Network Working Group
Internet Draft
expires in six months

W A Simpson [DayDreamer]
R Baldwin [RSA Data Security]

July 1998

The ESP DES-XEX3-CBC Transform
draft-simpson-desx-02.txt

Status of this Memo

Copyright Notice

Abstract

This document describes the "DESX" DES-XEX3-CBC block cipher transform interface used with the IP Encapsulating Security Payload (ESP).

1. Introduction

The Encapsulating Security Payload (ESP) [RFC-1827x] provides confidentiality for IP datagrams by encrypting the payload data to be protected. This specification describes the ESP use of a variant 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, also known as "DESX", processes each block three times, each time with a different key [Kaliski96]. The first and last pass are a simple and fast XOR. This was originally proposed by Ron Rivest in May of 1984 as a computationally cheap mechanism to protect DES against exhaustive key-search attacks.

Although XOR of a constant value over multiple blocks would not normally be considered cryptographically secure, the use of DES-CBC in the middle provides a background of highly random internal chaining. The XOR values are combined with these random blocks to provide a modest improvement in strength.

For an explanation of the use of CBC mode with this cipher, see [RFC-wwww].

For more explanation and implementation information for DESX, see [Schneier95].

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

In this document, the key words "MAY", "MUST", "recommended", "required", and "SHOULD", are to be interpreted as described in [RFC-2119].

1.1. Availability

The DESX algorithm has been previously described in [Kaliski96, Schneier95]. This algorithm is not protected by either patent or trade secret laws, though the DESX name is a trademark of RSA Data Security, a wholly owned subsidary of Security Dynamics Inc. Trademark fair-use laws allow vendors to label a product as being compatible with DESX. An implementation of DESX is available in RSA's BSAFE cryptography toolkit and interoperable implementations have been created outside of the United States.

1.2. Performance

The additional computational cost beyond DES is negligible.

2. Description

2.1. Block Size

The US Data Encryption Standard (DES) algorithm operates on blocks of 64-bits (8 bytes). This often requires padding before encrypting, and subsequent removal of padding after decrypting.

The output is the same number of bytes that are input. This facilitates in-place encryption and decryption.

2.2. Mode

The DES-XEX3-CBC algorithm is a simple variant of the DES-CBC algorithm [RFC-wwww, RFC-1829].

In DES-XEX3-CBC, the algorithms are an XOR (Xk1), followed by a DES encryption (Ek2), followed by another XOR (Xk3), which generates the ciphertext (C1) for the block. Each step uses an independant key: k1, k2 and k3.

To decrypt, the order of the functions is reversed: XOR with k3, DES decrypt with k2, XOR with k1.

Note that when the XOR keys (k1 and k3) are zero, DES-XEX3-CBC is equivalent to DES-CBC. This property allows the DES-XEX3 hardware implementations to operate in DES mode without modification.

2.3. Interaction with Authentication

There is no known interaction of DES with any currently specified Authenticator algorithm. Never-the-less, any Authenticator MUST use a separate and independently generated key.

3. Initialization Vector

DES-XEX3-CBC requires an Initialization Vector (IV) that is 64-bits (8 bytes) in length. By default, the IV is carried immediately following the ESP Sequence Number.

4. Keys

The secret DES-XEX3 keys shared between the communicating parties are effectively 184-bits long, but are represented as a 192-bit (24 byte) quantity.

The keys consist of three independent quantities: a 64-bit key used by an XOR, a 56-bit key used by the DES algorithm, and another 64-bit key used by an XOR. The middle 56-bit key is stored as a 64-bit (8 byte) quantity, with the least significant bit of each byte used as a parity bit.

4.1. Weak Keys

DES has 64 known weak keys, including so-called semi-weak keys and possibly-weak keys [Schneier95, pp 280-282]. The likelihood of picking one at random is negligible.

However, since checking for weak keys is quite easy, conformant implementations MUST test for weak DES keys.

Moreover, the XOR keys MUST NOT be zero.

4.2. Manual Key Management

When configured manually, three independently generated keys are required, in the order used for encryption, and 64-bits (8 bytes) are configured for each individual key.

Keys with incorrect parity SHOULD be rejected by the configuration utility, ensuring that the keys have been correctly configured.

Each key is examined sequentially, in the order used for encryption. A key that is identical to a previous key MUST be rejected. The 64 known weak DES keys MUST be rejected.

4.3. Automated Key Management

When configured via a Security Association management protocol, three independently generated keys are required, in the order used for encryption, and 64-bits (8 bytes) are returned for each individual key.

The key manager MAY be required to generate the correct parity for the DES key. Alternatively, the least significant bit of each key

byte is ignored, or locally set to parity by the DES implementation.

Each key is examined sequentially, in the order used for encryption. A key that is identical to a previous key MUST be rejected. The 64 known weak DES keys (for the DES key) MUST be rejected.

4.4. Refresh Rate

To prevent differential and linear cryptanalysis of collisions [RFC-wwww], no more than 2**32 plaintext blocks SHOULD be encrypted with the same keys. Depending on the average size of the datagrams, the keys SHOULD be changed at least as frequently as 2**30 datagrams.

Operational Considerations

The specification provides only a few manually configurable parameters:

SPI
: Manually configured SPIs are limited in range to aid operations. Automated SPIs are pseudo-randomly distributed throughout the remaining 2**32 values.

  Default: 0 (none). Range: 256 to 65,535.

SPI LifeTime (SPILT)
: Manually configured LifeTimes are generally measured in days. Automated LifeTimes are specified in seconds.

  Default: 32 days (2,764,800 seconds). Maximum: 182 days (15,724,800 seconds).

Key
: A 64-bit key, a 56-bit key with parity included as appropriate, and another 64-bit key, are configured in order as a 192-bit quantity.

Each party configures a list of known SPIs and symmetric secret-keys.

In addition, each party configures local policy that determines what access (if any) is granted to the holder of a particular SPI. For example, a party might allow FTP, but prohibit Telnet. Such considerations are outside the scope of this document.

Security Considerations

Users need to understand that the quality of the security provided by this specification depends completely on the strength of the DESX 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.

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.

It has been shown that DES-XEX3 is substantially stronger than DES alone, as it is less amenable to brute force attack with an exhaustive key search. When the number of plaintext blocks are limited to 2**32 as recommended, the time complexity of the idealized random permutation block cipher model is increased from an order 2**86 (for DES) to 2**134 [Kilian96, Rogaway96].

It should be noted that real cryptanalysis of DES-XEX3 might not use brute force methods at all. Instead, it might be performed using variants on differential [BS93] or linear [Matsui94] cryptanalysis. It has been estimated that differential cryptanalysis is increased from 2**47 (for DES) to 2**61 chosen-plaintext blocks, and linear cryptanalysis is increased from 2**43 (for DES) to 2**60 known-plaintext blocks [Kaliski96]. Although these attacks are not considered practical, this offers only a small improvement over DES alone.

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 cryptology, and especially cryptanalysis, and switch to other transforms should DES-XEX3 prove weak.

Acknowledgements

The basic field naming and layout is based on "swIPe" [IBK93, IB93].

Most of the text of this specification was derived from earlier work by William Allen Simpson and Perry Metzger in multiple Request for Comments.

Use of DES-XEX3 was proposed by William Allen Simpson and various other participants in the IETF IP Security Working Group in 1995 and 1996, but was prevented from publication through disregard of the IETF Standards Process.

References

[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.

[IB93] Ioannidis, J., and Blaze, M., "The Architecture and Implementation of Network-Layer Security Under Unix", Proceedings of the Fourth Usenix Security Symposium, Santa Clara California, October 1993.

[IBK93] Ioannidis, J., Blaze, M., and Karn, P., "swIPe: Network-Layer Security for IP", Presentation at the 26th Internet

Engineering Task Force, Columbus Ohio, March 1993.

[Kaliski96] Kaliski, B., and Robshaw, M., "Multiple Encryption: Weighing Security and Performance", Dr. Dobbs Journal, January 1996.

[Kilian96] Kilian J., and Rogaway, P., "How to protect DES against exhaustive key search", Advances in Cryptology -- Crypto '96 Proceedings, Berlin: Springer-Verlag, 1996, http://wwwcsif.cs.ucdavis.edu/~rogaway.

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

[Rogaway96] Rogaway, P., "The Security of DESX", CryptoBytes, v 2 n 2, RSA Laboratories, Redwood City, CA, USA, Summer 1996.

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

[RFC-1827x] Simpson, W., "IP Encapsulating Security Protocol (ESP) for implementors", work in progress.

[RFC-1829] Karn, P., Metzger, P., Simpson, W.A., "The ESP DES-CBC Transform", August 1995.

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

[RFC-wwww] Simpson, W.A, "ESP with Cipher Block Chaining (CBC)", work in progress.

[Schneier95]
Schneier, B., "Applied Cryptography Second Edition", John Wiley & Sons, New York, NY, 1995. ISBN 0-471-12845-7.

Contacts

Comments about this document should be discussed on the ipsec@tis.com mailing list.

Questions about this document can also be directed to:

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

wsimpson@UMich.edu
wsimpson@GreenDragon.com (preferred)

Robert Baldwin
RSA Data Security Inc.
100 Marine Parkway
Redwood City, California 94065

baldwin@rsa.com

Full Copyright Statement