Internet Engineering Task Force (IETF)
Request for Comments: 5669
Category: Standards Track
ISSN: 2070-1721

S. Yoon
J. Kim
H. Park
H. Jeong
Y. Won
Korea Internet & Security Agency

August 2010

The SEED Cipher Algorithm and Its Use with the Secure Real-Time Transport Protocol (SRTP)

Abstract

This document describes the use of the SEED block cipher algorithm in the Secure Real-time Transport Protocol (SRTP) for providing confidentiality for Real-time Transport Protocol (RTP) traffic and for the control traffic for RTP, the Real-time Transport Control Protocol (RTCP).

Status of This Memo

This is an Internet Standards Track document.

This document is a product of the Internet Engineering Task Force (IETF). It represents the consensus of the IETF community. It has received public review and has been approved for publication by the Internet Engineering Steering Group (IESG). Further information on Internet Standards is available in Section 2 of RFC 5741.

Information about the current status of this document, any errata, and how to provide feedback on it may be obtained at http://www.rfc-editor.org/info/rfc5669.

Table of Contents

1. Introduction 3
   1.1. SEED 3
   1.2. Terminology 3
   1.3. Definitions 3
2. Cryptographic Transforms 4
   2.1. Counter 4
        2.1.1. Message Authentication/Integrity: HMAC-SHA1 4
   2.2. Counter with CBC-MAC (CCM) 4
   2.3. Galois/Counter Mode (GCM) 6
3. Nonce Format for CCM and GCM 6
   3.1. Nonce for SRTP 6
   3.2. Nonce for SRTCP 6
4. Key Derivation: SEED-CTR PRF 7
5. Mandatory-to-Implement Transforms 7
6. Security Considerations 7
7. IANA Considerations 8
8. Acknowledgements 8
9. References 8
   9.1. Normative References 8
   9.2. Informative References 9
Appendix A. Test Vectors 10
   A.1. SEED-CTR Test Vectors 10
   A.2. SEED-CCM Test Vectors 11
   A.3. SEED-GCM Test Vectors 12

1. Introduction

This document describes the use of the SEED [RFC4269] block cipher algorithm in the Secure Real-time Transport Protocol (SRTP) [RFC3711] for providing confidentiality for Real-time Transport Protocol (RTP) [RFC3550] traffic and for the control traffic for RTP, the Real-time Transport Control Protocol (RTCP) [RFC3550].

1.1. SEED

SEED is a symmetric encryption algorithm that was developed by the Korea Information Security Agency (KISA) and a group of experts, beginning in 1998. The input/output block size of SEED is 128-bit and the key length is also 128-bit. SEED has the 16-round Feistel structure. A 128-bit input is divided into two 64-bit blocks and the right 64-bit block is an input to the round function with a 64-bit subkey generated from the key scheduling.

SEED is easily implemented in various software and hardware because it is designed to increase the efficiency of memory storage and the simplicity of generating keys without degrading the security of the algorithm. In particular, it can be effectively adopted in a computing environment that has restricted resources such as mobile devices, smart cards, and so on.

SEED is a national industrial association standard [TTASSEED] and is widely used in South Korea for electronic commerce and financial services operated on wired and wireless PKI.

The algorithm specification and object identifiers are described in [RFC4269]. The SEED homepage, http://seed.kisa.or.kr/eng/main.jsp, contains a wealth of information about SEED, including detailed specification, evaluation report, test vectors, and so on.

1.2. Terminology

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in [RFC2119].

1.3. Definitions

|| concatenation
XOR exclusive or

2. Cryptographic Transforms

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

SEED does not have any restrictions for modes of operation that are used with this block cipher. We define three modes of running SEED: (1) SEED in counter mode, (2) SEED in counter mode with CBC-MAC (CCM), and (3) SEED in Galois/Counter Mode (GCM).

2.1. Counter

Section 4.1.1 of [RFC3711] defines AES counter mode encryption, which that document refers to as AES-CM. SEED counter mode is defined in a similar manner and is denoted as SEED-CTR. The plaintext inputs to the block cipher are formed as in AES-CM, and the block cipher outputs are processed as in AES-CM. The only difference in the processing is that SEED-CTR uses SEED as the underlying encryption primitive. When SEED-CTR is used, it MUST be used only in conjunction with an authentication function.

2.1.1. Message Authentication/Integrity: HMAC-SHA1

HMAC-SHA1 [RFC2104], as defined in Section 4.2.1 of [RFC3711], SHALL be the default message-authentication code to be used with SEED-CTR. The default session-authentication key length SHALL be 160 bits, the default authentication tag length SHALL be 80 bits, and the SRTP_PREFIX_LENGTH SHALL be zero for HMAC-SHA1. For SRTP, smaller values are NOT RECOMMENDED but MAY be used after careful consideration of the issues discussed in Sections 7.5 and 9.5 of [RFC3711].

2.2. Counter with CBC-MAC (CCM)

CCM [RFC3610] is a generic authenticate-and-encrypt block cipher mode. In this specification, CCM used with the SEED block cipher is denoted as SEED-CCM.

Section 3.3 of [RFC3711] defines procedures to construct or to authenticate and decrypt SRTP packets. For SEED-CCM, however, the sender performs Step 7 before Step 5 and the receiver performs the second half of Step 5 (verification) after Step 6. This means that authentication is performed on the plaintext rather than the ciphertext. This applies equally to SRTCP.

All SRTP packets MUST be authenticated and encrypted. Unlike SRTP, Secure Real-time Transport Control Protocol (SRTCP) packet encryption is optional (but authentication is mandatory). A sender can select which packets to encrypt and indicates this choice with a 1-bit encryption flag (located in the leftmost bit of the 32-bit word that contains the SRTCP index).

SEED-CCM has two parameters:

M M indicates the size of the authentication tag. In SRTP, a full 80-bit authentication tag SHOULD be used and implementation of this specification MUST support M values of 10 octets.

L L indicates the size of the length field in octets. The number of octets in the nonce MUST be 12, i.e., L is 3.

SEED-CCM has four inputs:

Key

A single key is used to calculate the authentication tag (using CBC-MAC) and to perform payload encryption using counter mode. SEED only supports a key size of 128 bits.

Nonce

The size of the nonce depends on the value selected for the parameter L. It is 15-L octets. L equals 3, and hence the nonce size equals 12 octets.

Plaintext

In the case of SRTP, the payload of the RTP packet, the RTP padding, and the RTP pad count field (if the latter two fields are present) are treated as plaintext.

In the case of SRTCP, when the encryption flag is set to 1, the Encrypted Portion described in Fig.2 of [RFC3711] is treated as plaintext. When the encryption flag is set to 0, the plaintext is zero-length.

Additional Authentication Data (AAD)

In the case of SRTP, the header of the RTP packet, including the contributing source (CSRC) identifier (if present) and the RTP header extension (if present), is considered AAD.

In the case of SRTCP, when the encryption flag is set to 0, the Authentication Portion described in Fig.2 of [RFC3711] is treated as AAD. When the encryption flag is set to 1, the first 8 octets, the encryption flag, and the SRTCP index are treated as AAD.

SEED-CCM accepts these four inputs and returns a ciphertext field.

2.3. Galois/Counter Mode (GCM)

GCM is a block cipher mode of operation providing both confidentiality and data origin authentication [GCM]. GCM used with the SEED block cipher is denoted as SEED-GCM.

SEED-GCM has four inputs: a key, a plaintext, a nonce, and the additional authenticated data (AAD), all described in Section 2.2.

The bit length of the tag, denoted t, is 12, and an authentication tag with a length of 12 octets (96 bits) is used.

3. Nonce Format for CCM and GCM

3.1. Nonce for SRTP

The nonce for SRTP SHALL be formed in the following way:

```
Nonce = (16 bits of zeroes || SSRC || ROC || SEQ) XOR Salt
```

The 4-octet SSRC and the 2-octet SEQ SHALL be taken from the RTP header. The 4-octet ROC is from the cryptographic context. The 12-octet Salt SHALL be produced by the SRTP key derivation function.

3.2. Nonce for SRTCP

The nonce for SRTCP SHALL be formed in the following way:

```
Nonce = (16 bits of zeroes || SSRC || 16 bits of zeroes ||
         SRTCP index) XOR Salt
```

The 4-octet SSRC SHALL be taken from the RTCP header and the 31-bit SRTCP index (packed zero-filled and right-justified into a 4-octet field) is from each packet. The 12-octet Salt SHALL be produced by the SRTP key derivation function.

4. Key Derivation: SEED-CTR PRF

Section 4.3.3 of [RFC3711] defines the AES-128 counter mode key derivation function, which it refers to as "AES-CM PRF". The SEED-CTR PRF is defined in a similar manner, but with each invocation of AES replaced with an invocation of SEED.

The currently defined PRF, keyed by the 128-bit master key, has input block size m = 128 and can produce n-bit outputs for n up to 2^23. SEED-PRF_n(k_master, x) SHALL be SEED in counter mode, as described in Section 2.1; it SHALL be applied to key k_master, have IV equal to (x*2^16), and have the output keystream truncated to the first n (leftmost) bits.

5. Mandatory-to-Implement Transforms

"Mandatory-to-implement" means conformance to this specification, and that Table 1 in this document does not supercede a similar table in Section 5 of [RFC3711]. An RTP implementation that supports SEED MUST implement the modes listed in Table 1 of this document.

	mandatory-to-implement	optional
encryption	SEED-CTR	SEED-CCM,SEED-GCM
message integrity	HMAC-SHA1	SEED-CCM,SEED-GCM
key derivation (PRF)	SEED-CTR	-

Table 1: Mandatory-to-implement and optional transforms in SRTP and SRTCP

6. Security Considerations

No security problem has been found on SEED. SEED is secure against all known attacks, including differential cryptanalysis, linear cryptanalysis, and related key attacks. The best known attack is only an exhaustive search for the key. For further security considerations, the reader is encouraged to read [SEED-EVAL].

See [RFC3610] and [GCM] for security considerations regarding the CCM and GCM Modes of Operation, respectively. In the context of SRTP, the procedures in [RFC3711] ensure the critical property of non-reuse of counter values.

7. IANA Considerations

[RFC4568] defines SRTP "crypto suites". In order to allow the Session Description Protocol (SDP) to signal the use of the algorithms defined in this document, IANA has registered the following crypto suites into the subregistry for SRTP crypto suites under the Media Stream Transports of the SDP Security Descriptions:

SEED_CTR_128_HMAC_SHA1_80
SEED_128_CCM_80
SEED_128_GCM_96

8. Acknowledgements

The authors would like to thank David McGrew, Eric Rescorla, Alexey Melnikov, Alfred Hoenes, Colin Perkins, Young-Chan Shin, the AVT WG (in particular, the chairmen Roni Even and Tom Taylor), and the Real-time Applications and Infrastructure Area Directors for their reviews and support.

9. References

9.1. Normative References

[GCM] Dworkin, M., "NIST Special Publication 800-38D: Recommendation for Block Cipher Modes of Operation: Galois/Counter Mode (GCM) and GMAC", U.S. National Institute of Standards and Technology, http://csrc.nist.gov/publications/nistpubs/800-38D/SP-800-38D.pdf

[RFC2104] Krawczyk, H., Bellare, M., and R. Canetti, "HMAC: Keyed-Hashing for Message Authentication", RFC 2104, February 1997.

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

[RFC3550] Schulzrinne, H., Casner, S., Frederick, R., and V. Jacobson, "RTP: A Transport Protocol for Real-Time Applications", STD 64, RFC 3550, July 2003.

[RFC3610] Whiting, D., Housley, R., and N. Ferguson, "Counter with CBC-MAC (CCM)", RFC 3610, September 2003.

[RFC3711] Baugher, M., McGrew, D., Naslund, M., Carrara, E., and K. Norrman, "The Secure Real-time Transport Protocol (SRTP)", RFC 3711, March 2004.

[RFC4269] Lee, H., Lee, S., Yoon, J., Cheon, D., and J. Lee, "The SEED Encryption Algorithm", RFC 4269, December 2005.

[RFC4568] Andreasen, F., Baugher, M., and D. Wing, "Session Description Protocol (SDP) Security Descriptions for Media Streams", RFC 4568, July 2006.

[TTASSEED] Telecommunications Technology Association (TTA), South Korea, "128-bit Symmetric Block Cipher (SEED)", TTAS.KO-12.0004/R1, December 2005, (In Korean) http://www.tta.or.kr/English/index.jsp.

9.2. Informative References

[SEED-EVAL] KISA, "Self Evaluation Report", http://seed.kisa.or.kr/eng/main.jsp

Appendix A. Test Vectors

All values are in hexadecimal.

A.1. SEED-CTR Test Vectors

Session Key:	0c5ffd37a11edc42c325287fc0604f2e
Rollover Counter:	00000000
Sequence Number:	315e
SSRC:	20e8f5eb
Authentication Key:	f93563311b354748c978913795530631
Session Salt:	cd3a7c42c671e0067a2a2639b43a
Initialization Vector:	cd3a7c42e69915ed7a2a263985640000
RTP Payload:	f57af5fd4ae19562976ec57a5a7ad55a 5af5c5e5c5fdf5c55ad57a4a7272d572 62e9729566ed66e97ac54a4a5a7ad5e1 5ae5fdd5fd5ac5d56ae56ad5c572d54a e54ac55a956afd6aed5a4ac562957a95 16991691d572fd14e97ae962ed7a9f4a 955af572e162f57a956666e17ae1f54a 95f566d54a66e16e4afd6a9f7ae1c5c5 5ae5d56afde916c5e94a6ec56695e14a fde1148416e94ad57ac5146ed59d1cc5
Encrypted RTP Payload:	df5a89291e7e383e9beff765e691a737 49c9e33139ad3001cd8da73ad07f69a2 805a70358b5c7c8c60ed359f95cf5e08 f713c53ff7b808250d79a19ccb8d1073 4e3cb72ed1f0a4e85b002b248049ab07 63dbe571bec52cf9153fdf2019e421ef 779cd6f4bd1c8211da8c272e2fce4393 4b9eabb87362510f254149f992599036 f5e43102327db1ac5e78adc4f66546ed 7abfb5a4db320fb7b9c52a61bc554e44
Authentication Tag:	a5cdaa4d9edc53763855

A.2. SEED-CCM Test Vectors

Key:	974bee725d44fc3992267b284c3c6750
Rollover Counter:	00000000
Sequence Number:	315e
SSRC:	20e8f5eb
Nonce:	000020e8f5eb00000000315e
Payload:	f57af5fd4ae19562976ec57a5a7ad55a 5af5c5e5c5fdf5c55ad57a4a7272d572 62e9729566ed66e97ac54a4a5a7ad5e1 5ae5fdd5fd5ac5d56ae56ad5c572d54a e54ac55a956afd6aed5a4ac562957a95 16991691d572fd14e97ae962ed7a9f4a 955af572e162f57a956666e17ae1f54a 95f566d54a66e16e4afd6a9f7ae1c5c5 5ae5d56afde916c5e94a6ec56695e14a fde1148416e94ad57ac5146ed59d1cc5
AAD:	8008315ebf2e6fe020e8f5eb
Encrypted RTP Payload:	486843a881df215a8574650ddabf5dbb 2650f06f51252bccaeb4012899d6d71e 30c64dad5ead5d8ba65ffe9d79aaf30d c9e6334490c07e7533d704114a9006ec b3b3bff59ecf585485bc0bd286ed434c fd684d19a1ad514ca5f37b71d93288c0 7cf4d5e9b83db8becc8c692a7279b6a9 ac62ba970fc54f46dcc926d434c0b5ad 8678fbf0e7a03037924dae342ef64fa6 5b8eaea260fecb477a57e3919c5dab82
Authentication Tag:	b0a8274cf6a8bb6cc466

A.3. SEED-GCM Test Vectors

Key:	e91e5e75da65554a48181f3846349562
Rollover Counter:	00000000
Sequence Number:	315e
SSRC:	20e8f5eb
Nonce:	000020e8f5eb00000000315e
Payload:	f57af5fd4ae19562976ec57a5a7ad55a 5af5c5e5c5fdf5c55ad57a4a7272d572 62e9729566ed66e97ac54a4a5a7ad5e1 5ae5fdd5fd5ac5d56ae56ad5c572d54a e54ac55a956afd6aed5a4ac562957a95 16991691d572fd14e97ae962ed7a9f4a 955af572e162f57a956666e17ae1f54a 95f566d54a66e16e4afd6a9f7ae1c5c5 5ae5d56afde916c5e94a6ec56695e14a fde1148416e94ad57ac5146ed59d1cc5
AAD:	8008315ebf2e6fe020e8f5eb
Encrypted RTP Payload:	8a5363682c6b1bbf13c0b09cf747a551 2543cb2f129b8bd0e92dfadf735cda8f 88c4bbf90288f5e58d20c4f1bb0d5844 6ea009103ee57ba99cdeabaaa18d4a9a 05ddb46e7e5290a5a2284fe50b1f6fe9 ad3f1348c354181e85b24f1a552a1193 cf0e13eed5ab95ae854fb4f5b0edb2d3 ee5eb238c8f4bfb136b2eb6cd7876042 0680ce1879100014f140a15e07e70133 ed9cbb6d57b75d574acb0087eefbac99
Authentication Tag:	36cd9ae602be3ee2cd8d5d9d

Authors' Addresses

Seokung Yoon
Korea Internet & Security Agency
IT Venture Tower, Jungdaero 135
Songpa-gu, Seoul, Korea 138-950
EMail: seokung@kisa.or.kr

Joongman Kim
Korea Internet & Security Agency
IT Venture Tower, Jungdaero 135
Songpa-gu, Seoul, Korea 138-950
EMail: seopo@kisa.or.kr

Haeryong Park
Korea Internet & Security Agency
IT Venture Tower, Jungdaero 135
Songpa-gu, Seoul, Korea 138-950
EMail: hrpark@kisa.or.kr

Hyuncheol Jeong
Korea Internet & Security Agency
IT Venture Tower, Jungdaero 135
Songpa-gu, Seoul, Korea 138-950
EMail: hcjung@kisa.or.kr

Yoojae Won
Korea Internet & Security Agency
IT Venture Tower, Jungdaero 135
Songpa-gu, Seoul, Korea 138-950
EMail: yjwon@kisa.or.kr