Network Working Group
Request for Comments: 1851
Category: Experimental

P. Karn
Qualcomm
P. Metzger
Piermont
W. Simpson
Daydreamer

September 1995

The ESP Triple DES Transform

Status of this Memo

This document defines an Experimental Protocol for the Internet community. This does not specify an Internet standard of any kind. Discussion and suggestions for improvement are requested. Distribution of this memo is unlimited.

Abstract

This document describes the Triple DES-CBC security transform for the IP Encapsulating Security Payload (ESP).

Table of Contents

1. Introduction

The Encapsulating Security Payload (ESP) [RFC-1827] provides confidentiality for IP datagrams by encrypting the payload data to be protected. This specification describes the ESP use of a variant of of the Cipher Block Chaining (CBC) mode of the US Data Encryption Standard (DES) algorithm [FIPS-46, FIPS-46-1, FIPS-74, FIPS-81]. This variant, known as Triple DES (3DES), processes each block of the plaintext three times, each time with a different key [Tuchman79].

This document assumes that the reader is familiar with the related document "Security Architecture for the Internet Protocol" [RFC-1825], which defines the overall security plan for IP, and provides important background for this specification.

1.1. Keys

The secret 3DES key shared between the communicating parties is effectively 168-bits long. This key consists of three independent 56-bit quantities used by the DES algorithm. Each of the three 56-bit subkeys is stored as a 64-bit (eight octet) quantity, with the least significant bit of each octet used as a parity bit.

1.2. Initialization Vector

This mode of 3DES requires an Initialization Vector (IV) that is eight octets in length.

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

The method for selection of IV values is implementation dependent.

Notes:

A common acceptable technique is simply a counter, beginning with a randomly chosen value. While this provides an easy method for preventing repetition, and is sufficiently robust for practical use, cryptanalysis may use the rare serendipitous occurrence when a corresponding bit position in the first DES block increments in exactly the same fashion.

Other implementations exhibit unpredictability, usually through a pseudo-random number generator. Care should be taken that the periodicity of the number generator is long enough to prevent repetition during the lifetime of the session key.

1.3. Data Size

The 3DES algorithm operates on blocks of eight octets. This often requires padding after the end of the unencrypted payload data.

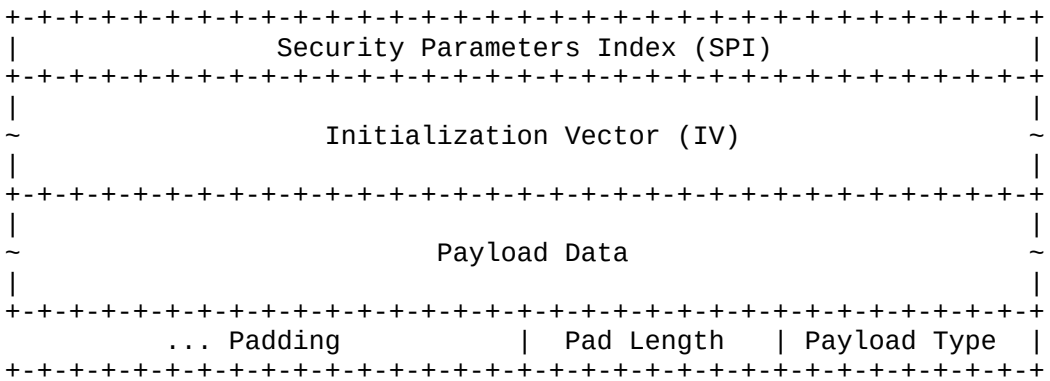
Both input and output result in the same number of octets, which facilitates in-place encryption and decryption.

On receipt, if the length of the data to be decrypted is not an integral multiple of eight octets, then an error is indicated, as described in [RFC-1825].

1.4. Performance

Three DES-CBC implementations may be pipelined in series to provide parallel computation. At the time of writing, at least one hardware implementation can encrypt or decrypt at about 1 Gbps [Schneier94, p. 231].

2. Payload Format



Security Parameters Index (SPI)

A 32-bit value identifying the Security Parameters for this datagram. The value MUST NOT be zero.

Initialization Vector (IV)

The size of this field is variable, although it is constant for all 3DES datagrams of the same SPI and IP Destination. Octets are sent in network order (most significant octet first) [RFC-1700].

The size MUST be a multiple of 32-bits. Sizes of 32 and 64 bits are required to be supported. The use of other sizes is beyond the scope of this specification. The size is expected to be indicated by the key management mechanism.

When the size is 32-bits, a 64-bit IV is formed from the 32-bit value followed by (concatenated with) the bit-wise complement of the 32-bit value. This field size is most common, as it aligns the Payload Data for both 32-bit and 64-bit processing.

All conformant implementations MUST also correctly process a 64-bit field size. This provides strict compatibility with existing hardware implementations.

It is the intent that the value not repeat during the lifetime of the encryption session key. Even when a full 64-bit IV is used, the session key SHOULD be changed at least as frequently as 2**32 datagrams.

Payload Data

The size of this field is variable.

Prior to encryption and after decryption, this field begins with the IP Protocol/Payload header specified in the Payload Type field. Note that in the case of IP-in-IP encapsulation (Payload Type 4), this will be another IP header.

Padding

The size of this field is variable.

Prior to encryption, it is filled with unspecified implementation dependent (preferably random) values, to align the Pad Length and Payload Type fields at an eight octet boundary.

After decryption, it MUST be ignored.

Pad Length

This field indicates the size of the Padding field. It does not include the Pad Length and Payload Type fields. The value typically ranges from 0 to 7, but may be up to 255 to permit hiding of the actual data length.

This field is opaque. That is, the value is set prior to encryption, and is examined only after decryption.

Payload Type

This field indicates the contents of the Payload Data field, using the IP Protocol/Payload value. Up-to-date values of the IP Protocol/Payload are specified in the most recent "Assigned Numbers" [RFC-1700].

This field is opaque. That is, the value is set prior to encryption, and is examined only after decryption.

For example, when encrypting an entire IP datagram (Tunnel-Mode), this field will contain the value 4, which indicates IP-in-IP encapsulation.

3. Algorithm

The 3DES algorithm is a simple variant on the DES-CBC algorithm. The DES function is replaced by three rounds of that function, an encryption followed by a decryption followed by an encryption, each with independant keys, k1, k2 and k3.

Note that when all three keys (k1, k2 and k3) are the same, 3DES is equivalent to DES-CBC. This property allows the 3DES hardware implementations to operate in DES mode without modification.

For more explanation and implementation information for Triple DES, see [Schneier94].

3.1. Encryption

Append zero or more octets of (preferably random) padding to the plaintext, to make its modulo 8 length equal to 6. For example, if the plaintext length is 41, 5 octets of padding are added.

Append a Pad Length octet containing the number of padding octets just added.

Append a Payload Type octet containing the IP Protocol/Payload value which identifies the protocol header that begins the payload.

Provide an Initialization Vector (IV) of the size indicated by the SPI.

Encrypt the payload with Triple DES (EDE mode), producing a ciphertext of the same length.

Octets are mapped to DES blocks in network order (most significant octet first) [RFC-1700]. Octet 0 (modulo 8) of the payload corresponds to bits 1-8 of the 64-bit DES input block, while octet 7 (modulo 8) corresponds to bits 57-64 of the DES input block.

Construct an appropriate IP datagram for the target Destination, with the indicated SPI, IV, and payload.

The Total/Payload Length in the encapsulating IP Header reflects the length of the encrypted data, plus the SPI, IV, padding, Pad Length, and Payload Type octets.

3.2. Decryption

First, the SPI field is removed and examined. This is used as an index into the local Security Parameter table to find the negotiated parameters and decryption key.

The negotiated form of the IV determines the size of the IV field. These octets are removed, and an appropriate 64-bit IV value is constructed.

The encrypted part of the payload is decrypted using Triple DES (DED mode).

The Payload Type is removed and examined. If it is unrecognized, the payload is discarded with an appropriate ICMP message.

The Pad Length is removed and examined. The specified number of pad octets are removed from the end of the decrypted payload, and the IP Total/Payload Length is adjusted accordingly.

The IP Header(s) and the remaining portion of the decrypted payload are passed to the protocol receive routine specified by the Payload Type field.

Security Considerations

Users need to understand that the quality of the security provided by this specification depends completely on the strength of the Triple DES algorithm, the correctness of that algorithm's implementation, the security of the key management mechanism and its implementation, the strength of the key [CN94], and upon the correctness of the implementations in all of the participating nodes.

Among other considerations, applications may wish to take care not to select weak keys for any of the three DES rounds, although the odds of picking one at random are low [Schneier94, p. 233].

It was originally thought that DES might be a group, but it has been demonstrated that it is not [CW92]. Since DES is not a group, composition of multiple rounds of DES is not equivalent to simply using DES with a different key.

Triple DES with independent keys is not, as naively might be expected, as difficult to break by brute force as a cryptosystem with three times the keylength. A space/time tradeoff has been shown which can brute-force break triple block encryptions in the time

naively expected for double encryption [MH81].

However, 2DES can be broken with a meet-in-the-middle attack, without significantly more complexity than breaking DES requires [ibid], so 3DES with independant keys is actually needed to provide this level of security. An attack on 3DES using two independent keys that is somewhat (sixteen times) faster than any known for independent keys has been shown [OW91].

The cut and paste attack described by [Bell95] exploits the nature of all Cipher Block Chaining algorithms. When a block is damaged in transmission, on decryption both it and the following block will be garbled by the decryption process, but all subsequent blocks will be decrypted correctly. If an attacker has legitimate access to the same key, this feature can be used to insert or replay previously encrypted data of other users of the same engine, revealing the plaintext. The usual (ICMP, TCP, UDP) transport checksum can detect this attack, but on its own is not considered cryptographically strong. In this situation, user or connection oriented integrity checking is needed [RFC-1826].

Although it is widely believed that 3DES is substantially stronger than DES, as it is less amenable to brute force attack, it should be noted that real cryptanalysis of 3DES might not use brute force methods at all. Instead, it might be performed using variants on differential [BS93] or linear [Matsui94] cryptanalysis. It should also be noted that no encryption algorithm is permanently safe from brute force attack, because of the increasing speed of modern computers.

As with all cryptosystems, those responsible for applications with substantial risk when security is breeched should pay close attention to developments in cryptography, and especially cryptanalysis, and switch to other transforms should 3DES prove weak.

Acknowledgements

Some of the text of this specification was derived from work by Randall Atkinson for the SIP, SIPP, and IPv6 Working Groups.

Comments should be submitted to the ipsec@ans.net mailing list.

References

[Bell95] Bellovin, S., "An Issue With DES-CBC When Used Without Strong Integrity", Proceedings of the 32nd IETF, Danvers, MA, April 1995.

[BS93] Biham, E., and Shamir, A., "Differential Cryptanalysis of the Data Encryption Standard", Berlin: Springer-Verlag, 1993.

[CN94] Carroll, J.M., and Nudiati, S., "On Weak Keys and Weak Data: Foiling the Two Nemeses", Cryptologia, Vol. 18 No. 23 pp. 253-280, July 1994.

[CW92] Campbell, K.W., and Wiener, M.J., "Proof that DES Is Not a Group", Advances in Cryptology -- Crypto ’92 Proceedings, Berlin: Springer-Verlag, 1993, pp 518-526.

[FIPS-46] US National Bureau of Standards, "Data Encryption Standard", Federal Information Processing Standard (FIPS) Publication 46, January 1977.

[FIPS-46-1] US National Bureau of Standards, "Data Encryption Standard", Federal Information Processing Standard (FIPS) Publication 46-1, January 1988.

[FIPS-74] US National Bureau of Standards, "Guidelines for Implementing and Using the Data Encryption Standard", Federal Information Processing Standard (FIPS) Publication 74, April 1981.

[FIPS-81] US National Bureau of Standards, "DES Modes of Operation" Federal Information Processing Standard (FIPS) Publication 81, December 1980.

[Matsui94] Matsui, M., "Linear Cryptanalysis method dor DES Cipher," Advances in Cryptology -- Eurocrypt ’93 Proceedings, Berlin: Springer-Verlag, 1994.

[MH81] Merle, R.C., and Hellman, M., "On the Security of Multiple Encryption", Communications of the ACM, v. 24 n. 7, 1981, pp. 465-467.

[OW91] van Oorschot, P.C., and Weiner, M.J. "A Known-Plaintext Attack on Two-Key Triple Encryption", Advances in Cryptology -- Eurocrypt '90 Proceedings, Berlin: Springer-Verlag, 1991, pp. 318-325.

[RFC-1800] Postel, J., "Internet Official Protocol Standards", STD 1, RFC 1800, USC/Information Sciences Institute, July 1995.

[RFC-1700] Reynolds, J., and J. Postel, "Assigned Numbers", STD 2, RFC 1700, USC/Information Sciences Institute, October 1994.

[RFC-1825] Atkinson, R., "Security Architecture for the Internet Protocol", RFC-1825, Naval Research Laboratory, July 1995.

[RFC-1826] Atkinson, R., "IP Authentication Header", RFC-1826, Naval Research Laboratory, July 1995.

[RFC-1827] Atkinson, R., "IP Encapsulating Security Protocol (ESP)", RFC-1827, Naval Research Laboratory, July 1995.

[Schneier94] Schneier, B., "Applied Cryptography", John Wiley & Sons, New York, NY, 1994. ISBN 0-471-59756-2

[Tuchman79] Tuchman, W, "Hellman Presents No Shortcut Solutions to DES", IEEE Spectrum, v. 16 n. 7, July 1979, pp. 40-41.

Author's Address

Questions about this memo can also be directed to:

Phil Karn
Qualcomm, Inc.
6455 Lusk Blvd.
San Diego, California 92121-2779

karn@unix.ka9q.ampr.org

Perry Metzger
Piermont Information Systems Inc.
160 Cabrini Blvd., Suite #2
New York, NY 10033

perry@piermont.com

William Allen Simpson
Daydreamer
Computer Systems Consulting Services
1384 Fontaine
Madison Heights, Michigan 48071

Bill.Simpson@um.cc.umich.edu
bsimpson@MorningStar.com