Global Information Assurance Certification Paper

AES: The New Key on the Block

Christopher Silveira
May 1, 2003

Cryptosystems, highly technical systems that provide privacy through secret encoding, have been an important part of the electronic information world for many years. These systems are the foundation for all electronic information exchange. For example, financial institutions and banks rely upon cryptography to securely transmit critical and private information over the Internet. Especially now, during the e-commerce explosion, secure and reliable exchange systems are vital for the world's economy.

These cryptosystems protect data by using hardware and software in a process that protects data by performing mathematical operations/algorithms on it. The result is data rendered unintelligible, which is called ciphertext. This ciphertext is then transmitted over insecure phone lines or networks such as the Internet. If someone intercepts this ciphertext, it is indecipherable and meaningless to him or her. When the ciphertext reaches its final destination, it can be decrypted into the original state of the data.

The most widely used encryption algorithm is the Data Encryption Standard (DES). Proposed in 1975, DES was adopted by the US government as the standard for all "unclassified computer data". (1) DES is a symmetric key block cipher. This means that data to be secured is encrypted with a "private key" in sections, or blocks, of 64-bits. Anyone who needs the data must then use this key to decrypt it. A DES key has a length of 56-bits, which results in a maximum combination of 2^{56} possible keys. With the tremendous growth of technology this number of keys did not seem like enough to keep DES safe from attempts to crack it. In 1997 a project was launched to see how much effort it would take to crack a DES key. From this project a DES key was cracked in less than 3 days and for less than $250,000. (1) More recently, in 1999 a network of 100,000 computers was able to decrypt a DES encrypted message in less than 24 hours. (3) This made it clear that a new system was needed.

In an effort to replace DES, the National Institute of Standards and Technology (NIST) started the task to implement a new standard. The Advanced Encryption Standard (AES) would be "an unclassified, publicly disclosed encryption algorithm(s), available royalty-free, worldwide." (2) There were several factors that would need to be considered in the algorithm's design. Security would be the most important factor, as the algorithm would need to be able to withstand attacks into the future. It would need to be simple and publicly available as well. This way the cryptography community could easily examine it for security and efficiency. Performance would also be a consideration. The algorithm would need to operate fast and effectively on several different platforms ranging from personal computers to smart cards.

The development effort for AES was made public in a call for algorithms in 1997. The requirements for submissions were that the algorithm must be a symmetric key

cryptosystem implemented as a block cipher, and must support a block size of 128-bits and key lengths of 128, 192, and 256-bits. Compared to a DES key of 56-bits that results in only 2^{56} possible keys, a 128-bit key would produce approximately 340,000,000,000,000,000,000,000,000,000,000,000,000 ($3.4 * 10^{38}$) possible keys. The algorithms must also be freely available, and without any royalties. Fifteen algorithms from candidates around the world were submitted to NIST. The selection process consisted of two rounds in which the candidates were reviewed by cryptographic researchers.

In the first round, the candidate algorithms were presented to the public for examination. According to NIST the candidates were evaluated according to criteria in three major categories, security, cost, and algorithm and implementation characteristics:

> ***Security*** is the most important factor in the evaluation. Security encompasses features such as resistance of the algorithm to cryptanalysis, soundness of its mathematical basis, randomness of the algorithm output, and relative security as compared to other candidates.
>
> ***Cost*** is a second important area of evaluation that encompasses licensing requirements, computational efficiency (speed) on various platforms, and memory requirements. Since one of NIST's goals is that the final AES algorithm(s) be available worldwide on a royalty-free basis, intellectual property claims and potential conflicts must be considered in the selection process. The speed of the algorithms on a variety of platforms must also be considered. During Round 1, the focus was primarily on the speed associated with 128-bit keys. Additionally, memory requirements and constraints for software implementations of the candidates are important considerations.
>
> The third area of evaluation is ***algorithm and implementation characteristics*** such as flexibility, hardware and software suitability, and algorithm simplicity. Flexibility includes the ability of an algorithm:
>
> - to handle key and block sizes beyond the minimum that must be supported,
> - to be implemented securely and efficiently in many different types of environments, and
> - to be implemented as a stream cipher, hashing algorithm, and to provide additional cryptographic services.
>
> It must be feasible to implement an algorithm in both hardware and software, and efficient firmware implementations are advantageous. The relative simplicity of an algorithm's design is also an evaluation factor. (5)

Round 1 was open to public review from which members of the cryptographic community were invited to analyze and test the fifteen candidates. NIST hosted an electronic forum during this time for discussions about the candidate algorithms as well as new analytical results. Twenty-eight papers evaluating the candidates were submitted to NIST and posted on the NIST web site for the public to review. NIST also hosted a conference to encourage discussions about the analysis that was presented to the global community.

NIST analyzed both ANSI C and Java implementations of the candidates. The ANSI C testing primarily focused on speed in different desktop systems using assorted processors, operating systems, and compilers. Testing on Java implementations focused on speed and memory usage along with other coding features. In addition to the main analysis, NIST also performed extensive statistical testing on the algorithms. This testing was used to determine if the algorithms generated output that is "statistically indistinguishable" from random data. (5)

NIST put together a team at the end of the first round to review the candidates and make a selection for the finalist algorithms. The team was a group of NIST employees who had been involved in the review of the algorithms throughout the span of the first round. Over a two month span, the team met to make their recommendations.

The NIST team took into consideration all information provided by the public. This included written comments and those papers reviewing the algorithms. The team also reviewed the NIST studies and proposed modifications. Each candidate was assessed according to the announced evaluation criteria and other criteria brought up by the public. According to NIST, each assessment followed a meticulous evaluation of the following factors:

- security (including any known attacks or weaknesses),
- efficiency (both speed and memory usage),
- flexibility (implementation on low- and high-end smart cards; support of additional key and block sizes, including whether the reference code actually supported the additional key sizes; suitability for use as a pseudo-random number generator, hashing algorithm, etc.; and whether or not encryption and decryption were the same procedure),
- algorithm simplicity, and
- other issues that were discussed in the received public comments. (5)

Security was considered to be the most important of the factors, so the NIST team made an initial selection based on that. After this selection, the candidates that remained were evaluated according to the other criteria. When this process was finished five of the original fifteen candidates were selected as the finalists for Round 2.

The five algorithms selected as finalists were MARS, RC6™, Rijndael, Serpent, and Twofish. According to NIST, there were no significant security vulnerabilities found in these algorithms and represented a potentially superior technology. Listed below are the summaries of each finalist provided by NIST:

> **MARS** incorporates its "cryptographic core" into an innovative, heterogeneous overall structure. It also features a variety of operations, including the technique of rotating digits by a varying number of places that is determined by both the data and the secret key. Consequently, while MARS performs well in general, it performs particularly well on computer platforms that support its rotation and multiplication operations efficiently. NIST accepted a modification to MARS for Round 2 (proposed by the submitter) that should improve its ability and flexibility to function in some memory-constrained environments, such as low-end smart cards. MARS was submitted to the AES development effort by the International Business Machines Corporation.

RC6 is an algorithm that is simple enough to memorize and should be easy to implement compactly in both software and hardware. Its simplicity also should facilitate its further security analysis in Round 2, which is assisted by the analysis of its predecessor, RC5. RC6 does not use substitution tables; instead, the principal engine for its security is the technique of rotating digits by a varying number of places that is determined by the data. In general, RC6 is fast and it is particularly fast on platforms that support its rotation and multiplication operations efficiently; its key setup is also fast. RC6 was submitted to the AES development effort by RSA Laboratories.

Rijndael performs excellently across all considered platforms. Its key setup is fast and its memory requirements are low, so it also should perform well in hardware and in memory-constrained environments. The straightforward design and the conservative choice of operations should facilitate its further analysis, and the operations should be relatively easy to defend against certain attacks on physical implementations. Even though parallel processing was not considered during the Round 1 selection process by the AES review team, Rijndael has the potential of benefiting from advances in computer processors that allow many instructions to be executed in parallel. Rijndael was submitted to the AES development effort by Joan Daemen and Vincent Rijmen.

Serpent is ultra-conservative in its security margin; the designers chose to use twice as many iterations as they believed secure against currently known attacks. Consequently, Serpent's performance is relatively slow compared to the other four finalists. In some settings, however, this should be mitigated by the efficiency of optimized implementations using what the submitters call the "bitslice" mode, for which the algorithm was specially designed. Serpent should fit well in hardware (with potential tradeoffs of speed versus space) and in memory-constrained environments. The straightforward design and the conservative choice of operations should facilitate further analysis of this candidate, and the operations should be easy to defend against certain attacks on physical implementations. Serpent was submitted to the AES development effort by Ross Anderson, Eli Biham, and Lars Knudsen.

Twofish exhibits fast and versatile performance across most platforms; it also should perform well both in hardware and in memory-constrained environments. It features variable substitution "tables" that depend on the secret key. The submitters believe that such tables generally offer greater security than tables with fixed values. The possibility of pre-computing these tables to varying degrees helps Twofish offer a wide variety of performance tradeoffs. Depending on the setting, Twofish can be optimized for speed, key setup, memory, code size in software, or space in hardware. Twofish was submitted to the AES development effort by Bruce Schneier, John Kelsey, Doug Whiting, David Wagner, Chris Hall, and Niels Ferguson. (5)

In the second round once again NIST hosted discussion forums and accepted comments and formal papers from the public. NIST also held another public conference near the end of the second round in order to distribute the information it had collected. Some of the finalist algorithms were updated in between the first and second rounds with suggestions made from the community and were analyzed again.

At the end of the second round NIST declared Rijndael the winner and would become the algorithm selected for AES. There were several reasons why Rijndael was chosen over the other four finalists. Rijndael performed well in both hardware and software implementations over a large range of environments. It is a fast algorithm in both key setup and in encryption/decryption operations. Rijndael's low-memory

requirements also helped it achieve excellent performance in restricted-space environments. Of all the finalists Rijndael's operations are among the easiest to defend against timing and power attacks. Plus defending against these types of attacks does not appear to have much of a significant impact on its performance. Rijndael also has a good deal of flexibility, allowing changes to be made to the number of rounds it uses for data encryption as well as in block and key sizes. In conclusion NIST stated that "…Rijndael's combination of security, performance, efficiency, implementability, and flexibility make it an appropriate selection for the AES for use in the technology of today and in the future." (6)

With an algorithm selected, AES now must be approved and published as a standard by the government. This is targeted for sometime in early 2001. Once it is identified as an approved algorithm, it can be used by US government organizations for unclassified information. Non-government and commercial organizations may also use the AES, but law will not require them to. There are already products available incorporating the AES algorithm at this time. The most notable of these products is Pretty Good Privacy (PGP), a popular email and file encryption program available on many different platforms. More information can be found at their home page http://www.pgp.com.

Electronic information has become a vital part of the world economy and will continue to grow in its importance. With our reliance on electronic information, we need ways to protect it from unauthorized sources. Up to now we have relied on DES encryption to protect our financial data and other important information. DES has outlasted its usability and is not considered to be secure enough for the future. The Rijndael algorithm was chosen to be the new AES encryption standard. Its speed and versatility, in addition to its resilience to security attacks promise to protect electronic information exchange in the years to come. As long as there is information to protect, there will always be individuals who work to break that protection. Time will tell how long the new standard can hold up.

References

(1) Electronic Frontier Foundation. "EFF DES Cracker Project." URL: http://www.eff.org/DESCracker/, (October 16, 2000).

(2) National Institute of Standards and Technology. "Advanced Encryption Standard (AES) Fact Sheet." URL: http://csrc.nist.gov/encryption/aes/round2/aesfact.html, (October 19, 2000).

(3) Srinivas, Raghavan. "AES: Cryptography advances into the future." URL: http://www.javaworld.com/javaworld/jw-04-2000/jw-0428-aes.html, (October 19, 2000).

(4) Schneier, Bruce. "Crypto-Gram – October 15, 2000." URL: http://www.counterpane.com/crypto-gram-0010.html, (October 31, 2000).

(5) National Institute of Standards and Technology. “ITL Bulletin - The Advanced Encryption Standard: A Status Report.” URL: http://www.nist.gov/itl/lab/bulletns/aug99.htm, (November 2, 2000).

(6) National Institute of Standards and Technology. “Report on the Development of the Advanced Encryption Standard (AES).” URL: http://csrc.nist.gov/encryption/aes/round2/r2report.pdf, (November 2, 2000)

Last Updated: June 26th, 2017

Upcoming Training

SANS London July 2017	London, United Kingdom	Jul 03, 2017 - Jul 08, 2017	Live Event
Cyber Defence Japan 2017	Tokyo, Japan	Jul 05, 2017 - Jul 15, 2017	Live Event
SANS Cyber Defence Singapore 2017	Singapore, Singapore	Jul 10, 2017 - Jul 15, 2017	Live Event
SANS Los Angeles - Long Beach 2017	Long Beach, CA	Jul 10, 2017 - Jul 15, 2017	Live Event
Community SANS Phoenix SEC401	Phoenix, AZ	Jul 10, 2017 - Jul 15, 2017	Community SANS
SANS Munich Summer 2017	Munich, Germany	Jul 10, 2017 - Jul 15, 2017	Live Event
Mentor Session - SEC401	Ventura, CA	Jul 12, 2017 - Sep 13, 2017	Mentor
Mentor Session - SEC401	Macon, GA	Jul 12, 2017 - Aug 23, 2017	Mentor
Community SANS Atlanta SEC401	Atlanta, GA	Jul 17, 2017 - Jul 22, 2017	Community SANS
SANSFIRE 2017	Washington, DC	Jul 22, 2017 - Jul 29, 2017	Live Event
SANSFIRE 2017 - SEC401: Security Essentials Bootcamp Style	Washington, DC	Jul 24, 2017 - Jul 29, 2017	vLive
Community SANS Fort Lauderdale SEC401	Fort Lauderdale, FL	Jul 31, 2017 - Aug 05, 2017	Community SANS
SANS San Antonio 2017	San Antonio, TX	Aug 06, 2017 - Aug 11, 2017	Live Event
SANS Boston 2017	Boston, MA	Aug 07, 2017 - Aug 12, 2017	Live Event
SANS Prague 2017	Prague, Czech Republic	Aug 07, 2017 - Aug 12, 2017	Live Event
SANS New York City 2017	New York City, NY	Aug 14, 2017 - Aug 19, 2017	Live Event
SANS Salt Lake City 2017	Salt Lake City, UT	Aug 14, 2017 - Aug 19, 2017	Live Event
Community SANS Omaha SEC401*	Omaha, NE	Aug 14, 2017 - Aug 19, 2017	Community SANS
Community SANS San Diego SEC401	San Diego, CA	Aug 21, 2017 - Aug 26, 2017	Community SANS
SANS Virginia Beach 2017	Virginia Beach, VA	Aug 21, 2017 - Sep 01, 2017	Live Event
Virginia Beach 2017 - SEC401: Security Essentials Bootcamp Style	Virginia Beach, VA	Aug 21, 2017 - Aug 26, 2017	vLive
SANS Adelaide 2017	Adelaide, Australia	Aug 21, 2017 - Aug 26, 2017	Live Event
SANS Chicago 2017	Chicago, IL	Aug 21, 2017 - Aug 26, 2017	Live Event
Community SANS Trenton SEC401	Trenton, NJ	Aug 21, 2017 - Aug 26, 2017	Community SANS
Mentor Session - SEC401	Minneapolis, MN	Aug 29, 2017 - Oct 10, 2017	Mentor
SANS San Francisco Fall 2017	San Francisco, CA	Sep 05, 2017 - Sep 10, 2017	Live Event
SANS Tampa - Clearwater 2017	Clearwater, FL	Sep 05, 2017 - Sep 10, 2017	Live Event
Mentor Session - SEC401	Edmonton, AB	Sep 06, 2017 - Oct 18, 2017	Mentor
SANS Network Security 2017	Las Vegas, NV	Sep 10, 2017 - Sep 17, 2017	Live Event
Community SANS Albany SEC401	Albany, NY	Sep 11, 2017 - Sep 16, 2017	Community SANS
Community SANS Dallas SEC401	Dallas, TX	Sep 18, 2017 - Sep 23, 2017	Community SANS