Network Working Group
Request for Comments: 2405
Category: Standards Track

C. Madson
Cisco Systems, Inc.
N. Doraswamy
Bay Networks, Inc.

November 1998

The ESP DES-CBC Cipher Algorithm With Explicit IV

Status of this Memo

Copyright Notice

Abstract

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

1. Introduction

This document describes the use of the DES Cipher algorithm in Cipher Block Chaining Mode as a confidentiality mechanism within the context of the Encapsulating Security Payload.

DES is a symmetric block cipher algorithm. The algorithm is described in [FIPS-46-2][FIPS-74][FIPS-81]. [Schneier96] provides a general description of Cipher Block Chaining Mode, a mode which is applicable to several encryption algorithms.

As specified in this memo, DES-CBC is not an authentication mechanism. [Although DES-MAC, described in [Schneier96] amongst other places, does provide authentication, DES-MAC is not discussed here.]

For further information on how the various pieces of ESP fit together to provide security services, refer to [ESP] and [road].

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 [RFC-2119].

2. Algorithm and Mode

DES-CBC is a symmetric secret-key block algorithm. It has a block size of 64 bits.

[FIPS-46-2][FIPS-74] and [FIPS-81] describe the DES algorithm, while [Schneier96] provides a good description of CBC mode.

2.1 Performance

Phil Karn has tuned DES-CBC software to achieve 10.45 Mbps with a 90 MHz Pentium, scaling to 15.9 Mbps with a 133 MHz Pentium. Other DES speed estimates may be found in [Schneier96].

3. ESP Payload

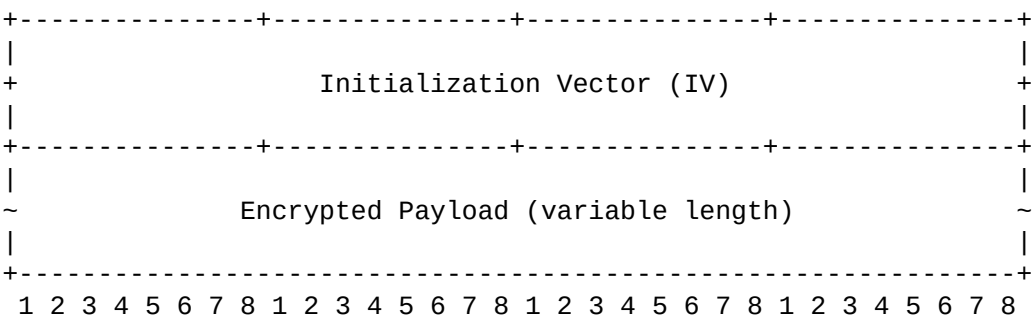
DES-CBC requires an explicit Initialization Vector (IV) of 8 octets (64 bits). This IV immediately precedes the protected (encrypted) payload. The IV MUST be a random value.

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.

Implementation note:

> Common practice is to use random data for the first IV and the last 8 octets of encrypted data from an encryption process as the IV for the next encryption process; this logically extends the CBC across the packets. It also has the advantage of limiting the leakage of information from the random number genrator. No matter which mechnism is used, the receiver MUST NOT assume any meaning for this value, other than that it is an IV.
>
> 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.

The payload field, as defined in [ESP], is broken down according to the following diagram:



3.1 Block Size and Padding

The DES-CBC algorithm described in this document MUST use a block size of 8 octets (64 bits).

When padding is required, it MUST be done according to the conventions specified in [ESP].

4. Key Material

DES-CBC is a symmetric secret key algorithm. The key size is 64-bits. [It is commonly known as a 56-bit key as the key has 56 significant bits; the least significant bit in every byte is the parity bit.]

[arch] describes the general mechanism to derive keying material for the ESP transform. The derivation of the key from some amount of keying material does not differ between the manually- and automatically-keyed security associations.

This mechanism MUST derive a 64-bit key value for use by this cipher. The mechanism will derive raw key values, the derivation process itself is not responsible for handling parity or weak key checks.

Weak key checks SHOULD be performed. If such a key is found, the key SHOULD be rejected and a new SA requested.

Implementation note:

> If an implementation chooses to do weak key checking, it should recognize that the known weak keys [FIPS74] have been adjusted for parity. Otherwise the handling of parity is a local issue.

A strong pseudo-random function MUST be used to generate the required key. For a discussion on this topic, reference [RFC1750].

4.1 Weak Keys

DES has 16 known weak keys, including so-called semi-weak keys. The list of weak keys can be found in [FIPS74].

4.2 Key Lifetime

[Blaze96] discusses the costs and key recovery time for brute force attacks. It presents various combinations of total cost/time to recover a key/cost per key recovered for 40-bit and 56-bit DES keys, based on late 1995 estimates.

While a brute force search of a 56-bit DES keyspace can be considered infeasable for the so-called casual hacker, who is simply using spare CPU cycles or other low-cost resources, it is within reach of someone willing to spend a bit more money.

For example, for a cost of $300,000, a 56-bit DES key can be recovered in an average of 19 days using off-the-shelf technology and in only 3 hours using a custom developed chip.

It should be noted that there are other attacks which can recover the key faster, that brute force attacks are considered the "worst case", although the easiest to implement.

[Wiener94] also discusses a $1M machine which can break a DES key in 3.5 hours (1993 estimates), using a known-plaintext attack. As discussed in the Security Considerations section, a known plaintext attack is reasonably likely.

It should also be noted that over time, the total and average search costs as well as the average key recovery time will continue to drop.

While the above does not provide specific recommendations for key lifetime, it does reinforce the point that for a given application the desired key lifetime is dependent upon the perceived threat (an educated guess as to the amount of resources available to the attacker) relative to the worth of the data to be protected.

While there are no recommendations for volume-based lifetimes made here, it shoud be noted that given sufficient volume there is an increased probabilty that known plaintext can be accumulated.

5. Interaction with Authentication Algorithms

As of this writing, there are no known issues which preclude the use of the DES-CBC algorithm with any specific authentication algorithm.

6. Security Considerations

[Much of this section was originally written by William Allen Simpson and Perry Metzger.]

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 Security Association management mechanism and its implementation, the strength of the key [CN94], and upon the correctness of the implementations in all of the participating nodes.

[Bell95] and [Bell96] describe a cut and paste splicing attack which applies to all Cipher Block Chaining algorithms. This attack can be addressed with the use of an authentication mechanism.

The use of the cipher mechanism without any corresponding authentication mechanism is strongly discouraged. This cipher can be used in an ESP transform that also includes authentication; it can also be used in an ESP transform that doesn't include authentication provided there is an companion AH header. Refer to [ESP], [AH], [arch], and [road] for more details.

When the default ESP padding is used, the padding bytes have a predictable value. They provide a small measure of tamper detection on their own block and the previous block in CBC mode. This makes it somewhat harder to perform splicing attacks, and avoids a possible covert channel. This small amount of known plaintext does not create any problems for modern ciphers.

At the time of writing of this document, [BS93] demonstrated a differential cryptanalysis based chosen-plaintext attack requiring 2^47 plaintext-ciphertext pairs, where the size of a pair is the size of a DES block (64 bits). [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, [Wiener94] 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.

The case for using random values for IVs has been refined with the following summary provided by Steve Bellovin. Refer to [Bell97] for further information.

> "The problem arises if you use a counter as an IV, or some other source with a low Hamming distance between successive IVs, for encryption in CBC mode. In CBC mode, the "effective plaintext" for an encryption is the XOR of the actual plaintext and the ciphertext of the preceeding block. Normally, that's a random value, which means that the effective plaintext is quite random. That's good, because many blocks of actual plaintext don't change very much from packet to packet, either.
>
> For the first block of plaintext, though, the IV takes the place of the previous block of ciphertext. If the IV doesn't differ much from the previous IV, and the actual plaintext block doesn't differ much from the previous packet's, then the effective plaintext won't differ much, either. This means that you have pairs of ciphertext blocks combined with plaintext blocks that differ in just a few bit positions. This can be a wedge for assorted cryptanalytic attacks."

The discussion on IVs has been updated to require that an implementation not use a low-Hamming distance source for IVs.

7. References

[Bell95] Bellovin, S., "An Issue With DES-CBC When Used Without Strong Integrity", Presentation at the 32nd Internet Engineering Task Force, Danvers Massachusetts, April 1995.

[Bell96] Bellovin, S., "Problem Areas for the IP Security Protocols", Proceedings of the Sixth Usenix Security Symposium, July 1996.

[Bell97] 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 (also http://www.research.att.com/~smb/papers/probtxt.{ps, pdf}).

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

[Blaze96] Blaze, M., Diffie, W., Rivest, R., Schneier, B., Shimomura, T., Thompson, E., and M. Wiener, "Minimal Key Lengths for Symmetric Ciphers to Provide Adequate Commercial Security", currently available at http://www.bsa.org/policy/encryption/cryptographers.html.

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

[FIPS-46-2] US National Bureau of Standards, "Data Encryption Standard", Federal Information Processing Standard (FIPS) Publication 46-2, December 1993, http://www.itl.nist.gov/div897/pubs/fip46-2.htm (supercedes FIPS-46-1).

[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, http://www.itl.nist.gov/div897/pubs/fip74.htm.

[FIPS-81] US National Bureau of Standards, "DES Modes of Operation", Federal Information Processing Standard (FIPS) Publication 81, December 1980, http://www.itl.nist.gov/div897/pubs/fip81.htm.

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

[RFC-1750] Eastlake, D., Crocker, S., and J. Schiller, "Randomness Recommendations for Security", RFC 1750, December 1994.

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

[Schneier96] Schneier, B., "Applied Cryptography Second Edition", John Wiley & Sons, New York, NY, 1996. ISBN 0-471-12845-7.

[Wiener94] 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. [Reprinted in "Practical Cryptography for Data Internetworks", W.Stallings, editor, IEEE Computer Society Press, pp.31-79 (1996). Currently available at ftp://ripem.msu.edu/pub/crypt/docs/des-key-search.ps.]

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

[AH] Kent, S., and R. Atkinson, "IP Authentication Header (AH)", RFC 2402, November 1998.

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

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

8. Acknowledgments

Much of the information provided here originated with various ESP-DES documents authored by Perry Metzger and William Allen Simpson, especially the Security Considerations section.

This document is also derived in part from previous works by Jim Hughes, those people that worked with Jim on the combined DES-CBC+HMAC-MD5 ESP transforms, the ANX bakeoff participants, and the members of the IPsec working group.

Thanks to Rob Glenn for assisting with the nroff formatting.

The IPSec working group can be contacted via the IPSec working group's mailing list (ipsec@tis.com) or through its chairs:

Robert Moskowitz
International Computer Security Association

EMail: rgm@icsa.net

Theodore Y. Ts'o
Massachusetts Institute of Technology

EMail: tytso@MIT.EDU

9. Editors' Addresses

Cheryl Madson
Cisco Systems, Inc.

EMail: cmadson@cisco.com

Naganand Doraswamy
Bay Networks, Inc.

EMail: naganand@baynetworks.com

10. Full Copyright Statement