

The Internet IP Security Domain of Interpretation for ISAKMP
<[draft-ietf-ipsec-ipsec-doi-02.txt](#)>

Status of this Memo

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

Internet Drafts are draft documents 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](#) (Australia), [ds.internic.net](#) (US East Coast), or [ftp.isi.edu](#) (US West Coast).

Distribution of this memo is unlimited. This draft will expire six months from date of issue.

[1.](#) Abstract

The Internet Security Association and Key Management Protocol (ISAKMP) defines a framework for security association management and cryptographic key establishment for the Internet. This framework consists of defined exchanges and processing guidelines that occur within a given Domain of Interpretation (DOI). This document details the Internet IP Security DOI, which is defined to cover the IP security protocols that use ISAKMP to negotiate their security associations.

[2.](#) Introduction

Within ISAKMP, a Domain of Interpretation is used to group related protocols using ISAKMP to negotiate security associations. Security protocols sharing a DOI choose security protocol and cryptographic transforms from a common namespace and share key exchange protocol

INTERNET DRAFT

IPSEC DOI

February 28, 1997

identifiers. They also share a common interpretation of DOI-specific payload data content, including the Security Association and Identification payloads.

Overall, ISAKMP places the following requirements on a DOI definition:

- o define the naming scheme for DOI-specific protocol identifiers
- o define the interpretation for the Situation field
- o define the set of applicable security policies
- o define the syntax for DOI-specific SA Attributes (phase II)
- o define the syntax for DOI-specific payload contents
- o define additional mappings or Key Exchange types, if needed

The remainder of this document details the instantiation of these requirements for using the IP Security (IPSEC) protocols to provide data origin authentication and/or data confidentiality for IP packets sent between cooperating host systems and/or firewalls.

[3. Terms and Definitions](#)

[3.1 Requirements Terminology](#)

In this document, the words that are used to define the significance of each particular requirement are usually capitalised. These words are:

– MUST

This word or the adjective "REQUIRED" means that the item is an absolute requirement of the specification.

– SHOULD

This word or the adjective "RECOMMENDED" means that there might exist valid reasons in particular circumstances to ignore this item, but the full implications should be understood and the case carefully weighed before taking a different course.

– MAY

This word or the adjective "OPTIONAL" means that this item is truly optional. One vendor might choose to include the item

because a particular marketplace requires it or because it enhances the product, for example; another vendor may omit the same item.

INTERNET DRAFT

IPSEC DOI

February 28, 1997

[4.1](#) IPSEC Naming Scheme

Within ISAKMP, all DOI's must be registered with the IANA in the ``Assigned Numbers'' RFC [STD-2]. The IANA Assigned Number for the Internet IP Security DOI is one (1). Within the IPSEC DOI, all well-known identifiers MUST be registered with the IANA under the Internet IP Security DOI. Unless otherwise noted, all tables within this document refer to IANA Assigned Numbers for the IPSEC DOI.

All multi-octet binary values are stored in network byte order.

[4.2](#) IPSEC Situation Definition

Within ISAKMP, the Situation provides information that can be used by the responder to make a policy determination about how to process the incoming Security Association request. For the IPSEC DOI, the Situation field is a four (4) octet bitmask with the following values.

Situation	Value
-----	-----
SIT_IDENTITY_ONLY	0x01
SIT_SECRECY	0x02
SIT_INTEGRITY	0x04

All other values are reserved to IANA.

[4.2.1](#) SIT_IDENTITY_ONLY

The SIT_IDENTITY_ONLY type specifies that the security association will be identified by source identity information present in an associated Identification Payload. See [Section 4.6.2](#) for a complete description of the various Identification types. All IPSEC DOI implementations MUST support SIT_IDENTITY_ONLY by including an Identification Payload in at least one of the phase I Oakley exchanges ([\[IO\]](#), Section 5) and MUST abort any association setup that

does not include an Identification Payload.

If an initiator supports neither SIT_SECRECY nor SIT_INTEGRITY, the situation consists only of the 4 octet situation bitmap and does not include the Labeled Domain Identifier field (Figure 1, [Section 4.6.1](#)) or any subsequent label information. Conversely, if the initiator supports either SIT_SECRECY or SIT_INTEGRITY, the Labeled Domain Identifier MUST be included in the situation payload.

[4.2.2](#) SIT_SECRECY

The SIT_SECRECY type specifies that the security association is being

Piper

Expires in 6 months

[Page 3]

INTERNET DRAFT

IPSEC DOI

February 28, 1997

negotiated in an environment that requires labeled secrecy. If SIT_SECRECY is present in the Situation bitmap, the Situation field will be followed by variable-length data that includes a sensitivity level and compartment bitmask. See [Section 4.6.1](#) for a complete description of the Security Association Payload format.

If an initiator does not support SIT_SECRECY, SIT_SECRECY MUST NOT be set in the Situation bitmap and no secrecy level or category bitmaps shall be included.

If a responder does not support SIT_SECRECY, a SITUATION-NOT-SUPPORTED Notification Payload SHOULD be returned and the security association setup MUST be aborted.

[4.2.3](#) SIT_INTEGRITY

The SIT_INTEGRITY type specifies that the security association is being negotiated in an environment that requires labeled integrity. If SIT_INTEGRITY is present in the Situation bitmap, the Situation field will be followed by variable-length data that includes an integrity level and compartment bitmask. If SIT_SECRECY is also in use for the association, the integrity information immediately follows the variable-length secrecy level and categories. See [section 4.6.1](#) for a complete description of the Security Association Payload format.

If an initiator does not support SIT_INTEGRITY, SIT_INTEGRITY MUST NOT be set in the Situation bitmap and no integrity level or category bitmaps shall be included.

If a responder does not support SIT_INTEGRITY, a SITUATION-NOT-SUPPORTED Notification Payload SHOULD be returned and the security association setup MUST be aborted.

[4.3](#) IPSEC Security Policy Requirement

The IPSEC DOI does not impose specific security policy requirements on any implementation. Host system policy issues are outside of the scope of this document.

However, the following sections touch on some of the issues that must be considered when designing an IPSEC DOI host implementation. This section should be considered only informational in nature.

[4.3.1](#) Key Management Issues

It is expected that many systems choosing to implement ISAKMP will strive to provide a protected domain of execution for a combined

Piper

Expires in 6 months

[Page 4]

INTERNET DRAFT

IPSEC DOI

February 28, 1997

ISAKMP/Oakley key management daemon. On protected-mode multiuser operating systems, this key management daemon will likely exist as a separate privileged process.

In such an environment, a formalized API to introduce keying material into the TCP/IP kernel may be desirable. The PF_KEY API [[PFKEY](#)] is an example of one such API that provides an abstracted key management interface.

[4.3.2](#) Static Keying Issues

Host systems that implement static keys, either for use directly by IPSEC, or for authentication purposes (see [[IO](#)] [Section 5.3](#)), should take steps to protect the static keying material when it is not residing in a protected memory domain or actively in use by the TCP/IP kernel.

For example, on a laptop, one might choose to store the static keys in a configuration store that is, itself, encrypted under a private password.

Depending on the operating system and utility software installed, it

may not be possible to protect the static keys once they've been loaded into the TCP/IP kernel, however they should not be trivially recoverable on initial system startup without having to satisfy some additional form of authentication.

[4.3.3](#) Host Policy Issues

It is not realistic to assume that the transition to IPSEC will occur overnight. Host systems must be prepared to implement flexible policy lists that describe which systems they desire to speak securely with and which systems they require speak securely to them. Some notion of proxy firewall addresses may also be required.

A minimal approach is probably a static list of IP addresses, network masks, and a security required flag or flags.

A more flexible implementation might consist of a list of wildcard DNS names (e.g. '*.foo.bar'), an in/out bitmask, and an optional firewall address. The wildcard DNS name would be used to match incoming or outgoing IP addresses, the in/out bitmask would be used to determine whether or not security was to be applied and in which direction, and the optional firewall address would be used to indicate whether or not tunnel mode would be needed to talk to the target system through an intermediate firewall.

[4.3.4](#) Certificate Management

Piper

Expires in 6 months

[Page 5]

INTERNET DRAFT

IPSEC DOI

February 28, 1997

Host systems implementing a certificate-based authentication scheme will need a mechanism for obtaining and managing a database of certificates.

Secure DNS is to be one certificate distribution mechanism, however the pervasive availability of secure DNS zones, in the short term, is doubtful for many reasons. What's far more likely is that hosts will need an ability to import certificates that they acquire through secure, out-of-band mechanisms, as well as an ability to export their own certificates for use by other systems.

However, manual certificate management should not be done so as to preclude the ability to introduce dynamic certificate discovery mechanisms and/or protocols as they become available.

[4.4](#) IPSEC Assigned Numbers

The following sections list the Assigned Numbers for the IPSEC DOI Security Protocol Identifiers, Transform Identifiers, and Security Association Attribute Types.

[4.4.1](#) IPSEC Security Protocol Identifiers

The ISAKMP proposal syntax was specifically designed to allow for the simultaneous negotiation of multiple security protocol suites within a single negotiation. As a result, the protocol suites listed below form the set of protocols that can be negotiated at the same time. It is a host policy decision as to what protocol suites might be negotiated together.

The following table lists the values for the Security Protocol Identifiers referenced in an ISAKMP Proposal Payload for the IPSEC DOI.

Protocol ID -----	Value -----
RESERVED	0
PROTO_ISAKMP	1
PROTO_IPSEC_AH	2
PROTO_IPSEC_ESP	3

The size of this field is one octet. The values 4-248 are reserved to IANA. Values 249-255 are reserved for private use.

4.4.1.1 PROTO_ISAKMP

The PROTO_ISAKMP type specifies message protection required during Phase I of the ISAKMP protocol. The specific protection mechanism

used for the IPSEC DOI is described in [\[10\]](#). All implementations within the IPSEC DOI MUST support PROTO_ISAKMP.

NB: ISAKMP reserves the value one (1) across all DOI definitions.

[4.4.1.2](#) PROTO_IPSEC_AH

The PROTO_IPSEC_AH type specifies IP packet data origin

authentication. The default AH transform includes data origin authentication and replay prevention. For export control considerations, confidentiality MUST NOT be provided by any PROTO_IPSEC_AH transform.

[4.4.1.3](#) PROTO_IPSEC_ESP

The PROTO_IPSEC_ESP type specifies IP packet confidentiality. Data origin authentication, if required, must be provided as part of the ESP transform. The default ESP transform includes data origin authentication, confidentiality, and replay prevention.

[4.4.2](#) IPSEC ISAKMP Transform Values

As part of an ISAKMP Phase I negotiation, the initiator's choice of Key Exchange offerings is made using some host system policy description. The actual selection of Key Exchange mechanism is made using the standard ISAKMP Proposal Payload. The following table lists the defined ISAKMP Phase I Transform Identifiers for the Proposal Payload for the IPSEC DOI.

Transform	Value
-----	-----
RESERVED	0
KEY_OAKLEY	1
KEY_MANUAL	2
KEY_KDC	3

The size of this field is one octet. The values 4-248 are reserved to IANA. Values 249-255 are reserved for private use.

[4.4.2.1](#) KEY_OAKLEY

The KEY_OAKLEY type specifies the hybrid ISAKMP/Oakley Diffie-Hellman key exchange as defined in the [[10](#)] document. All implementations within the IPSEC DOI MUST support KEY_OAKLEY.

[4.4.2.2](#) KEY_MANUAL

The KEY_MANUAL type specifies that a shared secret key mechanism is

to be used in lieu of a dynamic key mechanism. Specific details of a

static key establishment protocol will be described in a future document.

[4.4.2.3](#) KEY_KDC

The KEY_KDC type specifies that a secret-key based Key Distribution Center will be used to provide dynamic key exchange through a Kerberos-like ticket protocol. Specific details of a KDC-based key establishment protocol will be described in a future document.

[4.4.3](#) IPSEC AH Transform Values

The Authentication Header Protocol (AH) defines one mandatory and several optional transforms used to provide data origin authentication. The following table lists the defined AH Transform Identifiers for the ISAKMP Proposal Payload for the IPSEC DOI.

Transform ID	Value
-----	-----
RESERVED	0
AH_MD5_KPDK	1
AH_MD5	2
AH_SHA	3

The size of this field is one octet. The values 4-248 are reserved to IANA. Values 249-255 are reserved for private use.

[4.4.3.1](#) AH_MD5_KPDK

The AH_MD5_KPDK type specifies the AH transform (Key/Pad/Data/Key) described in [RFC-1826](#) and [RFC-1828](#). This mode MAY be used for compatibility with existing implementations. Implementations are not required to support this mode.

[4.4.3.2](#) AH_MD5

The AH_MD5 type specifies a generic AH transform using MD5. The actual protection suite is determined in concert with an associated SA attribute list. A generic MD5 transform is currently undefined.

All implementations within the IPSEC DOI MUST support AH_MD5 along with the HMAC and REPLAY attributes. This suite is defined as the HMAC-MD5 transform described in [RFC-2085](#).

[4.4.3.3](#) AH_SHA

The AH_SHA type specifies a generic AH transform using SHA-1. The

actual protection suite is determined in concert with an associated SA attribute list. A generic SHA transform is currently undefined.

All implementations within the IPSEC DOI are strongly encouraged to support AH_SHA along with the HMAC and REPLAY attributes. This suite is defined as the HMAC-SHA transform described in [[HMACSHA](#)].

[4.4.4](#) IPSEC ESP Transform Identifiers

The Encapsulating Security Protocol (ESP) defines one mandatory and several optional transforms used to provide data confidentiality. The following table lists the defined ESP Transform Identifiers for the ISAKMP Proposal Payload for the IPSEC DOI.

Transform ID	Value
-----	-----
RESERVED	0
ESP_DES_CBC	1
ESP_DES	2
ESP_3DES	3
ESP_RC5	4

The size of this field is one octet. The values 5-248 are reserved to IANA. Values 249-255 are reserved for private use.

[4.4.4.1](#) ESP_DES_CBC

The ESP_DES_CBC type specifies the DES-CBC transform defined in [RFC-1827](#) and [RFC-1829](#). This mode MAY be used for compatibility with existing implementations. Implementations are not required to support this mode.

[4.4.4.2](#) ESP_DES

The ESP_DES type specifies a generic DES transform using DES-CBC. The actual protection suite is determined in concert with an associated SA attribute list. A generic transform is currently undefined.

All implementations within the IPSEC DOI MUST support ESP_DES along with the HMAC and REPLAY attributes. This suite is defined as the [[Hughes](#)] transform.

[4.4.4.3](#) ESP_3DES

The ESP_3DES type specifies a generic triple-DES transform. The

actual protection suite is determined in concert with an associated SA attribute list. The generic transform is currently undefined.

All implementations within the IPSEC are strongly encouraged to support ESP_3DES along with the HMAC and REPLAY attributes. This suite is defined as the [Naganand] transform.

4.4.4.4 ESP_RC5

The ESP_RC5 type specifies the RC5 transform defined in [RC5].

4.5 IPSEC Security Association Attributes

The following SA attribute definitions are used in phase II of an ISAKMP/Oakley negotiation. Attribute types can be either Basic (B) or Variable-Length (V). Encoding of these attributes is defined in the base ISAKMP specification.

Attribute Classes

class	value	type
Auth Key Life Type	1	B
Auth Key Life Duration	2	B/V
Enc Key Life Type	3	B
Enc Key Life Duration	4	B/V
SA Life Type	5	B
SA Life Duration	6	B/V
Replay Protection	7	B
Group Description	8	B
CA Distinguished Name	9	B
Encapsulation Mode	10	B
HMAC Algorithm	11	B

Class Values

- Auth Key Life Type
- Auth Key Duration

Specifies the time-to-live for the authentication key(s) used in the corresponding AH HMAC transform.

Enc Key Life Type
Enc Key Duration

Specifies the time-to-live for the encryption key(s)
using in the corresponding ESP transform.

SA Life Type
SA Duration

Piper

Expires in 6 months

[Page 10]

INTERNET DRAFT

IPSEC DOI

February 28, 1997

Specifies the time-to-live for the overall security
association. When the SA expires, all keys negotiated
under the association (AH or ESP) must be renegotiated
regardless of the time-to-live remaining for the keys.

RESERVED	0
seconds	1
kilobytes	2

Values 3-61439 are reserved to IANA. Values 61440-65535
are for experimental use. For a given "Life Type," the
value of the "Life Duration" attribute defines the actual
length of the component lifetime -- either a number of
seconds, or a number of Kbytes that can be protected.

If unspecified, the default value shall be assumed to be
28800 seconds (8 hours) for Auth, Enc, and SA lifetime.

Replay Protection

RESERVED	0
required	1
disallowed	2

Values 3-61439 are reserved to IANA. Values 61440-65535
are for private use among mutually consenting parties.

There is no default value for Replay Protection, as it
must be specified to correctly identify the applicable
transform.

Group Description

RESERVED	0
----------	---

default group 1

Values 2-61439 are reserved to IANA. Values 61440-65535 are for private use among mutually consenting parties.

If unspecified, the default value shall be assumed to be the default Oakley group ([\[IO\]](#), Section 5.5.1).

CA Distinguished Name

RESERVED 0
DNS Security 1

Values 2-61439 are reserved to IANA. Values 61440-65535 are for private use among mutually consenting parties.

If unspecified, the default value shall be assumed to be

Piper Expires in 6 months [Page 11]

INTERNET DRAFT

IPSEC DOI

February 28, 1997

DNS Security ([Section 4.8](#)).

Encapsulation Mode

RESERVED 0
Tunnel 1
Transport 2

Values 3-61439 are reserved to IANA. Values 61440-65535 are for private use among mutually consenting parties.

If unspecified, the default value shall be assumed to be unspecified (host-dependent).

HMAC Algorithm

RESERVED 0
MD5 1
SHA-1 2

Values 3-61439 are reserved to IANA. Values 61440-65535 are for private use among mutually consenting parties.

There is no default value for HMAC Algorithm, as it must be specified to correctly identify the applicable transform.

[4.5.1](#) Required Attribute Support

To ensure basic interoperability, all implementations MUST support all of the following attributes:

- Auth Key Life Type
- Auth Key Duration
- Enc Key Life Type
- Enc Key Duration
- SA Life Type
- SA Duration
- Replay Protection
- HMAC Algorithm (MD5 required, SHA-1 optional)

[4.5.2](#) Attribute List Parsing Requirement

To allow for flexible semantics, the IPSEC DOI requires that a conforming ISAKMP implementation MUST correctly parse an attribute list that contains multiple instances of the same attribute class, so long as the different attribute entries do not conflict with one another.

To see why this is important, the following example shows the binary

Piper Expires in 6 months [Page 12]

INTERNET DRAFT IPSEC DOI February 28, 1997

encoding of a four entry attribute list that specifies an Encryption Key Lifetime of either 100MB or 24 hours. (See Section 3.3 of [\[ISAKMP\]](#) for a complete description of the attribute encoding format.)

Attribute #1:

0x80030001 (AF = 1, type = Enc Key Life Type, value = seconds)

Attribute #2:

0x00040004 (AF = 0, type = Enc Key Duration, length = 4 bytes)
0x00015180 (value = 0x15180 = 86400 seconds = 24 hours)

Attribute #3:

0x80030002 (AF = 1, type = Enc Key Life Type, value = KB)

Attribute #4:

0x00040004 (AF = 0, type = Enc Key Duration, length = 4 bytes)
0x000186A0 (value = 0x186A0 = 100000KB = 100MB)

If conflicting attributes are detected, an ATTRIBUTES-NOT-SUPPORTED Notification Payload SHOULD be returned and the security association setup MUST be aborted.

4.6 IPSEC Payload Content

The following sections describe those ISAKMP payloads whose data representations are dependent on the applicable DOI.

4.6.1 Security Association Payload

The following diagram illustrates the content of the Security Association Payload for the IPSEC DOI. See [Section 4.2](#) for a description of the Situation bitmap.

[illegible]

```

! Secrecy Cat. Length (in bits) !                               RESERVED                               !
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
~                               Secrecy Category Bitmap                               ~
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
! Integrity Length (in octets) !                               RESERVED                               !
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
~                               Integrity Level                               ~
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
! Integ. Cat. Length (in bits) !                               RESERVED                               !
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
~                               Integrity Category Bitmap                               ~

```


to designate integrity categories (compartments) that are required.

[4.6.1.1](#) Labeled Domain Identifier Values

The following table lists the assigned values for the Labeled Domain Identifier field contained in the Situation field of the Security Association Payload.

Domain -----	Value -----
RESERVED	0

The values 1-0x7fffffff are reserved to IANA. Values 0x80000000-0xffffffff are reserved for private use.

[4.6.2](#) Identification Payload Content

The Identification Payload is used to identify the initiator of the Security Association. The identity of the initiator SHOULD be used by the responder to determine the correct host system security policy requirement for the association. For example, a host might choose to require data origin authentication without confidentiality (AH) from a certain set of IP addresses and full authentication with confidentiality (Hughes) from another range of IP addresses. The Identification Payload provides information that can be used by the responder to make this decision.

The following diagram illustrates the content of the Identification Payload.

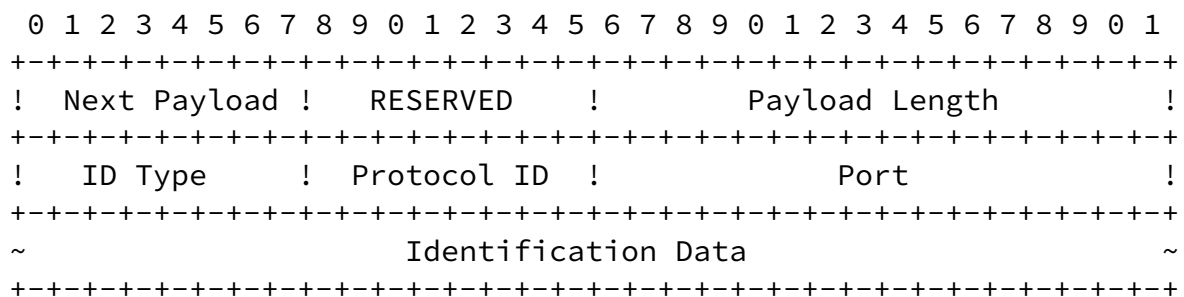


Figure 2: Identification Payload Format

The Identification Payload field is defined as follows:

- o Next Payload (2 octets) - Identifier for the payload type of the next payload in the message. If the current payload is the last in the message, this field will be zero (0).
- o RESERVED (1 octet) - Unused, must be zero (0).
- o Payload Length (2 octets) - Length, in octets, of the identification data, including the generic header.
- o Identification Type (1 octet) - Value describing the identity information found in the Identification Data field.
- o Protocol ID (1 octet) - Value specifying an associated IP protocol ID (e.g. UDP/TCP). A value of zero means that the Protocol ID field should be ignored.
- o Port (2 octets) - Value specifying an associated port. A value of zero means that the Port field should be ignored.
- o RESERVED (1 octet) - Unused, must be zero (0).

[4.6.2.1](#) Identification Type Values

The following table lists the assigned values for the Identification Type field found in the Identification Payload.

ID Type	Value
-----	-----
RESERVED	0
ID_IPV4_ADDR	1
ID_FQDN	2
ID_USER_FQDN	3
ID_IPV4_ADDR_SUBNET	4
ID_IPV6_ADDR	5
ID_IPV6_ADDR_SUBNET	6
ID_IPV4_ADDR_RANGE	7
ID_IPV6_ADDR_RANGE	8

The size of this field is one octet. The values 9-248 are reserved to IANA. Values 249-255 are reserved for private use.

[4.6.2.2](#) ID_IPV4_ADDR

The ID_IPV4_ADDR type specifies a single four (4) octet IPv4 address.

[4.6.2.3](#) ID_FQDN

The ID_FQDN type specifies a fully-qualified domain name string. An

example of a ID_FQDN is, "foo.bar.com".

[4.6.2.4](#) ID_USER_FQDN

The ID_USER_FQDN type specifies a fully-qualified username string, An example of a ID_USER_FQDN is, "piper@foo.bar.com".

[4.6.2.5](#) ID_IPV4_ADDR_SUBNET

The ID_IPV4_ADDR_SUBNET type specifies a range of IPv4 addresses, represented by two four (4) octet values. The first value is an IPv4 address. The second is an IPv4 network mask. Note that ones (1s) in the network mask indicate that the corresponding bit in the address is fixed, while zeros (0s) indicate a "wildcard" bit.

[4.6.2.6](#) ID_IPV6_ADDR

The ID_IPV6_ADDR type specifies a single sixteen (16) octet IPv6 address.

[4.6.2.7](#) ID_IPV6_ADDR_SUBNET

The ID_IPV6_ADDR_SUBNET type specifies a range of IPv6 addresses, represented by two sixteen (16) octet values. The first value is an IPv6 address. The second is an IPv6 network mask. Note that ones (1s) in the network mask indicate that the corresponding bit in the address is fixed, while zeros (0s) indicate a "wildcard" bit.

[4.6.2.8](#) ID_IPV4_ADDR_RANGE

The ID_IPV4_ADDR_RANGE type specifies a range of IPv4 addresses, represented by two four (4) octet values. The first value is the beginning IPv4 address (inclusive) and the second value is the ending IPv4 address (inclusive). All addresses falling between the two specified addresses are considered to be within the list.

[4.6.2.9](#) ID_IPV6_ADDR_RANGE

The ID_IPV6_ADDR_RANGE type specifies a range of IPv6 addresses, represented by two sixteen (16) octet values. The first value is the

beginning IPv6 address (inclusive) and the second value is the ending IPv6 address (inclusive). All addresses falling between the two specified addresses are considered to be within the list.

[4.7](#) IPSEC Security Parameter Index (SPI) Encoding

ISAKMP defines the SPI field as eight octets in length, however the IPSEC transforms use only four octets.

Piper

Expires in 6 months

[Page 17]

INTERNET DRAFT

IPSEC DOI

February 28, 1997

All implementation MUST use the following mapping for the ISAKMP SPI field in the IPSEC DOI.

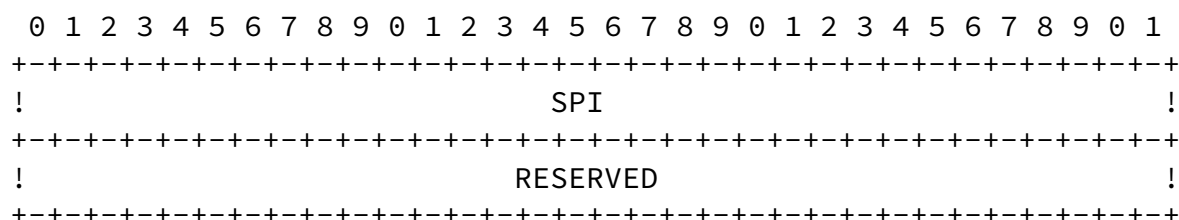


Figure 3: ISAKMP SPI Encoding

The ISAKMP SPI field is defined as follows:

- o SPI - Security Parameter Index (4 octets) - contains the SPI value which identifies the security association.
- o RESERVED (4 octets) - Unused, must be zero (0).

[4.8](#) IPSEC Certificate Authorities

The ISAKMP Certificate Request Payload allows either side of an ISAKMP negotiation to request its peer to provide a certificate chain needed to authenticate itself. The Certificate Request Payload includes a list of acceptable Certificate Types and Certificate Authorities. Appropriate certificate chains are then returned in a Certificate Payload response.

The IPSEC DOI defines the following Certificate Authorities for the processing of Certificate Request Payloads. See [Section 4.5](#) for details on the specific data attribute type values for these CAs.

Certificate Authority

DNS Security

4.8.1 DNS Security

This CA type represents the combination of KEY and SIG records, as defined in [RFC-2065](#), that can be used for authentication. The particular encoding required to formulate the Certificate Payload (response) is TBD.

[4.9](#) IPSEC Key Exchange Requirements

The IPSEC DOI introduces no additional Key Exchange types.

[5](#). Security Considerations

Piper Expires in 6 months [Page 18]

INTERNET DRAFT IPSEC DOI February 28, 1997

This entire draft pertains to a hybrid protocol, combining Oakley ([[OAKLEY](#)]) with ISAKMP ([[ISAKMP](#)]), to negotiate and derive keying material for security associations in a secure and authenticated manner. Specific discussion of the various security protocols and transforms identified in this document can be found in the associated base documents.

Acknowledgements

This document is derived, in part, from previous works by Douglas Maughan, Mark Schertler, Mark Schneider, Jeff Turner, Dan Harkins, and Dave Carrel.

References

[RFC-1825] Atkinson, R., "Security Architecture for the Internet Protocol," [RFC-1825](#), August 1995.

[RFC-1826] Atkinson, R., "IP Authentication Header," [RFC-1826](#), August 1995.

[RFC-1827] Atkinson, R., "IP Encapsulating Security Payload (ESP)," [RFC-1827](#), August 1995.

[RFC-1828] Metzger, P., Simpson, W., "IP Authentication using Keyed MD5," [RFC-1828](#), August 1995.

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

[RFC-2065] Eastlake 3rd, D., Kaufman, C., "Domain Name System Security Extensions," [RFC-2065](#), January 1997.

[RFC-2085] Oehler, M., Glenn, R., "HMAC-MD5 IP Authentication with Replay Prevention," [RFC-2085](#), February 1997.

[HMACSHA] Chang, S., Glenn, R., "HMAC-SHA IP Authentication with Replay Prevention," [draft-ietf-ipsec-ah-hmac-sha-03.txt](#).

[Hughes] Hughes, J., Editor, "Combined DES-CBC, HMAC and Replay Prevention Transform," [draft-ietf-ipsec-esp-des-md5-03.txt](#).

[IO] Carrel, D., Harkins, D., "The Resolution of ISAKMP with Oakley," [draft-ietf-ipsec-isakmp-oakley-03.txt](#).

[ISAKMP] Maughan, D., Schertler, M., Schneider, M., and Turner, J., "Internet Security Association and Key Management Protocol (ISAKMP)," [draft-ietf-ipsec-isakmp-07](#).{ps,txt}.

Piper

Expires in 6 months

[Page 19]

INTERNET DRAFT

IPSEC DOI

February 28, 1997

[Naganand] Daraswamy, N., "Combined 3DES-CBC, HMAC and Replay Detection Security Transform," [draft-ietf-ipsec-esp-3des-md5-00.txt](#).

[OAKLEY] H. K. Orman, "The OAKLEY Key Determination Protocol," [draft-ietf-ipsec-oakley-01.txt](#).

[PFKEY] McDonald, D. L., Metz, C. W., Phan, B. G., "PF_KEY Key Management API, Version 2", [draft-mcdonald-pf-key-v2-00.txt](#), work in progress.

[RC5] Howard, B., Baldwin, R., "The ESP RC5-CBC Transform," [draft-baldwin-esp-rc5-00.txt](#).

Author's Address:

Derrell Piper <piper@cisco.com>
cisco Systems
101 Cooper St.
Santa Cruz, California, 95060

United States of America
+1 408 457-5384

Piper

Expires in 6 months

[Page 20]