IPSEC Working Group
INTERNET-DRAFT

Ashar Aziz
Tom Markson
Hemma Prafullchandra
Sun Microsystems, Inc.

Expires in six months

December 21, 1995

SKIP Algorithm Discovery Protocol
<draft-ietf-ipsec-skip-adp-00.txt>

Status of this Memo

This document is a submission to the IETF Internet Protocol Security (IPSEC) Working Group. Comments are solicited and should be addressed to to the working group mailing list (ipsec@ans.net) or to the authors.

This document is an Internet-Draft. Internet Drafts are working documents of the Internet Engineering Task Force (IETF), its areas, and its working Groups. Note that other groups may also distribute working documents as Internet Drafts.

Internet-Drafts draft documents are valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

To learn the current status of any Internet-Draft, please check the "1id-abstracts.txt" listing contained in the Internet-Drafts Shadow Directories on ftp.is.co.za (Africa), nic.nordu.net (Europe), munnari.oz.au (Pacific Rim), ds.internic.net (US East Coast), or ftp.isi.edu (US West Coast). Distribution of this memo is unlimited.

Abstract

SKIP [1] provides privacy and authentication with Internet Protocols. It does not define a method by which two entities may mutually agree on encryption, authentication and compression algorithms. We describe a protocol which will allow one SKIP entity to inform another entity of the capabilities it supports.

CONTENTS

1. SKIP Algorithm Discovery

SKIP [1] allows two entities to communicate securely with no bilateral state other than the other party's public key. However, different entities may have different encryption, authentication or compression capabilities. The SKIP protocol does not define a method for discovering the algorithms that another entity supports. SKIP Algorithm Discovery enables one entity to inform another of the capabilities it supports.

SKIP Algorithm Discovery is in many ways analogous to algorithm negotiation in conventional session oriented key management schemes. However, "negotiation" is a misnomer as applied to most existing protocols that accommodate this feature. This is because in essence there is no negotiation, simply a statement of capabilities on both sides. The sides agree to pick a common subset of their capabilities.

SKIP Algorithm Discovery allows the same statement of capabilities to occur in a stateless manner, entirely analogous to how the IP protocol performs path MTU discovery. A SKIP implementation is free to choose a set of algorithms with a particular node. If it chooses incorrectly, it will discover this through an authenticated ICMP message, which is in effect a statement of capabilities and preferences for that node.

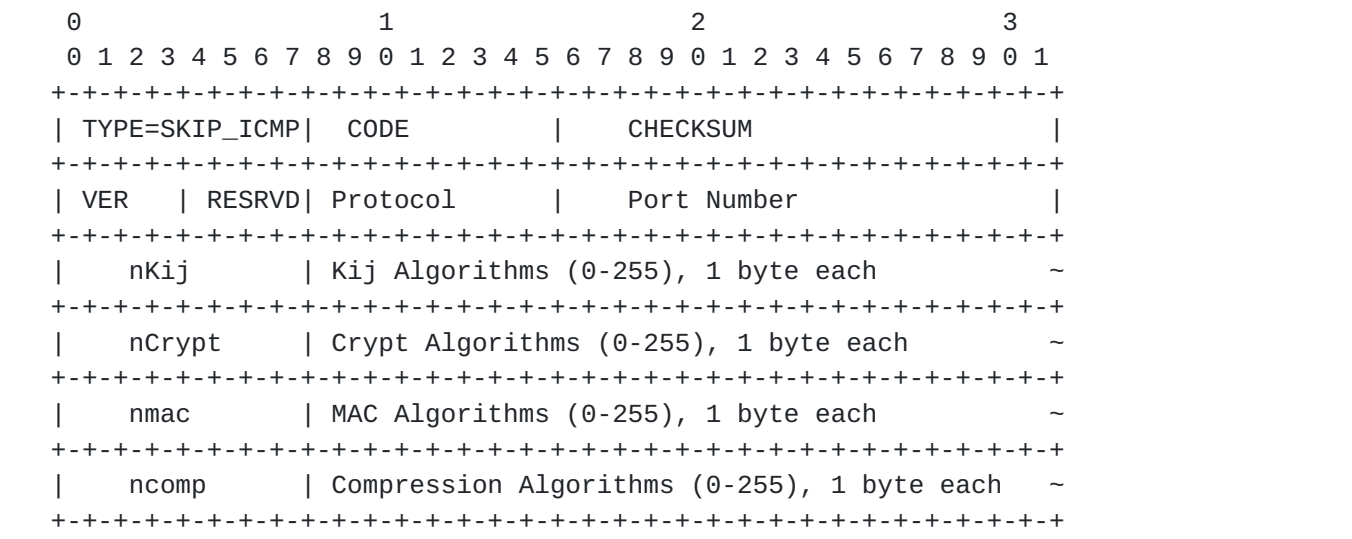
For instance, host A attempts to talk to host B with an encryption algorithm. Host B, however, does not support this algorithm. Host B will send an ICMP message indicating it does not support this algorithm and include the algorithms it does support.

RFC 1825 defines Keyed MD5 as mandatory to implement for Authentication. The SKIP Algorithm Discovery ICMP message MUST be authenticated using SKIP [1], AH [3] and Keyed-MD5 [5], which MUST be supported by all SKIP nodes that support this ICMP protocol.

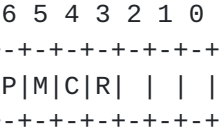
If a node (or communications end point) receives a SKIP packet which specifies algorithms it does not support (or prefer), it SHOULD send an authenticated ICMP message indicating this failure and specifying which algorithms it supports. The ICMP Packet MUST be encapsulated using SKIP and AH with keyed MD5 used as the authentication algorithm. Any received algorithm discovery ICMP message that is not authenticated MUST be ignored and SHOULD be recorded in the system log or audit log.

The ICMP message SHOULD always specify the complete set of Kij, Crypt, MAC and compression algorithms the host supports.

The SKIP Algorithm discovery ICMP message:



CODE should be interpreted as a bit field in the following way:



I is set if the Kij algorithm in the SKIP packet is unsupported.
P is set if the Crypt algorithm in the SKIP packet is unsupported.
M is set if the MAC algorithm in the SKIP packet is unsupported.
C is set if the compression algorithm in the SKIP packet is unsupported.
R is set if replay protection is required but was not used in the SKIP packet. In case a replay protection mechanism is defined, this bit MAY be used to request replay protection.

bits 0-2 are reserved and MUST be set to 0 by the sender and ignored by the receiver.

The ICMP type field SKIP_ICMP is specified later in this document.

The first field "VER" specifies the version of the ICMP message. The Version of the protocol described here is 1.

RESRVD specifies a reserved field. This field MUST be set to zero (0) by the sender and ignored by the receiver.

The next two fields "Protocol" and "Port Number", indicate if this algorithm discovery is to be applied only for a particular protocol/port # pair. This allows different communication end-points on an IP node to use different algorithms. An example of the protocol field could be the TCP protocol, followed by the port # which would identify a TCP end-point. If the Protocol field is non-zero, then the algorithm discovery packet MUST be applied ONLY for the specified communications end-point, as identified by the (Protocol, Port Number) fields.

If the algorithms are to be used on a per Master Key-ID, rather than a per communications end-point basis, then the "Protocol" field MUST be zero. If the "Protocol" field is zero, the Port Number field MUST be ignored. In this case, the algorithms SHOULD be used on a per Master Key-ID basis, where the Master Key-ID is the Source Master Key-ID in the SKIP_ICMP SKIP header. If the source Master Key-ID is absent from the SKIP header, then the algorithms SHOULD be used on a per node basis, using the source IP address of SKIP_ICMP message as the node identifier.

The nKij, ncrypt, nmac and ncomp fields should be filled in with the number of Kij, Crypt, MAC and Compression algorithms the system supports, respectively. If the system does not support a particular class of algorithms, the field should be set to 0. For example, if a system does not support compression, it would set ncomp to 0.

The Kij, Crypt, MAC and Compression algorithms fields should be filled in sequentially with the one byte identifiers for each of the algorithms that the system supports. The algorithms should be an ordered list with the most desirable algorithms first and the least desirable last.

For example, if the system supports 5 Kij algorithms, nKij would be set to 5 and the Kij Algorithms field would be 5 bytes long (one byte for each algorithm supported).

A host may elicit a SKIP_ICMP message by sending a SKIP packet to the remote host with Kij Alg set to zero.

2. Assigned Numbers

2.1 SKIP ICMP message (SKIP_ICMP)

The SKIP algorithm discovery ICMP message has been assigned the type 39 (SKIP_ICMP) by the Internet Assigned Numbers Authority (IANA).

3. Security Considerations

Security issues are the primary topic of this memo.

Unauthenticated SKIP Algorithm Discovery messages or messages which fail authentication MUST be discarded.

Acknowledgements

We would like to thank all of the people who helped make this draft possible.

Martin Patterson and Joseph Reveane for their help in the design of this protocol.

Germano Caronni for his help in reviewing this protocol.

Ran Atkinson for suggesting this protocol be independent of the primary SKIP document.

Bill Danielson, Marc Dye, Colin Plumb, Rich Skrenta and Ben Stoltz for reviewing this draft and providing constructive suggestions.

References

[1] Aziz, A., Markson, T., Prafullchandra, H., "Simple Key Management for Internet Protocols", (I-D draft-ietf-ipsec-skip-06.txt), Work In Progress

[2] Atkinson, R., "Security Architecture for the Internet Protocol", RFC 1825, August 1995

[3] Atkinson, R., "IP Authentication Header", RFC 1826, August 1995

[4] Rivest, R., "The MD5 Message Digest Algorithm", RFC 1321, April 1992

[5] Metzger, P., Simpson, W., "IP Authentication using Keyed MD5", RFC 1828, August 1995

Author's Address(es)

Ashar Aziz
Sun Microsystems, Inc.
M/S PAL1-550
2550 Garcia Avenue
Mountain View, CA 94043

Email: ashar.aziz@eng.sun.com
Alternate email address: ashar@incog.com

Tom Markson
Sun Microsystems, Inc.
M/S PAL1-550
2550 Garcia Avenue
Mountain View, CA 94043

Email: markson@incog.com
Alternate email address: markson@eng.sun.com

Hemma Prafullchandra
Sun Microsystems, Inc.
M/S PAL1-550
2550 Garcia Avenue
Mountain View, CA 94043

Email: hemma@eng.sun.com
Alternate email address: hemma@incog.com