The AES Winner

Dhiren R. Patel
S V Regional Engineering College, Surat
Gujarat, INDIA - 395 007
dhiren@svrec.ernet.in

Abstract :

Picking up a winner from the AES finalist symmetric-key block ciphers is really an awesome task. All the algorithm viz. MARS, RC6™, Rijndael, Serpent, and Twofish are well balanced looking to the various evaluation criteria of expected and proposed security, implementation and performance characteristics. Selecting all 5 as AES-FIP is inappropriate with initial goal of NIST. Since the AES winner is expected to continue in use for another 15 to 20 years after adoption, it is appropriate to look for adaptability on future infrastructure and applications. This paper discusses the AES finalists and helps in selecting the winner.

Keywords :
Advanced Encryption Standard(AES), Cryptanalysis, Symmetric-key Block Cipher

1. Introduction

Its already been three years since NIST started the efforts for the Advanced Encryption Standard(AES). In August 1999 NIST has declared algorithms MARS, RC6™, Rijndael, Serpent, and Twofish as finalists after lots of reviews and research results. This was based on the evaluation criteria of Security, Performance/Implementation, and adaptability issues - as well as algorithms' strength against various cryptanalysis and its trusted core mathematics. We all know that the AES will be used to protect enormous volumes of critical information, financial transactions, health records, and government information. So now - this is the last time a question is asked : "Which algorithm(s) should NIST include in the AES Federal Information Processing Standard(FIPS), and why?" This paper discusses various comparison issues and finally answers the appropriate AES winner.

2. Design Principles and Security

It is important to note that confidence in the security of a cipher grows if it has been subjected to and withstood expert cryptanalysis over a substantial time period, e.g., several years or more; such ciphers are certainly considered more secure than those who have not. Conservative designs with use of known structures can be easily analyzed by well known cryptanalytic techniques. Their security is thus easier to study, and they can be shown immune against the standard cryptanalytic tools. On the other hand, design with novel ideas might increase speed or improve other properties of the cipher. In such designs, however, it is difficult to bound the strength, as the used techniques are new, and no proven cryptanalytic techniques are known that can compare the strength to other known ciphers. It may happen

that somebody find a new attack against such ciphers. Such a new attack is less expected against conservative designs.

Three of the 5 finalist algorithms use conventional Feistel core with little or no modifications while remaining two use Substitution Permutation Network - one with Square as a core and other with introduction of bit-slice operations. Algorithm's complexity affects the implementation costs both in terms of development and fixed resources(hardware gate count or software code/data size), as well as real time performance for fixed resources(throughput).

General comments on algorithms' core structure and development design base are provided in table 1.

Algorithm	**Core structure / Design Principles used**
MARS	Modified Feistel Network - Mixed structure DES
RC6™	Feistel Network - Modified RC5
Rijndael	Modified Substitution Permutation Network - Square
Serpent	Substitution Permutation Network - Bitslice
Twofish	Feistel Network - Modified Blowfish

Table 1. Core structure / Design Principles used

Looking to the crypto core and historical design principles Rijndael looked to be weaker(because of Square's vulnerability) on security front than other finalist candidates, while RC6™ with proven RC5 modifications has an edge over remaining algorithms. All algorithms offer more or less similar level of security and confidence with different margin of safety. It should be worth presenting Biham's remarks on minimal no. of rounds for reasonable security given in table 2. It is also to note that round in different cipher are not straight metric - i.e. one more round of Twofish could in practice offers more security than the same of say RC6™. However this can be easily equalised by adjusting number of rounds, for desired level of security at given cost.

Algorithm/Cipher	**Original**	**Minimal Rounds**	**Relative speed with**

	Rounds/Stages		**minimal round**
MARS	32	20	1.00
RC6™	20	21	0.663
Rijndael	10	8	0.98
Serpent	32	17	1.04
Twofish	16	14	0.911

Table 2. Security with minimal no. of rounds

3. Some General Remarks

The fundamental problems in computer security are no longer about technology; they're about applying the technology. The real world offers the adversary a richer menu of options than mere cryptanalysis. Often more worrisome are protocol attacks, Trojan horses, viruses, electromagnetic monitoring, physical compromise, operating system bugs, memory environment bugs, application program bugs, hardware bugs, user errors, physical eavesdropping etc. Good quality cipher should have the nice property of making life much harder for the attacker than for the legitimate user. To be meaningful, attack comparisons based on different models(e.g. exhaustive precomputation, exhaustive search, linear cryptanalysis, differential cryptanalysis) must appropriately weigh the feasibility of extracting enormous amounts of chosen plaintexts, which is considerably more difficult to arrange than a comparable number of computing cycles on an adversary's own machine. For symmetric-key block ciphers, data complexity is beyond the control of the adversary and is passive complexity. When parallelization is possible, processing complexity may be divided across many processors, reducing attack time.

In the world of abundant computing one should not worry much about performance of these candidate algorithms on low end platforms or thinking on scarcity of resources. It is well known fact that logical operations, table lookups and fixed shifts are relatively easy to secure against Timing and power attacks, while multiplication or variable rotations are very difficult to secure. Author feels that these attacks are not posing any big threats in practical environments as countermeasures are readily available in terms of software balancing, desynchronization, and device/manufacturing technology. Once adopted as an AES FIP – chip manufacturers will produce hardware platforms with proper circuit balancing -- which will eliminate Power discrepancies. In general, one should look for algorithm's future-proof foundations.

4. Selection :

To simplify difficult task of picking up winner amongst the finalist candidates, certain parameters/metrics are used on 100 point scale. There are eight parameters with appropriate distribution of points based on NIST's goals and author's assessment thoughts. There may be little duplication/overlapping found in certain parameters which was deliberately kept on the basis of importance. Since it is quite comprehensive to discuss all those metrics for each

algorithm, after careful examination and review of research results(based on References and Author's assessment), a summary (table 3) is presented to pick up appropriate winner at the end of this section.

Algorithm Design & Presentation(10 points)
It is about simplicity, clearly understandable cryptographic core, round functions, and related building blocks giving idea about underlying design principles that present the Encryption/Decryption process and allow easy cryptanalysis. Here RC6™ is cleary ahead of others.[1, 2]

Security(30 points)
It is about inherent security of the algorithm when used as a block cipher. This includes usage of strong, tested and trusted cryptography core, well integrated design blocks giving proper Encryption/Decryption functions, proper key scheduling, the work factor, non use of weak operations, appropriate diffusion, whitening, no trapdoor, no weak keys(includes palindromic) or semi-weak keys, expected strength level, conservative safety margins. It also tries to look at assurance about one-to-one mapping of plaintext to ciphertext under the same key and different one-to-one mapping under different keys. Here RC6™ , MARS and Serpent are better. [1, 8,9,14,17]

Ease of Implementation(10 points)
It is largely influenced by simplicity and algorithm's coding capability to run elegantly on different hardware(standard as well as large microprocessors), software and operating environments. It also includes algorithm's implementation on dedicated VLSI hardware. Looking to the complexity of cryptographic mapping RC6™ has an edge over others. [2,6,7,9,17]

Usage Flexibility(10 points)
It is about algorithm's usage as building block for Hash function, Pseudo Random Number Generator, Message Authentication Codes, and Stream Cipher. Almost all finalists fulfil well here.[14]

Performance/Computational Efficiency(10 points)
It is about algorithm's performance characteristics on various platforms(standard as well as new) under different configuration(key size, block size, no of rounds). It does not include smart card performance as it is treated separately. Twofish and RC6™ performing better on general platforms. [2, 4,12,17]

Performance/Adaptability on Smart cards(10 points)
It is about algorithm's performance and adaptability on low end as well as advanced smart cards. Rijndael and Serpent are better here. [4,10,11,15]

Demonstrated/Expected strength against Cryptanalysis(10 points)
It is about algorithm's strength against linear, non-linear, statistical and differential cryptanalysis including power and timing attacks. It also covers interpolation attack, partial

key guessing, related key cryptanalysis, distributed key search, short-cut attacks as well as visual cryptanalysis. It is additional to the 30 points of Security. Twofish and RC6™ with good background win here.[9, 14, 15]

Future Resilience (10 points)
It includes algorithm's flexibility to be used with adjustable block size/key size, exploitable inherent parallelism, and suitability for future architectures(eg. Dataflow), operating environments as well as resilience against future/unknown attacks. Serpent looked to be the best suited here. [6,7,15]

Metric / Algorithm	**MARS**	**RC6™**	**Rijndael**	**Serpent**	**Twofish**
Algorithm Design & Presentation (10 points)	8	10	8	8	8
Security (30 points)	28	29	25	28	27
Ease of Implementation (10 points)	8	9	7	8	7
Usage Flexibility (10 points)	8	8	7	8	8
Performance/Efficiency (10 points)	8	9	8	7	9
Performance on Smart cards (10 points)	8	7	9	9	8
Strength against Cryptanalysis (10 points)	8	9	7	8	9
Future Resilience (10 points)	8	8	7	9	8
Total (max. 100)	**84**	**89**	**78**	**85**	**84**

Table 3. Final summary

5. Conclusion

As derived through summary table 3, RC6™ appears to be the best choice for the AES-FIPS. It scores maximum points because of its proven security, simplicity - implementation ease, speed/cost on present and future general platforms. Its scaleable performance characteristics make it more flexible for the different demanding environments. On security front - RC6™, though simple and so easy to launch attacks and cryptanalysis - no severe flaw has been reported yet shows its inherent strength. It is bit slower on low end smart cards, but advanced smart cards with more RAM/ROM and strong CPU -- compensates the implementation issues. Because of clear algorithm presentation and trust in conventional Feistel core makes RC6™ faithful(secure and without any possible trapdoor) and more suitable to be tuned for future platforms (64-bit or hybrid -RISC/DataFlow), smart cards, crypto hardware, as well as powerful parallel machines and author strongly feels that RC6™ should be used as an AES without major security and implementation issues.

References :

1. Alfred Menezes, Paul van Oorsohot, Scott Vanston
   Handbook of Applied Cryptograhy
   CRC Press

2. Brian Gladman
   Implementation Experience with AES Candidate Algorithms

3. Bruce Schneier
   Applied Cryptography
   John Wiley & Sons

4. Bruce Schneir et al.
   Twofish: A 128-bit Block Cipher

5. Bruce Schneir et al.
   Performance Comparison of the AES Submissions

6. Craig S. K. Clapp
   Instruction-level Parellelism in AES Candidates

7. Don B. Johnson
   Future Resiliency: A Possible New AES Evaluation Criterion

8. Eli Biham, Adi Shamir
   Power Analysis of the Key Scheduling of the AES Candidates

9. Eli Biham
   A Note on Comparing the AES Candidates

10. Gael Hachez et al.
cAESar results : Implementation of Four AES Candidates on Two Smart Cards

11. Geoffery Keating
Performance Analysis of AES candidates on the 6805 CPU core

12. Helger Lipmaa
AES Candidates: A Survey of Implementations

13. IBM Corporation
MARS - a candidate cipher for AES

14. James Nechvatal et al. NIST
Status Report on the First Round of the Development of the AES

15. Joan Daemen, Vincent Rijmen
AES Proposal : Rijndael

16. John Daemen, Vincent Rijmen
Resistance against Implementation Attacks : A Comparative Study of AES Proposals

17. Oliver Baudron et al.
Report on the AES Candidates

18. R. L. Rivest, M.J.B. Robshaw, R. Sidney, Y.L. Yin
The RC6^{TM} Block Cipher

19. Ross Anderson, Eli Biham, Lars Knudsen
Serpent: A Proposal for the Advanced Encryption Standard