Design Tradeoffs of the AES Candidates

Eli Biham

Computer Science Department
Technion, Haifa 32000, Israel

October 20, 1998

The Advanced Encryption Standard (AES)

- NIST/FIPS next encryption standard
- Encryption standard for the future
- Successor to DES
- Security for over 30 years
- May protect sensitive information for 100 years
- Competition
- International selection process
- Public confidence

Calendar

Announcement	January 1997
Requirements workshop	April 1997
Final requirements	September 1997
Pre-submission	April 15, 1998
Submission	June 15, 1998
AES conference 1 – presentation	August 20–22, 1998
Including publication of the descriptions + implementations	
AES conference 2 – analysis	March 22–23, 1999
Selection of 5 finalists	April 15, 1999
AES conference 3	Beginning of 2000?
Final AES selection	2000?
FIPS process	2000–2001?

AES Requirements

- Blockcipher
- 128-bit blocks
- 128/192/256-bit keys
- "with a strength equal to or better than that of Triple-DES and significantly improved efficiency"
- Provide description and analysis
- Provide three implementations in two languages (reference and optimized in C, optimized in Java)
- If selected, royalty free worldwide

The 15 Submissions

Cipher	Submitted by	Country
CAST-256	Entrust	Canada
Crypton	Future Systems	Korea‡
Deal	Outerbridge	Canada†
DFC	ENS–CNRS	France
E2	NTT	Japan
Frog*	TecApro	Costa Rica
HPC*	Schroeppel	USA
LOKI97*	Brown, Pieprzyk, Seberry	Australia
Magenta	Deutsche Telekom	Germany
Mars	IBM	USA†
RC6	RSA	USA†
Rijndael*	Daemen, Rijmen	Belgium‡
Safer+*	Cylink	USA†
Serpent*	Anderson, Biham, Knudsen	UK, Israel, Norway
Twofish*	Counterpane	USA†

* Placed in the public domain; † and foreign designers; ‡ foreign influence

General Structure

Cipher	Type	Rounds	Using
CAST-256	Ext. Feistel	48	
Crypton	Square	12	
Deal	Feistel	6, 8, 8	DES
DFC	Feistel	8	Decorrelation modules, mult.
E2	Feistel	12	
Frog	Special	8	BombPermu
HPC	Omni	8	Hasty Pudding
LOKI97	Feistel	16	
Magenta	Feistel	6, 6, 8	
Mars	Ext. Feistel	32	Var. rot., mult., non-crypt. rounds
RC6	Feistel	20	Var. rot., mult.
Rijndael	Square	10, 12, 14	
Safer+	SP network	8, 12, 16	PHT
Serpent	SP network	32	Bitslice
Twofish	Feistel	16	

Var. rot.=Variable rotation. Mult.=Multiplication.

Selection Criteria

- Security
- Cost/Efficiency:
  - Software, Hardware, Smartcard
  - Intellectual property
- Algorithm characteristics
  - Flexibility
  - Simplicity and elegance
- Not by: Exportability, Nationality

Design

- Based on existing design?
- Totally new cipher?
- Feistel / SP network?
- Design the rounds
- How many rounds?

Used Instructions

- XOR
- S Boxes: 4x4, 8x8, 8x32, 11x8, 13x8, 8x32
- Addition, Subtraction
- Rotate, Shift
- Multiplication:
  - modulo 2^{32}: Efficient on Pentium II, but very slow on older processors
  - modulo 2^{64}
  - modulo $2^{64} + 13$
- Variable Rotations

Techniques

- Bitslicing
- PHT
- Decorrelation
- Non-cryptographic rounds
- Using other ciphers (DES)

Optimization Target

The designers had to decide which target platform to optimize for

- Pentium? MMX? Pro?
- Pentium II?
- 64-bit processors?
- 16-bit processors?
- 8-bit processors?
- Smartcards?
- Hardware?

The decision may crucially affect the design.

Optimization Target (cont.)

Even the compiler to be used for comparisons has a large effect:

NIST decided to use Borland C compiler on a 200MHz Pentium for comparisons.

However, unlike gcc, this compiler does not translate

$$(x \ll n)||(x \gg (32 - n))$$

to a rotation instruction.

As many ciphers used rotations (Mars, RC6, Serpent, etc.), the final AES decision might be affected by the choice of the compiler.

Security/Speed Tradeoffs

Small margins: adding a few rounds:

- RC6: Rivest assumes that there is an attack on 16-round RC6 with complexity 2^{119}. Proposes 20 rounds.
- DFC: An attack on 5 rounds. 8 rounds are proposed.
- Deal: An attack on 5 rounds. 6 are proposed.

Large margins: Doubling the number of rounds in the expense of speed:

- Serpent: 16 rounds are secure. 32 are proposed.
- Twofish: The best known attack is on 5 rounds. 10 rounds using related or weak keys. 16 are proposed.

Speed Comparisons

The figures in the papers of the submitted AES candidates claim speeds based on various measurement assumptions.

Some measure the speed of the cipher with NIST API.

Some measure the speed of the native procedures. This is usually 10–20% faster than using the NIST API.

Some measure the speed using various optimizations, which are incompatible with the NIST API (such as setting the subkeys in a static array, or even statically planting the subkeys into the encryption code in assembler). This might give additional 20% in speed for almost every cipher.

Fair comparison: comparing the mathematically optimized C implementations of the designers using a common test program.

Speed Comparisons (cont.)

Although the sources CD contains all the code, there are many problems to solve:

Deal assumed that the caller makes memory allocation for it.

DFC receives the input length in bytes, while all others receive in bits. Thus, it seemed eight times slower.

HPC comes without include files, which should be created manually (typed from a printed paper which comes with the CD). It also malloc's memory each makeKey, but fails to free it. So measuring the speed of makeKey is problematic due to memory constraints.

Magenta's implementation fails when the plaintext and ciphertext blocks reside in same memory.

Mars returns wrong return values (0 rather than TRUE).

Rijndael added a non-standard parameter to the API: variable block size. So it cannot be used with the standard calling form.

Speed Comparisons (cont.)

Some submissions verify that in ECB mode the IV is set to NULL. Some other initialize the IV even in ECB mode. Thus, a single main program cannot work for all submission supplied on the CD.

Many have special optimization macros and qualifiers to set.

Speed Comparisons (cont.)

The following table show the speed of the optimized implementations on Linux/GCC-2.7.2.2/Pentium MMX.

Only 128-bit keys are considered.

Speed Comparisons (cont.)

Cipher	Encrypt (cycles)	Decrypt (cycles)	Key Setup* encrypt	Key Setup* decrypt	Init	
Twofish	1254	1162	18846	18634	20	95.4% 92pf+0w
Rijndael	1276	1276	17742	18886	28	99.5% 98pf+0w
Crypton	1282	1286	758*	824*	24	99.7% 66pf+0w
RC6	1436	1406	5186	5148	30	94.0% 92pf+0w
Mars	1600	1580	4708	5548	18	96.7% 92pf+0w
Serpent	1800	2102	13154	12648	14	94.7% 98pf+0w
E2	1808	1854	7980	7780	24	96.0% 76pf+0w
— DES with NIST API —						
CAST-256	2088	2080	11412	11478	34	99.9% 67pf+0w
Frog	2182	2668	3857000	3817100	22	95.6% 64pf+0w
HPC	2602	2962	234346	248444	20	64.1% 142pf+5557w
Safer+	4424	4620	4708	4668	38	95.7% 88pf+0w
DFC	5874	5586	23914	25616	534	98.6% 65pf+0w
LOKI97	6376	6118	22756	22490	148	96.7% 108pf+0w
Deal	8950	8910	108396	107996	36	97.3% 68pf+0w
Magenta	23186	23230	1490	1622	24	99.2% 89pf+0w

Fair Speed/Security Comparisons

Consider the speed of the variants with the same security level.

Minimal secure variants have margins of two extra passes.

The minimal number of rounds is either that described by the designers or other cryptographers, or my best guess.

Fair Speed/Security Comparisons (cont.)

Cipher	Original Rounds (cycles)		Minimal Rounds	Time (cycles)
Twofish	1254	16	$\frac{6+10}{2}+4=12$	940
Serpent	1800	32	$15+2=17$	956
Mars	1600	32	$12+8=20$	1000
Rijndael	1276	10	$6+2=8$	1021
Crypton	1282	12	$9+2=11$	1175
RC6	1436	20	$16+4=20$	1436
E2	1808	12	$8+2+\mathrm{IT}+\mathrm{FT}=10$	1507
CAST-256	2088	48	$32+8=40$	1740
— DES with NIST API —				
Safer+	4424	8	$5+2=7$	3871
DFC	5874	8	$5+4=9$	6608
Deal	8950	6	$5+4=9$	13425
LOKI97	6376	16	$>32+4=36$	14346
Magenta	23186	6	$>6+4=10$	38643
Frog	2182	8	?	
HPC	2602	8	?	

Key Setup Time

Cipher	Claimed (encryption) (some missing)	Measured (encryption) (inaccurate)	
CAST-256	5	5	
Crypton	?	0.6	(no conversion from ASCII)
Deal	8	12	
DFC	?	4.3	
E2	4.4	3	
Frog	?	1700	
HPC	40	90	
LOKI97	1	4	
Magenta	?	0.05	
Mars	10	3	
RC6	8	4	
Rijndael	?	14	
Safer+	2	1	
Serpent	1	6.5	(< 2 without conversion from ASCII)
Twofish	13	15	

Smartcard Implementations

RAM and code size are the main criteria. RAM:

- RC6: 176 bytes are required to keep the subkeys. About 200 in total. Cannot use less.
- Mars: Similar. Can reduce the RAM requirement but at a considerable cost in speed (7 times slower?)
- DFC: 160 bytes (268 data bytes).
- Serpent: less than 64 bytes in total.

Gemplus predicts that 70% of the smartcards sold in 2003 will have 128 bytes or less of RAM.

EEPROM: several ciphers require 4K for the tables.

Hardware Implementations

Cipher		gates/cycles	gates/cycles
CAST-256	not given		
Crypton		19000 / > 6	50000 / ?
Deal	not given		
DFC	not given		
E2		127000 nand / 16	
Frog	not given		
HPC	not given		
LOKI97	not given		
Magenta	not given		
Mars		70000 cells / 50	
RC6	100 nano-sec		
Rijndael	not given		
Safer+		62000 cells / 134	
Serpent		4500 / 32	70000 / 1 (fully pipelined)
Twofish		14000 / 64	23000 / 16

Hardware Implementations (cont.)

Effect on Speed:

All ciphers have tradeoffs between hardware size and speed:

- Duplicate bottlenecks
- Increase table sizes
- Unroll
- Interleave blocks

Should take gates×cycles as the comparison parameter.

Hardware Implementations (cont.)

Cipher	gates	cycles/block	gates×cycles	Minimal variant	
Serpent	70000	1	70000	17/32	37187
Crypton	50000	?	100000?	11/12	91667
Twofish	23000	16	368000	12/16	276000
DES	28000?	16?	448000?		
E2	127000 nand	16	2032000	10/12	1693333
Mars	70000 cells	50	3500000	20/32	2187500
Safer+	62000 cells	134	8308000	7/8	7269500

The table should be recompiled based on comparable figures for the number of gates of each candidate

Novel vs. Conservative Design

Advantages of novel design:

- Might be faster
- No known cryptanalysis methods

Advantages of conservative design:

- We know how to analyze
- Can show that the cipher is immune to all known attacks (DC, LC, etc.)

Current Knowledge on the Submitted Algorithms

Cryptanalyzed:

- Frog – also slow key setup
- LOKI97 – also slow
- Magenta – also very slow

Slow:

- Deal
- DFC
- HPC – slow key setup
- Safer+

Minor weaknesses

- Mars – A few equivalent keys, a few weak keys
- Deal – An attack slightly faster than exhaustive search (considerably faster with 192-bit keys)
- DFC – A few weak keys

Remaining Ciphers

- Twofish
- Serpent
- Mars
- Rijndael
- Crypton
- RC6
- E2
- CAST-256
- Safer+
- DFC

Selection Process

- Open process
- Rely on public participation
- International

- NIST receives comments
- All comments become public
- NSA's comments might be disclosed (hopefully no such comments)
- NIST decides
- The decision is based on the comments

Request for Comments

- Cryptanalysis (security)
- Efficiency testing (speed)
- Intellectual property issues
- Fitness for applications
- Flexibility, Simplicity
- Confidence
- Comparisons of candidates
- Your choice of: required security margins, novel vs conservative design, importance of smartcards and other 8-bit applications, importance of hardware applications, etc
- Other relevant information

NSA's Role

- The NSA will help
- Hardware design and hardware comparison in the second AES round
- Analysis (might remain secret)

Your Part

- Most submissions will not be broken
- The final decision will be highly influenced by the comments and preferences of the cryptographic and user communities
- $\Rightarrow$ **Analyze. Test. Decide. And tell NIST.**

More Information

- AES home page http://www.nist.gov/aes/
- Including: AES process information, descriptions of the submitted algorithms, errata, AES conference information, electronic discussion groups, instructions for ordering code and sending comments, etc.
- The Block Cipher Lounge - AES:
  http://www.ii.uib.no/~larsr/aes.html
- Candidate AES for Analysis and Reviews (CAESAR):
  http://www.dice.ucl.ac.be/crypto/CAESAR/caesar.html