Network Working Group
Request for Comments: 1829
Category: Standards Track

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

August 1995

The ESP DES-CBC Transform

Status of this Memo

This document specifies an Internet standards track protocol for the Internet community, and requests discussion and suggestions for improvements. Please refer to the current edition of the "Internet Official Protocol Standards" (STD 1) for the standardization state and status of this protocol. Distribution of this memo is unlimited.

Abstract

This document describes the 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 the Cipher Block Chaining (CBC) mode of the US Data Encryption Standard (DES) algorithm [FIPS-46, FIPS-46-1, FIPS-74, FIPS-81].

All implementations that claim conformance or compliance with the Encapsulating Security Payload specification MUST implement this DES-CBC transform.

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 DES key shared between the communicating parties is eight octets in length. This key consists of a 56-bit quantity used by the DES algorithm. The 56-bit key 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 DES 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 DES 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

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)                |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |                                                               |
   ~                   Initialization Vector (IV)                  ~
   |                                                               |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |                                                               |
   ~                          Payload Data                         ~
   |                                                               |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
             ... Padding           |  Pad Length   | Payload Type  |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

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 DES-CBC 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

In DES-CBC, the base DES encryption function is applied to the XOR of each plaintext block with the previous ciphertext block to yield the ciphertext for the current block. This provides for re-synchronization when datagrams are lost.

For more explanation and implementation information for 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 DES in CBC 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 DES in the CBC 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 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, although the odds of picking one at random are low [Schneier94, p 233].

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].

At the time of writing of this document, [BS93] demonstrated a differential cryptanalysis based chosen-plaintext attack requiring 2^47 plaintext-ciphertext pairs, and [Matsui94] demonstrated a linear cryptanalysis based known-plaintext attack requiring only 2^43 plaintext-ciphertext pairs. Although these attacks are not considered practical, they must be taken into account.

More disturbingly, [Weiner94] has shown the design of a DES cracking machine costing $1 Million that can crack one key every 3.5 hours. This is an extremely practical attack.

One or two blocks of known plaintext suffice to recover a DES key. Because IP datagrams typically begin with a block of known and/or guessable header text, frequent key changes will not protect against this attack.

It is suggested that DES is not a good encryption algorithm for the protection of even moderate value information in the face of such equipment. Triple DES is probably a better choice for such purposes.

However, despite these potential risks, the level of privacy provided by use of ESP DES-CBC in the Internet environment is far greater than sending the datagram as cleartext.

Acknowledgements

This document was reviewed by the IP Security Working Group of the Internet Engineering Task Force (IETF). Comments should be submitted to the ipsec@ans.net mailing list.

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

The use of DES for confidentiality is closely modeled on the work done for SNMPv2 [RFC-1446].

Steve Bellovin, Steve Deering, Karl Fox, Charles Lynn, Craig Metz, Dave Mihelcic and Jeffrey Schiller provided useful critiques of earlier versions of this draft.

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.

[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.

[RFC-1446] Galvin, J., and McCloghrie, K., "Security Protocols for Version 2 of the Simple Network Management Protocol (SNMPv2)", RFC-1446, DDN Network Information Center, April 1993.

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

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

[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

[Weiner94]
Wiener, M.J., "Efficient DES Key Search", School of Computer Science, Carleton University, Ottawa, Canada, TR-244, May 1994. Presented at the Rump Session of Crypto ’93.

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