Network Working Group
INTERNET DRAFT
March 2000
Expiration Date: September 2000

IPsec Working Group
S. Frankel, NIST
R. Glenn, NIST
S. Kelly, RedCreek

The Candidate AES Cipher Algorithms and Their Use With IPsec
<draft-ietf-ipsec-ciph-aes-cbc-00.txt>

Status of this Memo

This document is a submission to the IETF Internet Protocol Security (IPSEC) Working Group. Comments are solicited and should be addressed to the working group mailing list (ipsec@tis.com) or to the editors.

Distribution of this memo is unlimited.

Abstract

This document describes the use of the AES Cipher Algorithms in Cipher Block Chaining Mode, with an explicit IV, as a confidentiality mechanism within the context of the IPsec Encapsulating Security Payload (ESP).

This Internet Draft specifies the use of each of the 5 AES finalist candidates in the ESP Header. Once the AES cipher is chosen, this document will be changed to reflect that choice.

Table of Contents

1. Introduction

Recognizing that the venerable DES cipher was reaching the end of its useful life, in January 1997 NIST (the National Institute of Standards and Technology) announced a plan to select its successor, the AES (Advanced Encryption Standard). The AES will be the government's designated encryption cipher, and will be definitively described in a FIPS (Federal Information Processing Standard). The expectation is that the AES will suffice to protect sensitive government information at least until the next century. It is also expected to be widely adopted by businesses and financial institutions.

The initial call for AES candidates specified the following requirements:

+ unclassified

+ publicly disclosed

+ available royalty-free worldwide

+ capable of handling a block size of at least 128 bits

+ at a minimum, capable of handling key sizes of 128, 192, and 256 bits

The distinguishing characteristics on which the final AES cipher will be selected are:

+ security

+ computational efficiency and memory requirements on a variety of software and hardware, including smart cards

+ flexibility and simplicity

Of the 15 ciphers that were submitted as AES candidates in August 1998, 5 were designated as finalists. Analysis and discussion of the candidates continues. Either 1 or 2 of the finalists will be selected as the AES cipher; the AES FIPS is expected to be completed by summer 2001.

It is the intention of the IETF IPsec Working Group that AES will eventually be adopted as the default IPsec ESP cipher and will obtain the status of MUST be included in compliant IPsec implementations. However, until 1 or 2 of the finalists are selected and until there is more experience with regard to the cryptographic strengths and weaknesses of the algorithms, this document should be used to experiment with the AES candidates and determine how they can best be used in IPsec implementations. This document should be considered experimental.

The remainder of this document specifies the use of the five finalist AES candidate ciphers within the context of IPsec ESP. For further information on how the various pieces of ESP fit together to provide security services, refer to [ARCH], [ESP], and [ROAD].

1.1 Specification of Requirements

The keywords "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" that appear in this document are to be interpreted as described in [RFC-2119].

2. The Candidate AES Cipher Algorithms

All symmetric block cipher algorithms share common characteristics and variables, including mode, key size, weak keys, block size, and rounds. The following sections contain descriptions of the relevant characteristics of the candidate AES ciphers.

Some of the candidate AES ciphers are covered by copyrights, patents or patent applications. Each submitter has sworn that, if selected as the AES cipher algorithm, the algorithm will be made available world-wide on a royalty-free basis.

The AES homepage, http://www.nist.gov/aes, contains a wealth of information about the 5 finalists, including definitive descriptions of each algorithm, comparative analyses, performance statistics, test vectors and intellectual property information. This site also contains information on how to obtain reference implementations from NIST for each of the candidate algorithms.

2.1 Mode

No operational modes are currently defined for the AES ciphers. However, the Cipher Block Chaining (CBC) mode is well-defined and well-understood for symmetric ciphers, and is currently required for all other ESP ciphers. This document specifies the use of the AES ciphers in CBC mode within ESP. This mode requires an Initialization Vector (IV) that is the same size as the block size. Use of a randomly generated IV prevents generation of identical ciphertext from packets which have identical data that spans the first block of the cipher algorithm's block size.

The IV is XOR'd with the first plaintext block before it is encrypted. Then for successive blocks, the previous ciphertext block is XOR'd with the current plaintext, before it is encrypted.

More information on CBC mode can be obtained in [CRYPTO-S]. For the use of CBC mode in ESP with 64-bit ciphers, see [CBC].

[AUTHORS' NOTE: Should we require CBC mode using the ciphertext from the previously generated block? On the AES discussion list, it has been suggested that a Counter Feedback Mode be defined, which allows parallel encryption of blocks. Should we stick with CBC, use some variant of a Counter Feedback Mode, or wait for the AES FIPS to decide?]

2.2 Key Size

Some cipher algorithms allow for variable sized keys, while others only allow specific, pre-defined key sizes. The length of the key typically correlates with the strength of the algorithm; thus larger keys are usually harder to break than shorter ones.

This document stipulates that all key sizes MUST be a multiple of 8 bits.

This document specifies the default (i.e. MUST be supported) key size for all of the AES cipher algorithms. All of the candidate ciphers were required to accept key sizes of 128, 192 and 256 bits. The default key size that implementations MUST support for IPsec is 128 bits.

Algorithm	Key Sizes (bits)	Default
MARS	128 - 448*	128
RC6	variable up to 2040	128
Rijndael	128, 192, 256	128
Serpent	variable up to 256**	128
Twofish	variable up to 256***	128

*NOTE1: MARS key lengths must be multiples of 32 bits.

**NOTE2: Serpent keys are always padded to 256 bits. The padding consists of a "1" bit followed by "0" bits.

***NOTE3: Twofish keys, other than the default sizes, are always padded with "0" bits up to the next default size.

2.3 Weak Keys

At the time of writing this document there are no known weak keys for any of the AES ciphers.

Some cipher algorithms have weak keys or keys that MUST not be used due to their interaction with some aspect of the cipher's definition. If weak keys are discovered for any of the AES ciphers, then weak keys SHOULD be checked for and discarded when using manual key management. When using dynamic key management, such as [IKE], weak key checks SHOULD NOT be performed as they are seen as an unnecessary added code complexity that could weaken the intended security [EVALUATION].

2.4 Block Size and Padding

All of the algorithms described in this document use a block size of sixteen octets (128 bits), as required in the AES specifications. Some of the algorithms can handle larger block sizes as well.

Padding is required by the candidate AES algorithms to maintain a 16-octet (128-bit) blocksize. Padding MUST be added, as specified in [ESP], such that the data to be encrypted (which includes the ESP Pad Length and Next Header fields) has a length that is a multiple of 16 octets.

Because of the algorithm specific padding requirement, no additional padding is required to ensure that the ciphertext terminates on a 4-octet boundary (i.e. maintaining a 16-octet blocksize guarantees that the ESP Pad Length and Next Header fields will be right aligned within a 4-octet word). Additional padding may be included, as specifed in [ESP], as long as the 16-octet blocksize is maintained.

2.5 Rounds

This variable determines how many times a block is encrypted. While this variable MAY be negotiated, a default value MUST always exist when it is not negotiated.

Algorithm	Negotiable?	Default # of Rounds
MARS	Yes	32
RC6	Yes	20
Rijndael	Yes	10, 12, 14*
Serpent	Yes	32
Twofish	Yes	16

*NOTE1: Rijndael's Default # of Rounds is dependent on key size. Default # of Rounds = keylen/32 + 6.

2.6 Cipher-specific Information

MARS:

MARS is IBM's submission to the AES competition. The inventors, who are from the US and Switzerland, are: Carolynn Burwick, Don Coppersmith, Edward D'Avignon, Rosario Gennaro, Shai Halevi, Charanjit Jutla, Sstephen Matyas Jr., Luke O'Connor, Mohammad Peyravian, David Safford, and Nevenko Zunic, A patent application, IBM application CR99802, is pending. However, the MARS homepage contains the following statement: "MARS is now available world-wide under a royalty-free license from Tivoli." MARS is defined in [MARS-1] and [MARS-2]. A change to the key generation technique is described in [MARS-3]. The MARS homepage is: http://www.research.ibm.com/security/mars.html.

RC6:

RC6 was invented by Ronald Rivest of MIT, and by Matthew Robshaw, Ray Sidney, and Yiqun Lisa Yin, all from RSA Laboratories. The name RC6 is protected by a copyright. The algorithm is covered by USA patent number 5,724,428 (granted March 3, 1998); two other US patents are pending: application serial numbers 08/854,210 (filed April 21, 1997) and 09/094,649 (filed June 15, 1998). The RC6 family of algorithms is defined in [RC6]. The RC6 homepage is: http://www.rsasecurity.com/rsalabs/aes/.

Rijndael:

Rijndael was invented by Joan Daemen from Banksys/PWI and Vincent Rijmen from ESAT-COSIC, both in Belgium. It is not covered by any patents, and the Rijndael homepage contains the following statement: "Rijndael is available for free. You can use it for whatever purposes you want, irrespective of whether it is accepted as AES or not." Rijndael's description can be found in [RIJNDAEL]. The Rijndael homepage is: http://www.esat.kuleuven.ac.be/~rijmen/rijndael/.

Serpent:

Serpent was invented by Ross Anderson of Cambridge University, Eli Biham of the Technion, Israel and Lars Knudsen of the University of Bergen, Norway. Two UK patent applications are pending: 9722789.7 (filed October 29, 1997) and 9722798.9 (filed October 30, 1997). However, the Serpent homepage contains the following statement: "Serpent is now completely in the public domain, and we impose no restrictions on its use." Serpent is defined in [SERPENT-1] and [SERPENT-2]. The Serpent homepage is: http://www.cl.cam.ac.uk/~rja14/serpent.html.

Twofish:

Twofish was invented by Bruce Schneier, John Kelsey, Chris Hall and Niels Ferguson, all from Counterpane Systems, Doug Whiting of Hi/fn, and David Wagner from the University of California Berkeley. It is not covered by any patents, and the Twofish homepage contains the following statement: "Twofish is unpatented, and the source code is uncopyrighted and license-free; it is free for all uses." Twofish is defined in [TWOFISH-1] and [TWOFISH-2]. The Twofish homepage is: http://www.counterpane.com/twofish.html.

2.7 Performance

For a comparison table of the estimated speeds of these and other cipher algorithms, please see [PERF-1], [PERF-2], [PERF-3], or [PERF-4]. The AES homepage, http://www.nist.gov/aes, has pointers to other analyses. The individual cypher documents, [MARS-1], [MARS-2], [RC6], [RIJNDAEL], [SERPENT-1], [SERPENT-2], [TWOFISH-1] and [TWOFISH-2] also contain performance statistics.

3. ESP Payload

The ESP payload is made up of the IV followed by raw cipher-text. Thus the payload field, as defined in [ESP], is broken down according to the following diagram:

```
 +---------------+---------------+---------------+---------------+
 |                                                               |
 +               Initialization Vector (16 octets)               +
 |                                                               |
 +---------------+---------------+---------------+---------------+
 |                                                               |
 ~ Encrypted Payload (variable length, a multiple of 16 octets)  ~
 |                                                               |
 +---------------------------------------------------------------+
```

The IV field MUST be the same size as the block size of the cipher algorithm being used. The IV MUST be chosen at random. Common practice is to use random data for the first IV and the last block of encrypted data from an encryption process as the IV for the next encryption process.

Including the IV in each datagram ensures that decryption of each received datagram can be performed, even when some datagrams are dropped, or datagrams are re-ordered in transit.

To avoid ECB encryption of very similar plaintext blocks in different packets, implementations MUST NOT use a counter or other low-Hamming distance source for IVs.

3.1 ESP Algorithmic Interactions

Currently, there are no known issues regarding interactions between these algorithms and other aspects of ESP, such as use of certain authentication schemes.

3.2 Keying Material

The minimum number of bits sent from the key exchange protocol to the ESP algorithm must be greater than or equal to the key size.

The cipher's encryption and decryption key is taken from the first <x> bits of the keying material, where <x> represents the required key size.

3.3 IKE Interactions

To facilitate the experimental use of the AES candidate ciphers, it would be useful to temporarily define standard IPsec ESP Transform Identifiers for each of the AES algorithms. [DOI] reserves the values 249-255 for "private use amongst cooperating systems." The following IPsec ESP Transform Identifiers are suggested for IKE interoperability using the AES candidate ciphers:

Transform ID	Value
ESP_AES_MARS	249
ESP_AES_RC6	250
ESP_AES_RIJNDAEL	251
ESP_AES_SERPENT	252
ESP_AES_TWOFISH	253

Since the AES candidate ciphers allow variable key lengths, the Key Length attribute MUST be specified in a Phase 2 exchange [DOI]. The Key Length attribute MAY be specified in a Phase 1 exchange [IKE]; if it is not specified, the default key length is 128 bits.

If IKE is used to negotiate keys for the AES candidate ciphers, the recommended characteristics of the groups governing the Diffie-Hellman exchange are as follows:

Key Size	Exponent Size	Modulus Size	Group Type
128	256	3240	MODP
192	384	7945	MODP
256	512	15430	MODP
128	248	248	EC2N
192	376	376	EC2N
256	504	504	EC2N

NOTE: This table is based on Section 4.5 in [KEYLEN-1] and on email communications with Hilarie Orman [KEYLEN-2].

Additional information about the relationship between the group governing a Diffie-Hellman exchange and the symmetric keys derived from the exchange can be found in [KEYLEN-1].

For symmetric key lengths that exceed the output of the hash used to generate the key, the Diffie-Hellman shared secret MUST be hashed twice, and the resulting values combined to form the keying material [KEYLEN-2], as follows:

```
P1 = Hash(0|shared_secret)
P2 = Hash(1|shared_secret)

keying_material = (P1 << shift_bits XOR P2)
```

The first hash output, P1, is shifted left a variable number of bits, depending upon the hash and the key length, prior to XOR'ing it with the second hash output, P2.

Key Size	Hash	Dual DH?	# of Shift Bits
128	MD5	N	-
128	SHA-1	N	-
192	MD5	Y	64
192	SHA-1	Y	32
256	MD5	Y	128
256	SHA-1	Y	96

If additional keying material is required for an authentication key, IKE's iterative key-boosting algorithm MUST be used [IKE, Section 6.2].

4. Security Considerations

Implementations are encouraged to use the largest key sizes they can when taking into account performance considerations for their particular hardware and software configuration. Note that encryption necessarily impacts both sides of a secure channel, so such consideration must take into account not only the client side, but the server as well.

Because these candidate AES algorithms are relatively new and have only undergone limited cryptographic analysis, their use in IPsec implementations should be considered experimental. Once NIST has published the AES FIPS, and at the recommendation of cryptographic experts, AES should become a default and mandatory-to-implement cipher algorithm for IPsec.

For more information regarding the necessary use of random IV values, see [CRYPTO-B].

For further security considerations, the reader is encouraged to read the documents that describe the actual cipher algorithms.

5. Intellectual Property Rights Statement

Pursuant to the provisions of [RFC-2026], the authors represent that they have disclosed the existence of any proprietary or intellectual property rights in the contribution that are reasonably and personally known to the authors. The authors do not represent that they personally know of all potentially pertinent proprietary and intellectual property rights owned or claimed by the organizations they represent or third parties.

The IETF takes no position regarding the validity or scope of any intellectual property or other rights that might be claimed to pertain to the implementation or use of the technology described in this document or the extent to which any license under such rights might or might not be available; neither does it represent that it has made any effort to identify any such rights. Information on the IETF's procedures with respect to rights in standards-track and standards-related documentation can be found in BCP-11. Copies of claims of rights made available for publication and any assurances of licenses to be made available, or the result of an attempt made to obtain a general license or permission for the use of such proprietary rights by implementers or users of this specification can be obtained from the IETF Secretariat.

6. Acknowledgments

Portions of this text, as well as its general structure, were unabashedly lifted from [CBC].

The authors want to thank Hilarie Orman for providing expert advice (and a sanity check) on key sizes, requirements for Diffie-Hellman groups, and IKE interactions.

7. References

[ARCH] Kent, S. and R. Atkinson, "Security Architecture for the Internet Protocol", RFC 2401, November 1998.

[CBC] Pereira, R. and R. Adams, "The ESP CBC-Mode Cipher Algorithms," RFC 2451, November 1998.

[CRYPTO-B] Bellovin, S., "Probable Plaintext Cryptanalysis of the IP Security Protocols", Proceedings of the Symposium on Network and Distributed System Security, San Diego, CA, pp. 155-160, February 1997.
http://www.research.att.com/~smb/probtxt.{ps, pdf}).

[CRYPTO-M] A. Menezes, P. Van Oorschot, S. Vanstone, "Handbook of Applied Cryptography", CRC Press, 1997, ISBN 0-8493-8523-7.

[CRYPTO-S] B. Schneier, "Applied Cryptography Second Edition", John Wiley & Sons, New York, NY, 1995, ISBN 0-471-12845-7.

[DOI] Piper, D., "The Internet IP Security Domain of Interpretation for ISAKMP," RFC 2407, November 1998.

[ESP] Kent, S. and R. Atkinson, "IP Encapsulating Security Payload (ESP)", RFC 2406, November 1998.

[EVALUATION] Ferguson, N. and B. Schneier, "A Cryptographic Evaluation of IPsec," Counterpane Internet Security, Inc., January 2000.

[IKE] Harkins, D. and D. Carrel, "The Internet Key Exchange (IKE)", draft-ietf-ipsec-ike-01.txt, May 1999.

[IKE-ECC] Panjwani, P. and Y. Poeluev, "Additional ECC Groups For IKE," draft-ietf-ipsec-ike-ecc-groups-01.txt, Septermber 1999.

[ISAKMP] Maughan, D., M. Schertler, M. Schneider, and J. Turner, "The Internet Security Association and Key Management Protocol (ISAKMP),"

[KEYLEN-1] Orman, H. and P. Hoffman, "Determining Strengths For Public Keys Used For Exchanging Symmetric Keys," draft-orman-public-key-lengths-00.txt, February 2000.

[KEYLEN-2] Orman, H., email communications, February 2000.

[MARS-1] Burwick, C., D. Coppersmith, E. D'Avignon, R. Gennaro, S. Halevi, C. Jutla, S. Matyas Jr., L. O'Connor, M. Peyravian, D. Safford, and N. Zunic, "MARS - a candidate cipher for AES," NIST AES Proposal, Jun 1998.
http://csrc.nist.gov/encryption/aes/round2/AESAlgs/MARS/mars.pdf
http://www.research.ibm.com/security/mars.html

[MARS-2] Burwick, C., D. Coppersmith, E. D'Avignon, R. Gennaro, S. Halevi, C. Jutla, S. Matyas Jr., L. O'Connor, M. Peyravian, D. Safford, and N. Zunic, "The MARS Encryption Algorithm," NIST AES Proposal, Jun 1998.
http://csrc.nist.gov/encryption/aes/round2/AESAlgs/MARS/mars-int.pdf

[MARS-3] Zunic, N., "Suggested 'tweaks' for the MARS cipher," NIST AES Proposal, May 1999.
http://csrc.nist.gov/encryption/aes/round2/AESAlgs/MARS/mars-twk.txt

[PERF-1] Bassham, L. III, "Efficiency Testing of ANSI C Implementations of Round1 Candidate Algorithms for the Advanced Encryption Standard".
http://csrc.nist.gov/encryption/aes/round1/r1-ansic.pdf

```
   [PERF-2]    Lipmaa, Helger, "Efficiency Testing Table."
   http://home.cyber.ee/helger/aes

   [PERF-3]    Nechvetal, J., E. Barker, D. Dodson, M. Dworkin, J.
               Foti and E. Roback, "Status Report on the First Round
               of the Development of the Advanced Encryption
               Standard".
   http://csrc.nist.gov/encryption/aes/round1/r1report.pdf

   [PERF-4]    Schneier, B., J. Kelsey, D. Whiting, D. Wagner, C.
               Hall, and N. Ferguson, "Performance Comparison of the
               AES Submissions."
   http://www.counterpane.com/AES-performance.html

   [RC6]       Rivest, R., M. Robshaw, R. Sidney, and Y. Yin, "The
               RC6[TM] Block Cipher," NIST AES Proposal, Jun 1998.
   http://csrc.nist.gov/encryption/aes/round2/AESAlgs/RC6/cipher.pdf

   [RFC-2026]  Bradner, S., "The Internet Standards Process --
               Revision 3", RFC2026, October 1996.

   [RFC-2119]  Bradner, S., "Key words for use in RFCs to Indicate
               Requirement Levels", RFC-2119, March 1997.

   [RIJNDAEL]  Daemen, J. and V. Rijman, "AES Proposal: Rijndael,"
               NIST AES Proposal, Jun 1998.
http://csrc.nist.gov/encryption/aes/round2/AESAlgs/Rijndael/Rijndael.pdf

   [ROAD]      Thayer, R., N. Doraswamy and R. Glenn, "IP Security
               Document Roadmap", RFC 2411, November 1998.

   [SERPENT-1] Anderson, R., E. Biham, and L. Knudsen, "Serpent: A
               Proposal for the Advanced Encryption Standard," NIST
               AES Proposal, Jun 1998.
  http://csrc.nist.gov/encryption/aes/round2/AESAlgs/Serpent/Serpent.pdf

   [SERPENT-2] Biham, E., R. Anderson, L. Knudsen, "Serpent: A New
               Block Cipher Proposal," Fast Software Encryption -
               FSE98, Springer LNCS, vol. 1372, pp. 222-238.

   [TWOFISH-1] Schneier, B., J. Kelsey, D. Whiting, D. Wagner, C.
               Hall, and N. Ferguson, "Twofish: A 128-Bit Block
               Cipher," NIST AES Proposal, Jun 1998.
  http://csrc.nist.gov/encryption/aes/round2/AESAlgs/Twofish/Twofish.pdf

   [TWOFISH-2] Schneier, B., J. Kelsey, D. Whiting, D. Wagner, C.
               Hall, and N. Ferguson, "The Twofish Encryption
               Algorithm: A 128-Bit Block Cipher," John Wiley & Sons,
               1999.
```

http://www.counterpane.com/ipsec.html

8. Authors' Addresses

Sheila Frankel
NIST
820 West Diamond Ave.
Room 680
Gaithersburg, MD 20899
Phone: +1 (301) 975-3297
Email: sheila.frankel@nist.gov

Rob Glenn
NIST
820 West Diamond Ave.
Room 455
Gaithersburg, MD 20899
Phone: +1 (301) 975-3667
Email: rob.glenn@nist.gov

Scott Kelly
RedCreek Communications
3900 Newpark Mall Road
Newark, CA 94560
Phone: +1 (510) 745-3969
Email: skelly@redcreek.com

The IPsec working group can be contacted through the chair:

Ted T'so
Massachusetts Institute of Technology
e-mail: tytso@mit.edu

9. Full Copyright Statement