

**Serpent**

<table>
<thead>
<tr>
<th>+</th>
<th>–</th>
</tr>
</thead>
<tbody>
<tr>
<td>• large security margin</td>
<td>• slow in software</td>
</tr>
<tr>
<td>• conservative construction</td>
<td>• moderate flexibility</td>
</tr>
<tr>
<td>• very fast in hardware</td>
<td></td>
</tr>
<tr>
<td>• cryptanalytical reputation of authors</td>
<td></td>
</tr>
</tbody>
</table>
## Leading candidates (3)

### Twofish

<table>
<thead>
<tr>
<th>+</th>
</tr>
</thead>
<tbody>
<tr>
<td>• good security margin</td>
</tr>
<tr>
<td>• fast encryption/decryption in software</td>
</tr>
<tr>
<td>• American</td>
</tr>
<tr>
<td>• strongly advertised</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>–</th>
</tr>
</thead>
<tbody>
<tr>
<td>• moderately fast in hardware</td>
</tr>
<tr>
<td>• slow key setup in software</td>
</tr>
<tr>
<td>• moderate flexibility</td>
</tr>
</tbody>
</table>

### Major operations
### Major operations of AES finalists

<table>
<thead>
<tr>
<th></th>
<th>Serpent</th>
<th>Rijndael</th>
<th>Twofish</th>
<th>RC6</th>
<th>Mars</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>S-boxes</strong></td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td></td>
</tr>
<tr>
<td><strong>Multiplication</strong></td>
<td></td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td></td>
</tr>
<tr>
<td>in GF($2^m$)</td>
<td></td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td></td>
</tr>
<tr>
<td><strong>Integer multiplication</strong></td>
<td></td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td></td>
</tr>
<tr>
<td><strong>Variable rotation</strong></td>
<td></td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td></td>
</tr>
</tbody>
</table>

### Auxiliary operations of AES finalists

<table>
<thead>
<tr>
<th></th>
<th>Serpent</th>
<th>Twofish</th>
<th>Rijndael</th>
<th>RC6</th>
<th>Mars</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Boolean</strong></td>
<td>✔</td>
<td>✔</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Fixed rotation</strong></td>
<td>✔</td>
<td>✔</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Addition/subtraction</strong></td>
<td></td>
<td>✔</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Permutation</strong></td>
<td>✔</td>
<td>✔</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
### Types of ciphers

#### AES: Types of candidate algorithms

<table>
<thead>
<tr>
<th>Feistel Networks</th>
<th>Modified Feistel Network</th>
</tr>
</thead>
<tbody>
<tr>
<td>Twofish</td>
<td>Deal</td>
</tr>
<tr>
<td>E2</td>
<td>LOKI97</td>
</tr>
<tr>
<td>DFC</td>
<td>Magenta</td>
</tr>
<tr>
<td>Rijndael</td>
<td>Safer+</td>
</tr>
<tr>
<td>Serpent</td>
<td>Crypton</td>
</tr>
<tr>
<td></td>
<td>RC6</td>
</tr>
<tr>
<td></td>
<td>MARS</td>
</tr>
<tr>
<td></td>
<td>CAST-256</td>
</tr>
</tbody>
</table>

#### Others
- Frog
- HPC
**Feistel Network:** Single Round of Twofish

$$D'[3] \quad D'[2] \quad D'[1] \quad D'[0]$$

$$D[3] \quad D[2] \quad D[1] \quad D[0]$$

- Units shared between encryption and decryption

**Modified Feistel Network:** Single Round of MARS

$$D'[3] \quad D'[2] \quad D'[1] \quad D'[0]$$

$$D[3] \quad D[2] \quad D[1] \quad D[0]$$

- Units shared between encryption and decryption

$k = K[4+2i]$, $k' = K[5+2i]$, $i$ - round no.
Substitution-Linear Transformation Network: Single Round of Serpent

- units shared between encryption and decryption

Substitution-Linear Transformation Network: Serpent in Hardware
Substitution-Linear Transformation Network:

Rijndael in Hardware

- units shared between encryption and decryption

Inversion in GF($2^8$)

- affine transformation
- ShiftRow
- MixColumn
- InvShiftRow
- InvMixColumn

Number and complexity of rounds
Number vs. complexity of a round

Number of rounds

Complexity of a round

 Modes of operation
### Secret-key cryptography standards

<table>
<thead>
<tr>
<th>Federal standards</th>
<th>Banking standards</th>
<th>International standards</th>
</tr>
</thead>
<tbody>
<tr>
<td>NIST</td>
<td>ANSI</td>
<td>ISO</td>
</tr>
</tbody>
</table>

- **FIPS 46-1 DES**
  - FIPS 46-2 DES
- **FIPS 81 Modes of operation**
  - X3.106 DES modes of operation
- **FIPS 46-3 Triple DES**
- **AES**
- **X3.92** DES
- **ISO 8732** Modes of operation of a 64-bit cipher
- **X9.52** Modes of operation of Triple DES
- **ISO 10116** Modes of operation of an n-bit cipher

### Operating Modes Contest

- **4 Old Modes** (CBC, CFB, OFB, ECB)
  - **April 2001**
    - 10 New Candidates from Egypt, Estonia, Norway, Sweden, Thailand, USA
  - **Counter mode**
  - **Summer 2001**
    - 5 Standard Modes
  - **2002**
    - New Standard Modes
<table>
<thead>
<tr>
<th>Modes submitted to the contest (1)</th>
<th>Full name</th>
<th>Authors</th>
<th>Institution</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>2DEM</strong> 2D-Encryption Mode</td>
<td>A. A. Belal, M. A. Abdel-Gawad</td>
<td>Alexandria University, <strong>Egypt</strong></td>
<td></td>
</tr>
<tr>
<td><strong>ABC</strong> Accumulated Block Chaining</td>
<td>L. Knudsen</td>
<td>U. of Bergen <strong>Norway</strong></td>
<td></td>
</tr>
<tr>
<td><strong>CTR</strong> Counter Mode</td>
<td>H. Lipmaa, P. Rogaway, D. Wagner</td>
<td><strong>Finland, Estonia, USA, Thailand</strong></td>
<td></td>
</tr>
<tr>
<td><strong>IACBC</strong> Integrity Aware CBC</td>
<td>C. Jutla</td>
<td>IBM, <strong>USA</strong></td>
<td></td>
</tr>
<tr>
<td><strong>IAPM</strong> Integrity Aware Parallalizable Mode</td>
<td>C. Jutla</td>
<td>IBM, <strong>USA</strong></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Modes submitted to the contest (2)</th>
<th>Full name</th>
<th>Authors</th>
<th>Institution</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>IGE</strong> Infinite Garble Extension</td>
<td>V. D. Gligor, P. Donescu</td>
<td>VDG, Inc., <strong>USA</strong></td>
<td></td>
</tr>
<tr>
<td><strong>KFB</strong> Key Feedback Mode</td>
<td>J. Hästad, M. Naslund</td>
<td>NADA, Ericsson <strong>Sweden</strong></td>
<td></td>
</tr>
<tr>
<td><strong>OCB</strong> Offset Codebook</td>
<td>P. Rogaway</td>
<td>UCSD, <strong>USA, Thailand</strong></td>
<td></td>
</tr>
<tr>
<td><strong>PCFB</strong> Propagating Cipher Feedback</td>
<td>H. Hellström</td>
<td>StreamSec, <strong>Sweden</strong></td>
<td></td>
</tr>
<tr>
<td><strong>XCBC</strong> eXtended CBC Encryption</td>
<td>V. D. Gligor, P. Donescu</td>
<td>VDG, Inc., <strong>USA</strong></td>
<td></td>
</tr>
</tbody>
</table>
Evaluation Criteria for Modes of Operation

- Security
- Efficiency
- Functionality

Evaluation criteria (1)

Security

- resistance to attacks
- proof of security
- random properties of the ciphertext

Efficiency

- number of calls of the block cipher
- capability for parallel processing
- memory/area requirements
- initialization time
- capability for preprocessing
Evaluation criteria (2)

Functionality

- **security services**
  - confidentiality, integrity, authentication
- flexibility
  - variable lengths of blocks and keys
  - different amount of precomputations
  - requirements on the length of the message
- **vulnerability to implementation errors**
  - requirements on the amount of keys, initialization vectors, random numbers, etc.
  - error propagation and the capability for resynchronization
  - patent restrictions

Modes of operation: Current standard - CBC

Problems:
- No parallel processing of blocks from the same packet
- No speed-up by preprocessing
- No integrity or authentication
Counter mode

Features:
+ Potential for parallel processing
+ Speed-up by preprocessing
- No integrity or authentication

Properties of existing and new cipher modes

<table>
<thead>
<tr>
<th></th>
<th>CBC</th>
<th>CFB</th>
<th>OFB</th>
<th>New standard</th>
</tr>
</thead>
<tbody>
<tr>
<td>Proof of security</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Parallel processing</td>
<td></td>
<td></td>
<td>decryption only</td>
<td>✓</td>
</tr>
<tr>
<td>Preprocessing</td>
<td>–</td>
<td>–</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Integrity and</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>✓</td>
</tr>
<tr>
<td>authentication</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Resistance</td>
<td>✓</td>
<td>✓</td>
<td>–</td>
<td>✓</td>
</tr>
<tr>
<td>to implementation</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>errors</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
## Encryption with authentication

<table>
<thead>
<tr>
<th></th>
<th>Full name</th>
<th>Authors</th>
<th>Institutions</th>
</tr>
</thead>
<tbody>
<tr>
<td>IACBC</td>
<td>Integrity Aware CBC</td>
<td>C. Jutla</td>
<td>IBM (patent)</td>
</tr>
<tr>
<td>IAPM</td>
<td>Integrity Aware Parallelizable Mode</td>
<td>C. Jutla</td>
<td>IBM (patent)</td>
</tr>
<tr>
<td>XCBC-XOR</td>
<td>eXtended CBC Encryption</td>
<td>V. D. Gligor, P. Donescu</td>
<td>VDG, Inc., (patent)</td>
</tr>
<tr>
<td>XECB-XOR</td>
<td>eXtended ECB Encryption</td>
<td>V. D. Gligor, P. Donescu</td>
<td>VDG, Inc., (patent)</td>
</tr>
<tr>
<td>OCB</td>
<td>Offset Codebook</td>
<td>P. Rogaway</td>
<td>UCSD, USA, Thailand</td>
</tr>
</tbody>
</table>

**OCB**

$$\begin{align*}
\text{IV} & \quad 0 \quad M_1 \quad M_2 \quad \ldots \quad M_{N-1} \quad M_N \\
E & \quad E \quad E \quad \ldots \quad E \quad E \quad E \\
L & \quad \bigoplus \quad \bigoplus \quad \bigoplus \quad \bigoplus \quad \bigoplus \quad \bigoplus \\
E & \quad \bigoplus \quad \bigoplus \quad \bigoplus \quad \bigoplus \quad \bigoplus \quad \bigoplus \\
\text{Z}_1 & \quad \text{Z}_2 \quad \text{Z}_{N-1} \quad \text{Z}_N \\
\text{R} & \quad C_1 \quad C_2 \quad \ldots \quad C_{N-1} \quad C_N \\
\text{Z}_1 &= f(L, R) \\
\text{Control sum} & \quad \text{length} \quad g(L) \quad \text{Z}_N \quad \text{t bits} \\
\end{align*}$$
Features of two new modes adopted by NIST

<table>
<thead>
<tr>
<th></th>
<th>CBC</th>
<th>CFB</th>
<th>OFB</th>
<th>CTR</th>
<th>CCM</th>
<th>GCM</th>
</tr>
</thead>
<tbody>
<tr>
<td>Proof of security</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Parallel processing</td>
<td></td>
<td></td>
<td></td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td></td>
<td>only decryption</td>
<td>–</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Preprocessing</td>
<td>–</td>
<td>–</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Integritly and authentication</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Resistance to implementation errors</td>
<td>✓</td>
<td>✓</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>–</td>
</tr>
</tbody>
</table>

Modes of Operation of Block Ciphers

Timeline

<table>
<thead>
<tr>
<th>NIST Standards:</th>
<th>Dec 2001</th>
<th>May 2004</th>
</tr>
</thead>
<tbody>
<tr>
<td>FIPS 81 (DES only)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CTR (counter mode)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CBC, CFB, OFB, ECB</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CCM</td>
<td></td>
<td></td>
</tr>
<tr>
<td>SP 800-38A</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Contests:

Apr 2001

10 modes submitted to the contest (e.g., CTR, OCB, IACBC, IAPM)

Problems with patent issues.

Attacks:

Aug 2001

DCM – mode developed by NSA
broken a few days after publication
Other contests

<table>
<thead>
<tr>
<th>Europe</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>NESSIE Project</strong></td>
</tr>
<tr>
<td><em>New European Schemes for Signatures, Integrity, and Encryption</em></td>
</tr>
<tr>
<td>2000-2002</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Japan</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>CRYPTREC Project</strong></td>
</tr>
<tr>
<td>2000-2002</td>
</tr>
</tbody>
</table>
NESSIE, CRYPTREC

Multiple types of transformations:
- Symmetric-key block ciphers
- Stream ciphers
- Hash functions
- MACs
- Asymmetric encryption schemes
- Asymmetric digital signature schemes
- Asymmetric identification schemes

Development of methodology of a **fair evaluation and comparison** of algorithms belonging to the same class.
### NESSIE: Winners of the contest (1)
#### Symmetric block ciphers
128-bit input block

<table>
<thead>
<tr>
<th>Name</th>
<th>Institution</th>
<th>Problems</th>
</tr>
</thead>
<tbody>
<tr>
<td>AES</td>
<td>NIST standard</td>
<td></td>
</tr>
<tr>
<td>Camellia</td>
<td>NTT</td>
<td></td>
</tr>
</tbody>
</table>

#### Participants of the second round:
1. RC6
2. Safer++

### NESSIE: Winners of the contest (2)
#### Symmetric block ciphers
64-bit input block

<table>
<thead>
<tr>
<th>Name</th>
<th>Institution</th>
<th>Problems</th>
</tr>
</thead>
<tbody>
<tr>
<td>MISTY1</td>
<td>Mitsubishi</td>
<td></td>
</tr>
</tbody>
</table>

#### Participants of the second round:
1. IDEA
2. Khazad
3. Safer++

#### Triple DES
- NIST standard
- Speed (software)

### Participants of the second round:
1. IDEA
2. Khazad
3. Safer++

#### Problems:
- Patent, key scheduling
- Structural symmetry
- Structural properties, speed
NESSIE: Winners of the contest (3)
Symmetric block ciphers
256-bit input block

1. SHACAL-2
   Gemplus

Participants of the second round:
1. RC6
   RSA Security
   patent issues

NESSIE: Winners of the contest - 2002
Stream ciphers and pseudorandom number generators

All analyzed ciphers rejected because of the detected weaknesses
**eSTREAM - Contest for a new stream cipher standard, 2004-2008**

**PROFILE 1**
- Stream cipher suitable for **software implementations** optimized for **high speed**
- Key size - 128 bits
- Initialization vector – 64 bits or 128 bits

**PROFILE 2**
- Stream cipher suitable for **hardware implementations** with **limited memory, number of gates, or power supply**
- Key size - 80 bits
- Initialization vector – 32 bits or 64 bits

**eSTREAM - Contest for a new stream cipher standard, 2004-2008**

**PROFILE 1A**
- PROFILE 1 + authentication mechanism
- Message authentication code should allow lengths 32, 64, 96 and 128 bits

**PROFILE 2A**
- PROFILE 2 + authentication mechanism
- Message authentication code should allow lengths 32 and 64 bits
eSTREAM - Contest for a new stream cipher standard, 2004-2008

Schedule of the contest

- November 2004: Request for proposals
- 29 April 2005: Deadline for submissions
  - 34 ciphers, 23 candidates for PROFILE 1
  - 26 candidates for PROFILE 2
- 26-27 May 2005: Workshop in Aarhus, Denmark
- February 2006: Workshop in Leuven, Belgium
- July 2006: Beginning of Phase II
- Jan-Feb 2007: Workshop in Bochum, Germany
- April 2007: Beginning of Phase III
- February 2008: Workshop in Lausanne, Switzerland
- May 2008: Final report

GMU Hardware Implementation Results: Throughput vs. Area
FPGA: Xilinx Spartan 3 family
First conclusions from January 2007

• **Very large differences** among candidate ciphers
  (much larger than for five final candidates in the AES contest)

Possible reasons:
• variety of ciphers based on different design principles
• different internal state, key, and IV sizes
• early stage of the contest

**Trivium and Grain** outperform other eSTREAM ciphers
in terms of
• flexibility
• minimum area
• maximum throughput to area ratio.

Once again ciphers based on LFSR and NFSRs show their
superiority in hardware implementations

Security analysis should **focus first on the most efficient ciphers**