Network Working Group
Request for Comments: 2410
Category: Standards Track

R. Glenn
NIST
S. Kent
BBN Corp
November 1998

The NULL Encryption Algorithm and Its Use With IPsec

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.

Copyright Notice

Abstract

This memo defines the NULL encryption algorithm and its use with the IPsec Encapsulating Security Payload (ESP). NULL does nothing to alter plaintext data. In fact, NULL, by itself, does nothing. NULL provides the means for ESP to provide authentication and integrity without confidentiality.

Further information on the other components necessary for ESP implementations is provided by [ESP] and [ROAD].

1. Introduction

This memo defines the NULL encryption algorithm and its use with the IPsec Encapsulating Security Payload [ESP] to provide authentication and integrity without confidentiality.

NULL is a block cipher the origins of which appear to be lost in antiquity. Despite rumors that the National Security Agency suppressed publication of this algorithm, there is no evidence of such action on their part. Rather, recent archaeological evidence suggests that the NULL algorithm was developed in Roman times, as an exportable alternative to Ceaser ciphers. However, because Roman numerals lack a symbol for zero, written records of the algorithm's development were lost to historians for over two millennia.

[ESP] specifies the use of an optional encryption algorithm to provide confidentiality and the use of an optional authentication algorithm to provide authentication and integrity. The NULL encryption algorithm is a convenient way to represent the option of not applying encryption. This is referred to as ESP_NULL in [DOI].

The IPsec Authentication Header [AH] specification provides a similar service, by computing authentication data which covers the data portion of a packet as well as the immutable in transit portions of the IP header. ESP_NULL does not include the IP header in calculating the authentication data. This can be useful in providing IPsec services through non-IP network devices. The discussion on how ESP_NULL might be used with non-IP network devices is outside the scope of this document.

In this memo, NULL is used within the context of ESP. 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 Definition

NULL is defined mathematically by the use of the Identity function I applied to a block of data b such that:

$$NULL(b) = I(b) = b$$

2.1 Keying Material

Like other modern ciphers, e.g., RC5 [RFC-2040], the NULL encryption algorithm can make use of keys of varying lengths. However, no measurable increase in security is afforded by the use of longer key lengths.

2.2 Cryptographic Synchronization

Because of the stateless nature of the NULL encryption algorithm, it is not necessary to transmit an IV or similar cryptographic synchronization data on a per packet (or even a per SA) basis. The NULL encryption algorithm combines many of the best features of both block and stream ciphers, while still not requiring the transmission of an IV or analogous cryptographic synchronization data.

2.3 Padding

NULL has a block size of 1 byte, thus padding is not necessary.

2.4. Performance

The NULL encryption algorithm is significantly faster than other commonly used symmetric encryption algorithms and implementations of the base algorithm are available for all commonly used hardware and OS platforms.

2.5 Test Vectors

The following is a set of test vectors to facilitate in the development of interoperable NULL implementations.

```
test_case =      1
data =           0x123456789abcdef
data_len =       8
NULL_data =      0x123456789abcdef

test_case =      2
data =           "Network Security People Have A Strange Sense Of Humor"
data_len =       53
NULL_data =      "Network Security People Have A Strange Sense Of Humor"
```

3. ESP_NULL Operational Requirements

ESP_NULL is defined by using NULL within the context of ESP. This section further defines ESP_NULL by pointing out particular operational parameter requirements.

For purposes of IKE [IKE] key extraction, the key size for this algorithm MUST be zero (0) bits, to facilitate interoperability and to avoid any potential export control problems.

To facilitate interoperability, the IV size for this algorithm MUST be zero (0) bits.

Padding MAY be included on outgoing packets as specified in [ESP].

4. Security Considerations

The NULL encryption algorithm offers no confidentiality nor does it offer any other security service. It is simply a convenient way to represent the optional use of applying encryption within ESP. ESP can then be used to provide authentication and integrity without confidentiality. Unlike AH these services are not applied to any

part of the IP header. At the time of this writing there is no evidence to support that ESP_NULL is any less secure than AH when using the same authentication algorithm (i.e. a packet secured using ESP_NULL with some authentication algorithm is as cryptographically secure as a packet secured using AH with the same authentication algorithm).

As stated in [ESP], while the use of encryption algorithms and authentication algorithms are optional in ESP, it is imperative that an ESP SA specifies the use of at least one cryptographically strong encryption algorithm or one cryptographically strong authentication algorithm or one of each.

At the time of this writing there are no known laws preventing the exportation of NULL with a zero (0) bit key length.

5. Intellectual Property Rights

Pursuant to the provisions of [RFC-2026], the authors represent that they have disclosed the existence of any proprietary or intellectual property rights in the contribution that are reasonably and personally known to the authors. The authors do not represent that they personally know of all potentially pertinent proprietary and intellectual property rights owned or claimed by the organizations they represent or third parties.

6. Acknowledgments

Steve Bellovin suggested and provided the text for the Intellectual Property Rights section.

Credit also needs to be given to the participants of the Cisco/ICSA IPsec & IKE March 1998 Interoperability Workshop since it was there that the need for this document became apparent.

7. References

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

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

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

[DOI] Piper, D., "The Internet IP Security Domain of Interpretation for ISAKMP", RFC 2408, November 1998.

[IKE] Harkins, D., and D. Carrel, "The Internet Key Exchange (IKE)", RFC 2409, November 1998.

[RFC-2026] Bradner, S., "The Internet Standards Process -- Revision 3", BCP 9, RFC 2026, October 1996.

[RFC-2040] Baldwin, R., and R. Rivest, "The RC5, RC5-CBC, RC5-CBC-Pad, and RC5-CTS Algorithms", RFC 2040, October 1996

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

6. Editors' Addresses

Rob Glenn
NIST

EMail: rob.glenn@nist.gov

Stephen Kent
BBN Corporation

EMail: kent@bbn.com

The IPsec working group can be contacted through the chairs:

Robert Moskowitz
ICSA

EMail: rgm@icsa.net

Ted T'so
Massachusetts Institute of Technology

EMail: tytso@mit.edu

7. Full Copyright Statement