Internet Draft
draft-ietf-ipsec-ike-monitor-mib-04.txt
April 15, 2003
Expires in six months

Editor: Paul Hoffman
VPN Consortium

Internet Key Exchange (IKE) Monitoring MIB

Status of this Memo

This document is an Internet-Draft and is in full conformance with all provisions of Section 10 of RFC2026.

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 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."

The list of current Internet-Drafts can be accessed at
http://www.ietf.org/ietf/1id-abstracts.txt

The list of Internet-Draft Shadow Directories can be accessed at
http://www.ietf.org/shadow.html.

Table of Contents

[[Needs to be generated in the RFC publication step]]

1. Introduction

This document defines monitoring and status MIBs for use when the (Internet Key Exchange) IKE protocol [IKE] is used to create IPsec security associations (SAs). As such, the MIBs provide the linkage between IKE (phase 1) SAs and the IPsec (phase 2) SAs created by those SAs.

It does not define MIBs that may be used for configuring IPsec implementations or for providing low-level diagnostic or debugging information. It assumes no specific use of IPsec SAs, except that they were created using IKE. Further, it does not provide policy information.

The purpose of the MIBs is to allow system administrators to determine operating conditions and perform system operational level monitoring of the IPsec portion of their network. Statistics are provided as well. Additionally, it may be used as the basis for application specific MIBs for specific uses of IPsec.

Note: SNMPv1 implementations of this MIB may need to adjust limits on SNMP message sizes to accommodate some of the variables (for example, modpPrime) that would cause those messages to exceed 484 octets in length.

2. The SNMP Management Framework

The SNMP Management Framework presently consists of five major components:

- An overall architecture, described in RFC 2571 [RFC2571].
- Mechanisms for describing and naming objects and events for the purpose of management. The first version of this Structure of Management Information (SMI) is called SMIv1 and described in STD 16, RFC 1155 [RFC1155], STD 16, RFC 1212 [RFC1212] and RFC 1215 [RFC1215]. The second version, called SMIv2, is described in STD 58, RFC 2578 [RFC2578], RFC 2579 [RFC2579] and RFC 2580 [RFC2580].
- Message protocols for transferring management information. The first version of the SNMP message protocol is called SNMPv1 and described in STD 15, RFC 1157 [RFC1157]. A second version of the SNMP message protocol, which is not an Internet standards track protocol, is called SNMPv2c and described in RFC 1901 [RFC1901] and RFC 1906 [RFC1906]. The third version of the message protocol is called SNMPv3 and described in RFC 1906 [RFC1906], RFC 2572 [RFC2572] and RFC 2574 [RFC2574].
- Protocol operations for accessing management information. The first set of protocol operations and associated PDU formats is described in STD 15, RFC 1157 [RFC1157]. A second set of protocol operations and associated PDU formats is described in RFC 1905 [RFC1905].
- A set of fundamental applications described in RFC 2573 [RFC2573] and the view-based access control mechanism described in RFC 2575 [RFC2575].

A more detailed introduction to the current SNMP Management Framework can be found in RFC 2570 [RFC2570].

Managed objects are accessed via a virtual information store, termed the Management Information Base or MIB. Objects in the MIB are defined using the mechanisms defined in the SMI.

This memo specifies a MIB module that is compliant to the SMIv2. A MIB conforming to the SMIv1 can be produced through the appropriate translations. The resulting translated MIB must be semantically equivalent, except where objects or events are omitted because no translation is possible (use of Counter64). Some machine readable

information in SMIv2 will be converted into textual descriptions in SMIv1 during the translation process. However, this loss of machine readable information is not considered to change the semantics of the MIB.

2.1 Object Definitions

Managed objects are accessed via a virtual information store, termed the Management Information Base or MIB. Objects in the MIB are defined using the subset of Abstract Syntax Notation One (ASN.1) defined in the SMI. In particular, each object type is named by an OBJECT IDENTIFIER, an administratively assigned name. The object type together with an object instance serves to uniquely identify a specific instantiation of the object. For human convenience, we often use a textual string, termed the descriptor, to refer to the object type.

3. Definitions

3.1 Security Association, Inbound and Outbound

This document uses the same definitions of "security association", "inbound" and "outbound" as [IMMIB].

3.2 Phase 2 Security Association Suite

This MIB uses a concept of a phase 2 security association suite. A phase 2 security association suite is defined as the set of SAs that result from each SA payload in a successful IKE Quick Mode exchange. This entity is called a suite for the remainder of this document to reduce the usage of the ambiguous term "security association".

Phrased another way, a suite is the set of IPsec phase 2 SAs created when negotiated using IKE, and the phase 2 SAs were negotiated as part of the same SA payload.

As such, a suite is a subset of an SA bundle as defined in RFC 2401. In RFC 2401, the SA pairs in the bundle may be negotiated separately and independently.4. IPsec MIB Objects Architecture

The IPsec MIB consists of a number of separate tables.

First, there is an IKE SA table that provides monitoring for phase 1 security associations (SAs). This table is a DOI-specific table that uses the base ISAKMP SA table from the ISAKMP DOI-independent MIB as its base. Specifically, the IKE SA table has a sparse dependent relationship to the ISAKMP SA table.

Secondly, there is an endpoint table. This table is used to store the identities of all endpoints that have been (or may be) involved in IKE SA negotiation with the local entity. This includes any identities that the local entity itself may use. Additionally, this table also has some endpoint specific information.

Additionally, there are a number of tables associated with the phase 2 SAs. One of these tables contains suites. Each row of this table contains common and general objects that are part of the suite.

Another table allows determination of the specific phase 2 SAs that are the components of each suite. This table augments the suite table. This table, called the phase 2 SA table, has a dependent expansion relationship to the suite table. Note that this table should not be confused with the collection of phase 2 SA tables found in [IMMIB]. However, this table indirectly refers to those tables.

Configuration about the phase 1 IKE SAs and the suites is provided as are statistics related to the phase 1 IKE SAs and the suites themselves. Additionally, the MIBs provide a number of entity level aggregate totals for the phase 1 SAs and suites.

A general picture of the relationship of the tables is shown in Figure 1. The individual phase 2 SA tables and the selector table are from [IMMIB] and the phase 1 DOI-independent SA table is from [IDIMIB]. The tables from this MIB require both the IPsec tables and the ISAKMP table.

Both the phase 1 SA and suite tables refer to the endpoint table to subsequently refer to the peers that negotiated the SAs and suites.

Additional tables, such as the Oakley groups, the exchange table and helper tables are not shown in Figure 1.

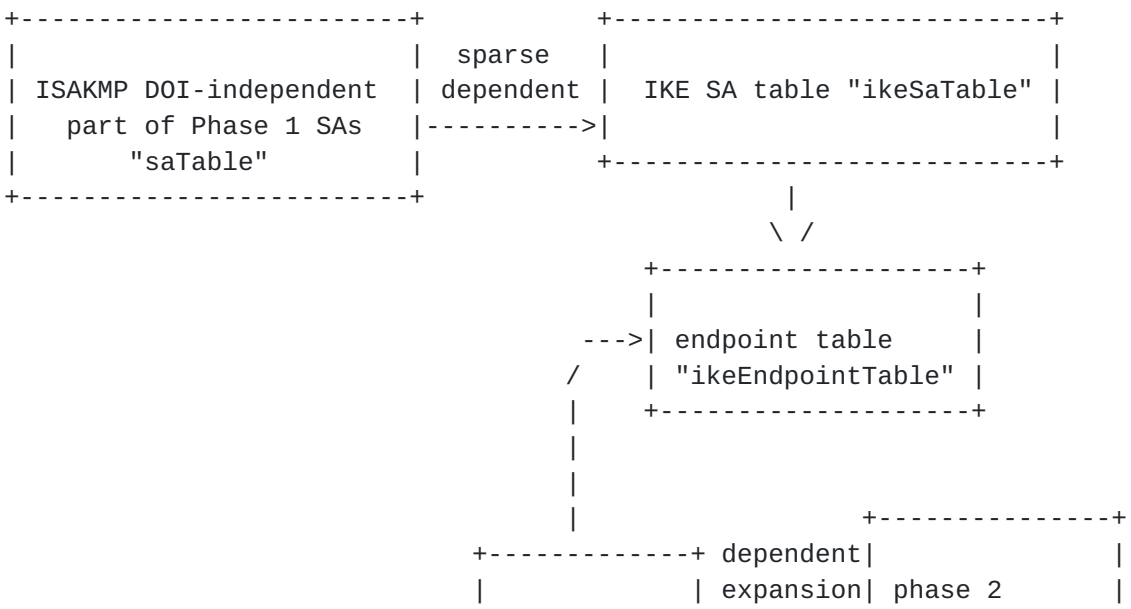

Figure 1. Relationship of MIB Tables

4.1 Endpoint Table

This table is used to allow the endpoints involved in the IKE negotiations to be identified. It provides the ID type and value used by both ends during phase 1 negotiations, as well as certificate information. (See next section.)

Additionally, it indicates if the endpoint is local or remote, and provides basic statistics for the endpoints with respect to the number of IKE SAs and phase 2 SA suites the endpoints have created.

Implementations could also use this as a base table for more detailed per endpoint statistics, such as error counts or traffic counts. However, these are not specified in this MIB.

4.1.1 Peer Certificate Information

The MIB provides certificate information related to the authentication of the peer entity. This information is the ID used in

phase 1, and the certificate's serial number and issuer. It is intended that this information be sufficient to determine the certificate that was used for peer authentication.

No certificate chain information is provided. The reasons for this are that the chain may not be available to the entity and the chain is not necessarily exchanged in phase 1. A more appropriate place for this type of information might be in a PKI MIB; as such, it is beyond the scope of this document.

4.2 IKE Security Association Table

IKE SAs presented in the table contain information about the services provided, their lifetime, endpoint authentication and some aggregate performance statistics.

This table extends the ISAKMP DOI-independent phase 1 SA table, so is indexed by the same indices. It does not use the AUGMENTS capability of SNMP, since all ISAKMP SAs are not necessarily IKE SAs. As stated earlier, it has a sparse dependent relationship to the ISAKMP SA table.

In addition to the information already provided by the DOI-independent phase 1 SA table, the IKE SA table adds to information related to the identities of the two endpoint entities, the security information of the IKE SA, some expiration limits and some additional operating statistics.4.2.1 Phase 1 SA Helper Tables

The MIB provides one helper table to modify the search order for phase 1 SAs. This table uses the endpoints along with an arbitrary value as its index.

The rows of this table contain the endpoint addresses and cookies of the individual SAs that exist between the endpoints. This allows look up of the specific phase 1 SAs from these values.

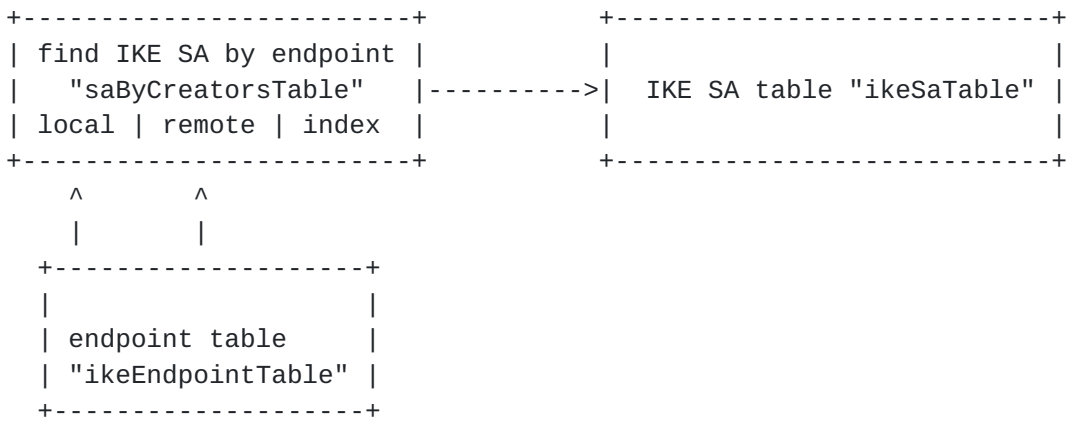

Figure 2. IKE SA Table Helper Table

4.3 Phase 2 Security Association Suite Table

Suites are as defined above (in Section 3). This MIB makes no assumptions about the order or protocol of the individual SAs within the suite.

Individual bi-directional SAs that are negotiated using IKE's quick mode are treated as a suite that uses only a single security protocol.

[ISAKMP] requires that common attributes negotiated within a suite apply to all SAs. Therefore, the suite table provides expiration values and selectors for the suite. In order to get the statistics for the individual SAs, the phase 2 SA table provides the ability to get to the SAs themselves.

The suite table is indexed by an arbitrary integer. This was done to ease implementation, since the number of objects that are required to uniquely identify individual suites is very high. (For a suite with three inbound/outbound SA pairs, there would be 11 indices required.) This also allows the suite table to be independent of the number and order of SAs as used within the suite.

Helper tables may be used to provide a list of suites in the desired order; a number of these are provided for what are expected to be desirable sorting orders.

In order to link the creation of suites (and thereby SAs) to specific endpoints, the suite table also contains references to the endpoints that negotiated the SAs. No direct link is possible since there is no requirement that any phase 1 SA exists after creation of a suite.

Many of the objects of each suite are duplicates of objects found in the SAs' entries in their respective tables. This is done to allow a faster lookup of the SA information as the SAs are being used by IKE. As part of this, some statistical aggregation is done as well.

As stated earlier, the suite table itself does not provide knowledge of which specific SAs make up the suite. This information is obtained from the phase 2 SA table.

4.3.1 Suite Helper Tables

There are three helper tables provided to allow searching of suites in a non-arbitrary order. These tables are ordered by endpoints, by SA selector and phase 2 SA identifiers, respectively.

The first table is indexed in the same way as the IKE SA helper

table, but provides a reference to a specific suite index value. It can be used to look up suites based on a specific set of entity IDs.

SA selectors index the second table, by augmenting the selectors table from IPsec Monitoring MIB. Since duplicates of suites with the same selector is permitted, and is normal during re-keying, the additional index is an arbitrary integer. Each row provides a reference to a specific suite index value.

The third helper table is provided to allow the determination of which suite a particular SA is being used in. This table is indexed by a destination address, a protocol and an SPI (CPI if the protocol is IPcomp). These are the objects that make an SA unique. Each row then provides a reference to a specific suite in which the SA is being used.

Figure 3. Suite Table Helper Tables

4.3.2 Phase 2 SA Table

This table allows the determination of which SAs from the IPsec monitoring MIB are in the SA suites. It is indexed by the suite table's index with an additional integer object added. This effectively causes expansion of the suite table for suites that have more than one SA. As stated earlier, it has a dependent expansion relationship to the suite table, and is shown in Figure 1.

The value of the additional index object is the position of the particular SA in the suite. The value one is used to indicate the outer most SA; that is, the SA whose header appears as the outer most after application of all the SA's headers. (In the case of IPcomp, the header may be missing for specific packets if the packet was not considered compressible; for the purposes of this definition, it is assumed the IPcomp header is always applied.)

The other row elements in this table are the security protocol and the SPIs of the inbound and outbound SAs. This information, along with the addresses of the suite, can be used to form a lookup into the IPsec monitoring MIB's SA table for specific SAs.

4.4 Security Association Bundles

This MIB does not explicitly show SA bundles or any combination of layered SAs that do not meet the suite definition as defined in this document. However, these may be represented in these MIBs by separate protection suites with the appropriate set of selectors.

4.5 Uni-directional Suites

This MIB does not explicitly support suites that are uni-directional. However, this can be supported by the suite to SA table using a value of 0 for the SPI in the particular direction that is not used.

4.6 Oakley Group Tables

These tables are used to allow an entity to describe the Oakley groups that it knows about. Each table contains a row for each of the Oakley groups of a specific type. This table does not contain the well-known groups.

The structure of each table is taken directly from Appendix A of [OAKLEY].

The tables are used to allow both phase 1 SAs and suites to indicate how their source keying material was generated if they did not use one of the well-known groups. Additionally in the case of suites, this method is used if the phase 2 keying material was not derived

from the phase 1 SA's keying material.

4.7 Exchange Table

This table provides the number of IPsec DOI exchanges tried that were used in a phase 1 IKE SA, the number successfully responded to in a phase 1 IKE SA and the total number successfully completed in a phase 1 IKE SA. This table augments the phase 1 security associations table (but again, not using the AUGMENTS clause of SNMP).

4.8 Notify Messages

Notify messages sent from peer to peer are collected as they occur and accumulated in a parse table structure.

A notify message object is defined. This object is used as the index into the table of accumulated notify messages. This helps system administrators determine if there are potential configuration problems or attacks on their network.

4.9 Traps

Traps are provided to let system administrators know about the existence of error conditions occurring in the entity. Errors are associated with the creation and deletion of SAs, and also operational errors that may indicate the presence of attacks on the system.

Traps are not provided when SAs come up or go down, unless they cannot be negotiated or go down due to error conditions.

The causes of SA negotiation failure are indicated by a notify message object.

The transmission of traps may be controlled as well.

4.10 Entity Level Objects

This part of the MIB carries statistics global to the device.

Statistics included are aggregate usage and aggregate errors for both phase 1 SAs and phase 2 suites. The statistics are provided as objects in a tree below these groups.

5. MIB Definitions

```
 IKE-MON-MIB DEFINITIONS ::= BEGIN
```

```
    IMPORTS

    MODULE-IDENTITY, OBJECT-TYPE, Counter32, Counter64,
    Unsigned32, Gauge32, OBJECT-IDENTITY,
    experimental, NOTIFICATION-TYPE
                                    FROM SNMPv2-SMI
    TruthValue                      FROM SNMPv2-TC
    InetAddressType, InetAddress
                                    FROM INET-ADDRESS-MIB
    IpsecRawId, selectorIndex       FROM IPSEC-SA-MON-MIB
    saLocalIpAddressType, saLocalIpAddress, saRemoteIpAddressType,
    saRemoteIpAddress, saInitiatorCookie, saResponderCookie,
    IsakmpCookie, localIpAddressType, localIpAddress, localUdpPort,
    remoteIpAddressType, remoteIpAddress, remoteUdpPort
                                    FROM ISAKMP-DOI-IND-MON-MIB
    IpsecDoiIdentType, IkeAuthMethod, IkeEncryptionAlgorithm,
    IkeGroupDescription, IkePrf, IkeNotifyMessageType,
    IkeHashAlgorithm, IpsecDoiTransformIdent, IkeExchangeType,
    IpsecDoiSecProtocolId           FROM IPSEC-ISAKMP-IKE-DOI-TC
    OBJECT-GROUP, NOTIFICATION-GROUP, MODULE-COMPLIANCE
                                    FROM SNMPv2-CONF;

ikeMonModule MODULE-IDENTITY
    LAST-UPDATED    "0110031200Z"
    ORGANIZATION    "IETF IPsec Working Group"
    CONTACT-INFO
            "   Tim Jenkins
                Catena Networks
                307 Legget Drive
                Kanata, ON
                Canada
                K2K 3C8
                +1 (613) 599-6430
                tjenkins@catena.com

                John Shriver
                Intel Corporation
                28 Crosby Drive Bedford, MA
                01730
                +1 (781) 687-1329
                John.Shriver@intel.com
            "
    DESCRIPTION
        "The MIB module to describe IKE phase 1 SAs, security
        association suites, and entity level objects and events for
        those types."

    REVISION    "9910211200Z"
    DESCRIPTION
        "Initial revision."
```

```
    REVISION    "0007101200Z"
    DESCRIPTION
        "Group and compliance statements added.
        Endpoint table added and used in place of explicit phase 1
        IDs.
        Selector table from IPsec Monitoring MIB used in place of
        explicit selectors.
        Replaced addresses with types from INET-ADDRESS-MIB.
        Added IANA assigned experimental number of 106.
        Changes to notify parameters.
        More text pictures."

    REVISION    "0102071200Z"
    DESCRIPTION
        "Change MAX-ACCESS clause of index objects to
        not-accessible. This lead to other changes due to
        restrictions on the use of objects with MAX-ACCESS clause
        values of not-accessible."

    REVISION    "0110031200Z"
    DESCRIPTION
        "A number of typo errors corrected. Also:
         -- descriptions of suiteOakleyGroupDesc and
            suiteOakleyGroup enhanced
         -- change kilobytes to Kilobytes and make it 1024 bytes
         -- used plurals for some counter object names"

-- replace xxx in next line before release, uncomment before release
--     ::= { mib-2 xxx }
-- delete next line before release
        ::= { experimental 106 }


ikeMonMIBObjects OBJECT-IDENTITY
    STATUS      current
    DESCRIPTION
        "This is the base object identifier for all IKE monitoring
        MIB branches."
    ::= { ikeMonModule 1 }

--
-- significant branches
--

ikePhase1Objects OBJECT-IDENTITY
    STATUS  current
    DESCRIPTION
        "This is the base object identifier for IKE phase 1
        objects."
    ::= { ikeMonMIBObjects 1 }
```

```
ikePhase2Objects OBJECT-IDENTITY
    STATUS  current
    DESCRIPTION
        "This is the base object identifier for IKE phase 2 objects,
        including the suite and phase 2 SA tables."
    ::= { ikeMonMIBObjects 2 }

oakleyObjects OBJECT-IDENTITY
    STATUS  current
    DESCRIPTION
        "This is the base object identifier for Oakley groups."
    ::= { ikeMonMIBObjects 3 }

ikeGroups OBJECT-IDENTITY
    STATUS  current
    DESCRIPTION
        "This is the base object identifier for all objects which
        describe the groups in this MIB."
    ::= { ikeMonMIBObjects 4 }

ikeConformance OBJECT-IDENTITY
    STATUS  current
    DESCRIPTION
        "This is the base object identifier for all objects which
        describe the conformance for this MIB."
    ::= { ikeMonMIBObjects 5 }

--
-- significant IKE phase 1 SA branches
--

ikeTables OBJECT-IDENTITY
    STATUS  current

    DESCRIPTION
        "This is the base object identifier for the IKE phase 1
        security associations table."
    ::= { ikePhase1Objects 1 }

ikeGlobals OBJECT-IDENTITY
    STATUS  current
    DESCRIPTION
        "This is the base object identifier for all objects which
        are global values for IKE."
    ::= { ikePhase1Objects 2 }

ikeTrafStats OBJECT-IDENTITY
    STATUS  current
    DESCRIPTION
        "This is the base object identifier for all objects which
        are traffic statistic values for IKE."
```

```
    ::= { ikePhase1Objects 3 }

ikeErrors OBJECT-IDENTITY
    STATUS  current
    DESCRIPTION
        "This is the base object identifier for all objects which
        are error values for IKE."
    ::= { ikePhase1Objects 4 }

ikeTrapObjects OBJECT-IDENTITY
    STATUS  current
    DESCRIPTION
        "This is the base object identifier for all trap objects for
        the IKE phase 1 SA portion of this MIB."
    ::= { ikePhase1Objects 5 }

ikeTrapControl OBJECT-IDENTITY
    STATUS  current
    DESCRIPTION
        "This is the base object identifier for all trap controls
        for the IKE phase 1 SA portion of this MIB."
    ::= { ikePhase1Objects 6 }

ikeTraps OBJECT-IDENTITY
    STATUS  current
    DESCRIPTION
        "This is the base object identifier for all traps for the
        IKE phase 1 SA portion of this MIB."
    ::= { ikePhase1Objects 7 }

ikeNotifications OBJECT-IDENTITY
    STATUS  current
    DESCRIPTION
        "This is the base object identifier for all notification
        objects of this MIB."
    ::= { ikePhase1Objects 8 }

--
-- significant SA suite branches
--

suiteTables OBJECT-IDENTITY
    STATUS  current
    DESCRIPTION
        "This is the base object identifier for the suite table."
    ::= { ikePhase2Objects 1 }

suiteGlobals OBJECT-IDENTITY
    STATUS  current
    DESCRIPTION
        "This is the base object identifier for all objects which
```

```
        are global values for suites."
    ::= { ikePhase2Objects 2 }

suiteTrafStats OBJECT-IDENTITY
    STATUS  current
    DESCRIPTION
        "This is the base object identifier for all objects which
        are global counters for suite traffic statistics."
    ::= { ikePhase2Objects 3 }

suiteErrors OBJECT-IDENTITY
    STATUS  current
    DESCRIPTION
        "This is the base object identifier for all objects which
        are global error counters for suites."
    ::= { ikePhase2Objects 4 }

suiteTrapControl OBJECT-IDENTITY
    STATUS  current
    DESCRIPTION
        "This is the base object identifier for all trap controls
        for the suite portion of this MIB."
    ::= { ikePhase2Objects 5 }

suiteTraps OBJECT-IDENTITY
    STATUS  current
    DESCRIPTION
        "This is the base object identifier for all traps for the
        suite portion of this MIB."
    ::= { ikePhase2Objects 6 }

--
-- the Oakley Group MIB-Group
--
-- a collection of objects providing information about the
-- Oakley Groups that the entity knows about that are not well known
--
-- A table is defined for each type of Oakley group
-- (each value in 'IkeGroupDescription').
--
-- This MIB has tables for groups of type MODP, ECP, or EC2N.
-- For groups that are not MODP, ECP, or EC2N, a new table should be
-- defined in a MIB for that group.  The table should have one
-- integer index, which should be the first column.  The columns
-- should be the IKE attributes used by that new type of group.
--


modpGroupTable OBJECT-TYPE
    SYNTAX      SEQUENCE OF ModpGroupEntry
    MAX-ACCESS  not-accessible
```

```
    STATUS      current
    DESCRIPTION
        "The (conceptual) table containing Oakley MODP groups that
        are not well known that the entity has negotiated or knows
        about.

        There should be one row for every Oakley MODP group
        negotiated or supported by the entity that is not a well-
        known group. The maximum number of rows is implementation
        dependent."
    ::= { oakleyObjects 1 }

modpGroupEntry OBJECT-TYPE
    SYNTAX      ModpGroupEntry
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "An entry (conceptual row) containing the information on a
        particular Oakley MODP group.

        A row in this table cannot be created or deleted by SNMP
        operations on columns of the table."
    INDEX   { modpGroupIndex }
    ::= { modpGroupTable 1 }

ModpGroupEntry ::= SEQUENCE {
    modpGroupIndex          Unsigned32,

-- component parts
    modpFieldSize           Unsigned32,
    modpPrime               OCTET STRING,
    modpGenerator           OCTET STRING,
    modpLPF                 OCTET STRING,
    modpStrength            Unsigned32
}

modpGroupIndex OBJECT-TYPE
    SYNTAX      Unsigned32 (1..16777215)
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "A unique value, greater than zero, for each Oakley MODP
        group. It is recommended that values are assigned
        contiguously starting from 1.

        The value for each MODP group must remain constant at least
        from one re-initialization of entity's network management
        system to the next re-initialization."
    ::= { modpGroupEntry 1 }

modpFieldSize OBJECT-TYPE
```

```
    SYNTAX      Unsigned32
    UNITS       "bits"
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The size of a field element, in bits."
    REFERENCE   "RFC 2412 Appendix A"
    ::= { modpGroupEntry 2 }

modpPrime OBJECT-TYPE
    SYNTAX      OCTET STRING (SIZE (0..511))
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The prime of the MODP group."
    REFERENCE   "RFC 2412 Appendix A"
    ::= { modpGroupEntry 3 }

modpGenerator OBJECT-TYPE
    SYNTAX      OCTET STRING (SIZE (0..511))
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The generator value of the MODP group."
    REFERENCE   "RFC 2412 Appendix A"
    ::= { modpGroupEntry 4 }

modpLPF      OBJECT-TYPE
    SYNTAX      OCTET STRING (SIZE (0..511))
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The largest prime factor of the group size, or 0 if
        unspecified."
    REFERENCE   "RFC 2412 Appendix A"
    ::= { modpGroupEntry 5 }

modpStrength OBJECT-TYPE
    SYNTAX      Unsigned32
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The strength of the group, which is approximately the
        number of key-bits protected, or 0 if unspecified."
    REFERENCE   "RFC 2412 Appendix A"
    ::= { modpGroupEntry 6 }


ecpGroupTable OBJECT-TYPE
    SYNTAX      SEQUENCE OF EcpGroupEntry
    MAX-ACCESS  not-accessible
```

```
    STATUS      current
    DESCRIPTION
        "The (conceptual) table containing Oakley ECP groups that
        are not well known that the entity has negotiated or knows
        about.

        There should be one row for every Oakley ECP group
        negotiated or supported by the entity that is not a well-
        known group. The maximum number of rows is implementation
        dependent."
    ::= { oakleyObjects 2 }

ecpGroupEntry OBJECT-TYPE
    SYNTAX      EcpGroupEntry
    MAX-ACCESS  not-accessible        STATUS      current
    DESCRIPTION
        "An entry (conceptual row) containing the information on a
        particular Oakley ECP group.

        A row in this table cannot be created or deleted by SNMP
        operations on columns of the table."
    INDEX   { ecpGroupIndex }
    ::= { ecpGroupTable 1 }

EcpGroupEntry  ::= SEQUENCE {
    ecpGroupIndex       Unsigned32,

-- component parts
    ecpFieldSize            Unsigned32,
    ecpPrime                OCTET STRING,
    ecpGeneratorOne         OCTET STRING,
    ecpGeneratorTwo         OCTET STRING,
    ecpParameterOne         OCTET STRING,
    ecpParameterTwo         OCTET STRING,
    ecpLPF                  OCTET STRING,
    ecpOrder                OCTET STRING,
    ecpStrength             Unsigned32
}

ecpGroupIndex OBJECT-TYPE
    SYNTAX      Unsigned32 (1..16777215)
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "A unique value, greater than zero, for each Oakley ECP
        group. It is recommended that values are assigned
        contiguously starting from 1.

        The value for each ECP group must remain constant at least
        from one re-initialization of entity's network management
        system to the next re-initialization."
```

```
    ::= { ecpGroupEntry 1 }

ecpFieldSize OBJECT-TYPE
    SYNTAX      Unsigned32
    UNITS       "bits"
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The size of a field element, in bits."
    REFERENCE   "RFC 2412 Appendix A"
    ::= { ecpGroupEntry 2 }
ecpPrime OBJECT-TYPE
    SYNTAX      OCTET STRING (SIZE (0..511))
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The prime of the ECP group."
    REFERENCE   "RFC 2412 Appendix A"
    ::= { ecpGroupEntry 3 }

ecpGeneratorOne  OBJECT-TYPE
    SYNTAX      OCTET STRING (SIZE (0..511))
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The first generator value of the group."
    REFERENCE   "RFC 2412 Appendix A"
    ::= { ecpGroupEntry 4 }

ecpGeneratorTwo OBJECT-TYPE
    SYNTAX      OCTET STRING (SIZE (0..511))
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The second generator value of the group."
    REFERENCE   "RFC 2412 Appendix A"
    ::= { ecpGroupEntry 5 }

ecpParameterOne OBJECT-TYPE
    SYNTAX      OCTET STRING (SIZE (0..511))
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The first elliptic curve parameter value of the group."
    REFERENCE   "RFC 2412 Appendix A"
    ::= { ecpGroupEntry 6 }

ecpParameterTwo OBJECT-TYPE
    SYNTAX      OCTET STRING (SIZE (0..511))
    MAX-ACCESS  read-only
    STATUS      current
```

```
    DESCRIPTION
        "The second elliptic curve parameter value of the group."
    REFERENCE   "RFC 2412 Appendix A"
    ::= { ecpGroupEntry 7 }

ecpLPF OBJECT-TYPE
    SYNTAX      OCTET STRING (SIZE (0..511))        MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The largest prime factor of the group size, or 0 if
        unspecified."
    REFERENCE   "RFC 2412 Appendix A"
    ::= { ecpGroupEntry 8 }

ecpOrder OBJECT-TYPE
    SYNTAX      OCTET STRING (SIZE (0..511))
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The order of the group, or 0 if it is unspecified."
    REFERENCE   "RFC 2412 Appendix A"
    ::= { ecpGroupEntry 9 }

ecpStrength OBJECT-TYPE
    SYNTAX      Unsigned32
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The strength of the group, which is approximately the
        number of key-bits protected."
    REFERENCE   "RFC 2412 Appendix A"
    ::= { ecpGroupEntry 10 }


ec2nGroupTable  OBJECT-TYPE
    SYNTAX      SEQUENCE OF Ec2nGroupEntry
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "The (conceptual) table containing Oakley EC2N groups that
        are not well known that the entity has negotiated or knows
        about.

        There should be one row for every Oakley group negotiated or
        supported by the entity that is not a well-known group. The
        maximum number of rows is implementation dependent."
    ::= { oakleyObjects 3 }

ec2nGroupEntry OBJECT-TYPE
    SYNTAX      Ec2nGroupEntry
    MAX-ACCESS  not-accessible
```

```
    STATUS      current

    DESCRIPTION
        "An entry (conceptual row) containing the information on a
        particular Oakley EC2N group.

        A row in this table cannot be created or deleted by SNMP
        operations on columns of the table."
    INDEX   { ec2nGroupIndex }
    ::= { ec2nGroupTable 1 }

Ec2nGroupEntry  ::= SEQUENCE {
    ec2nGroupIndex          Unsigned32,

-- component parts
    ec2nDegree              Unsigned32,
    ec2nIrrPoly             OCTET STRING,
    ec2nGeneratorOne        OCTET STRING,
    ec2nGeneratorTwo        OCTET STRING,
    ec2nParameterOne        OCTET STRING,
    ec2nParameterTwo        OCTET STRING,
    ec2nLPF                 OCTET STRING,
    ec2nOrder               OCTET STRING,
    ec2nStrength            Unsigned32
}

ec2nGroupIndex OBJECT-TYPE
    SYNTAX      Unsigned32 (1..16777215)
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "A unique value, greater than zero, for each Oakley EC2N
        group. It is recommended that values are assigned
        contiguously starting from 1.

        The value for each EC2N group must remain constant at least
        from one re-initialization of entity's network management
        system to the next re-initialization."
    ::= { ec2nGroupEntry 1 }

ec2nDegree OBJECT-TYPE
    SYNTAX      Unsigned32
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The degree of the irreducible polynomial."
    REFERENCE   "RFC 2412 Appendix A"
    ::= { ec2nGroupEntry 2 }

ec2nIrrPoly  OBJECT-TYPE
    SYNTAX      OCTET STRING (SIZE (0..511))
```

```
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The prime or the irreducible field polynomial."
    REFERENCE   "RFC 2412 Appendix A"
    ::= { ec2nGroupEntry 3 }

ec2nGeneratorOne OBJECT-TYPE
    SYNTAX      OCTET STRING (SIZE (0..511))
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The first generator value of the group."
    REFERENCE   "RFC 2412 Appendix A"
    ::= { ec2nGroupEntry 4 }

ec2nGeneratorTwo OBJECT-TYPE
    SYNTAX      OCTET STRING (SIZE (0..511))
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The second generator value of the group."
    REFERENCE   "RFC 2412 Appendix A"
    ::= { ec2nGroupEntry 5 }

ec2nParameterOne OBJECT-TYPE
    SYNTAX      OCTET STRING (SIZE (0..511))
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The first elliptic curve parameter value of the group."
    REFERENCE   "RFC 2412 Appendix A"
    ::= { ec2nGroupEntry 6 }

ec2nParameterTwo OBJECT-TYPE
    SYNTAX      OCTET STRING (SIZE (0..511))
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The second elliptic curve parameter value of the group."
    REFERENCE   "RFC 2412 Appendix A"
    ::= { ec2nGroupEntry 7 }

ec2nLPF     OBJECT-TYPE
    SYNTAX      OCTET STRING (SIZE (0..511))
    MAX-ACCESS  read-only        STATUS      current
    DESCRIPTION
        "The largest prime factor of the group size, or 0 if
        unspecified."
    REFERENCE   "RFC 2412 Appendix A"
    ::= { ec2nGroupEntry 8 }
```

```
ec2nOrder OBJECT-TYPE
    SYNTAX      OCTET STRING (SIZE (0..511))
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The order of the group, or 0 if it is unspecified."
    REFERENCE   "RFC 2412 Appendix A"
    ::= { ec2nGroupEntry 9 }

ec2nStrength OBJECT-TYPE
    SYNTAX      Unsigned32
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The strength of the group, which is approximately the
        number of key-bits protected, or 0 if it is unspecified."
    REFERENCE   "RFC 2412 Appendix A"
    ::= { ec2nGroupEntry 10 }


--
-- the IKE Endpoint Table
--
-- a collection of objects providing information about
-- the endpoints involved with IKE in this entity
--

ikeEndpointTable OBJECT-TYPE
    SYNTAX      SEQUENCE OF IkeEndpointEntry
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "The (conceptual) table containing information about the
        endpoints involved IKE in this entity.

        There is one row for each endpoint that is active in or with
        the entity, including remote endpoints and local endpoints.

        The maximum number of rows is implementation dependent."
    ::= { ikeTables 1 }
ikeEndpointEntry OBJECT-TYPE
    SYNTAX      IkeEndpointEntry
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "An entry (conceptual row) containing an IKE ID.

        A row in this table cannot be created or deleted by SNMP
        operations on columns of the table.
```

```
        It is not necessary to delete rows for endpoints that are no
        longer active; this is implementation dependent."
    INDEX   {   endpointIndex   }
    ::= { ikeEndpointTable 1 }

IkeEndpointEntry   ::= SEQUENCE {
 -- index
    endpointIndex               Unsigned32,

-- ID and authentication information
    endpointIdType              IpsecDoiIdentType,
    endpointIdValue             IpsecRawId,
    endpointCertSerialNum       OCTET STRING,
    endpointCertIssuer          OCTET STRING,

-- other info about the ID, including statistics
    endpointIsLocal             TruthValue,
    endpointCurrentIkeSAs       Gauge32,
    endpointTotalIkeSAs         Counter32,
    endpointCurrentSuites       Gauge32,
    endpointTotalSuites         Counter32

}

endpointIndex OBJECT-TYPE
    SYNTAX      Unsigned32
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "A unique value, greater than zero, for each endpoint
        associated with the entity, whether local or remote. It is
        recommended that values are assigned contiguously starting
        from 1."
    ::= { ikeEndpointEntry 1 }

endpointIdType OBJECT-TYPE
    SYNTAX      IpsecDoiIdentType
    MAX-ACCESS  read-only        STATUS      current
    DESCRIPTION
        "The type of ID used by the endpoint. This is the type of
        the ID that is used by the endpoint during phase 1
        negotiations.

        If this is not a local endpoint, then this value is taken
        directly from the phase 1 exchange with the remote
        endpoint."
    REFERENCE   "RFC 2407 Section 4.6.2.1"
    ::= { ikeEndpointEntry 2 }

endpointIdValue OBJECT-TYPE
    SYNTAX      IpsecRawId
```

```
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The ID of the endpoint. This is the ID value that is used
        by the endpoint during phase 1 negotiations.

        If this is not a local endpoint, then this value is taken
        directly from the phase 1 exchange with the remote
        endpoint."
    REFERENCE   "RFC 2407 Section 4.6.2.1"
    ::= { ikeEndpointEntry 3 }

endpointCertSerialNum OBJECT-TYPE
    SYNTAX      OCTET STRING (SIZE (0..63))
    MAX-ACCESS  read-only
    STATUS  current
    DESCRIPTION
        "The serial number of the certificate used by the endpoint.

        This object has no meaning if a certificate was not used in
        authenticating the endpoint."
    ::= { ikeEndpointEntry 4 }

endpointCertIssuer OBJECT-TYPE
    SYNTAX      OCTET STRING (SIZE (0..511))
    MAX-ACCESS  read-only
    STATUS  current
    DESCRIPTION
        "The issuer name of the certificate used by the endpoint.

        This object has no meaning if a certificate was not used in
        authenticating the endpoint."
    ::= { ikeEndpointEntry 5 }
endpointIsLocal OBJECT-TYPE
    SYNTAX      TruthValue
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "True if this row represents a local endpoint (the entity
        uses this endpoint)."
    ::= { ikeEndpointEntry 6 }

endpointCurrentIkeSAs OBJECT-TYPE
    SYNTAX      Gauge32
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The number of current IKE SAs in the entity for which this
        endpoint is found at one end."
    ::= { ikeEndpointEntry 7 }
```

```
endpointTotalIkeSAs OBJECT-TYPE
    SYNTAX      Counter32
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The total number of IKE SAs in the entity for which this
        endpoint is or was found at one end."
    ::= { ikeEndpointEntry 8 }

endpointCurrentSuites OBJECT-TYPE
    SYNTAX      Gauge32
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The number of current phase 2 SA suites in the entity that
        this endpoint was involved in the creation of."
    ::= { ikeEndpointEntry 9 }

endpointTotalSuites  OBJECT-TYPE
    SYNTAX      Counter32
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        " The total number of phase 2 SA suites in the entity that
        this endpoint was involved in the creation of."
    ::= { ikeEndpointEntry 10 }



--
-- the IKE Phase 1 SA MIB-Group
--
-- a collection of objects providing information about
-- the IKE phase 1 SAs
--

ikeSaTable OBJECT-TYPE
    SYNTAX      SEQUENCE OF IkeSaEntry
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "The (conceptual) table containing the IKE SAs.

        The number of rows is the same as the number of IKE phase 2
        SAs that are in the process of being negotiated or are
        negotiated in the entity. Phrased another way, there is a
        row in this table for each row in 'saTable' for which
        'saDoi' is 'ipsecDOI(1)'.

        The maximum number of rows is implementation dependent."
    ::= { ikeTables 2 }
```

```
ikeSaEntry OBJECT-TYPE
    SYNTAX      IkeSaEntry
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "An entry (conceptual row) containing the information on a
        particular IKE SA. There is an entry in this table for each
        'saEntry' in which which 'saDoi' is 'ipsecDOI(1)'.

        A row in this table cannot be created or deleted by SNMP
        operations on columns of the table."
    INDEX
        {
            saLocalIpAddressType,
            saLocalIpAddress,
            saRemoteIpAddressType,
            saRemoteIpAddress,
            saInitiatorCookie,
            saResponderCookie
        }
    ::= { ikeSaTable 1 }

IkeSaEntry ::= SEQUENCE {
-- ID and authentication information
    saAuthMethod                IkeAuthMethod,
    saPeerEndpoint              Unsigned32,
    saLocalEndpoint             Unsigned32,

-- security algorithm information
    saEncAlg                    IkeEncryptionAlgorithm,
    saEncKeyLength              Unsigned32,
    saHashAlg                   IkeHashAlgorithm,
    saHashKeyLength             Unsigned32,
    saPRF                       IkePrf,
    saOakleyGroupDesc           IkeGroupDescription,
    saOakleyGroup               OBJECT IDENTIFIER,

-- expiration limits
    saLimitSeconds              Unsigned32, -- 0 if none
    saLimitKbytes               Unsigned32, -- 0 if none
    saLimitKeyUses              Unsigned32, -- 0 if none

-- current operating statistics
    saAccKbytes                 Counter32,
    saKeyUses                   Counter32,
    saCreatedSuites             Counter32,
    saDeletedSuites             Counter32,

-- error counts
    saDecryptErrors             Counter32,
```

```
    saHashErrors                Counter32,
    saOtherReceiveErrors        Counter32,
    saSendErrors                Counter32
}

saAuthMethod OBJECT-TYPE
    SYNTAX      IkeAuthMethod
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The authentication method used to authenticate the peers.

        Note that this does not include the specific method of
        extended authentication if extended authentication is used."
    ::= { ikeSaEntry 1 }

saPeerEndpoint OBJECT-TYPE
    SYNTAX      Unsigned32
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The index of the endpoint table row for the peer endpoint
        that negotiated this SA. In other words, the value of
        'endpointIndex' for the appropriate row ('ikeEndpointEntry')
        from the 'ikeEndpointTable'."
    ::= { ikeSaEntry 2 }

saLocalEndpoint OBJECT-TYPE
    SYNTAX      Unsigned32
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The index of the endpoint table row for the local endpoint
        that negotiated this SA. In other words, the value of
        'endpointIndex' for the appropriate row ('ikeEndpointEntry')
        from the 'ikeEndpointTable'."
    ::= { ikeSaEntry 3 }

saEncAlg OBJECT-TYPE
    SYNTAX      IkeEncryptionAlgorithm
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The encryption algorithm used to protect this SA."
    ::= { ikeSaEntry 4 }

saEncKeyLength OBJECT-TYPE
    SYNTAX      Unsigned32 (0..65531)
    UNITS       "bits"
    MAX-ACCESS  read-only
    STATUS      current
```

```
    DESCRIPTION
        "The length of the encryption key in bits used for the
        algorithm specified in the 'saEncAlg' object. It may be 0 if
        the key length is implicit in the specified algorithm."
    ::= { ikeSaEntry 5 }

saHashAlg OBJECT-TYPE
    SYNTAX      IkeHashAlgorithm
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The hash algorithm used to protect this SA."
    ::= { ikeSaEntry 6 }

saHashKeyLength OBJECT-TYPE
    SYNTAX      Unsigned32 (0..65531)
    UNITS       "bits"
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The length of the encryption key in bits used for the
        algorithm specified in the 'saHashAlg' object. It may be 0
        if the key length is implicit in the specified algorithm."
    ::= { ikeSaEntry 7 }

saPRF OBJECT-TYPE
    SYNTAX      IkePrf
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The pseudo-random function used by this SA, or 0 if the
        HMAC version of the negotiated hash algorithm is used as a
        pseudo-random function."
    REFERENCE   "RFC 2409 Appendix A"
    ::= { ikeSaEntry 8 }

saOakleyGroupDesc OBJECT-TYPE
    SYNTAX      IkeGroupDescription
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The group number used to generate the Diffie-Hellman key
        pair when setting up the SA, or 0 if none of the defined
        groups was used.

        If this value is 0, the 'saOakleyGroup' must not also be
        OBJECT IDENTIFIER { 0 0 }."
    REFERENCE   "RFC 2409 Section 6."
    ::= { ikeSaEntry 9 }

saOakleyGroup OBJECT-TYPE
```

```
    SYNTAX   OBJECT IDENTIFIER
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The object identifier of the Oakley group row that was used
        if a well-known group was not used to generate the Diffie-
        Hellman key pair for this SA.

        If a well-known group was used, the value should be set to
        the OBJECT IDENTIFIER { 0 0 }.

        For example, if the group is a MODP group, the value of this
        object is the object identifier of 'modpGroupIndex' of the
        appropriate row ('modpGroupEntry') in 'modpGroupTable'."
    REFERENCE   "RFC 2409 Section 6"
    ::= { ikeSaEntry 10 }
saLimitSeconds OBJECT-TYPE
    SYNTAX      Unsigned32
    UNITS       "seconds"
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The maximum number of seconds the SA is allowed to exist,
        or 0 if there is no time-based limit on the existence of the
        SA.

        The display value is limited to 4,294,967,295 seconds (more
        than 136 years); values greater than that value will be
        truncated."
    ::= { ikeSaEntry 11 }

saLimitKbytes OBJECT-TYPE
    SYNTAX      Unsigned32
    UNITS       "Kilobytes"
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The maximum number of Kilobytes (1024 bytes) the SA is
        allowed to encrypt before it expires, or 0 if there is no
        traffic-by-byte-based limit on the existence of the SA.

        The display value is limited to 4,294,967,295 Kilobytes
        (more than 4,194,304 Mbyte); values greater than that value
        will be truncated."
    ::= { ikeSaEntry 12 }

saLimitKeyUses OBJECT-TYPE
    SYNTAX      Unsigned32
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
```

```
        "The maximum number of times the SA is allowed to provide
        keying material from its own Diffie-Hellman exchange before
        it expires, or 0 if there is no keying material-based limit
        on the existence of the SA."
    ::= { ikeSaEntry 13 }

saAccKbytes OBJECT-TYPE
    SYNTAX      Counter32
    UNITS       "Kilobytes"
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The number of Kilobytes (1024 bytes) the SA has encrypted
        that count against any lifetime restriction based on
        traffic. This value may be 0 if there is no such
        restriction."
    ::= { ikeSaEntry 14 }

saKeyUses OBJECT-TYPE
    SYNTAX      Counter32
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The number of times the SA has provided keying material
        derived from its own original Diffie-Hellman exchange."
    ::= { ikeSaEntry 15 }

saCreatedSuites OBJECT-TYPE
    SYNTAX      Counter32
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The total number of SA suites that this SA has successfully
        created. In other words, the total number of successful
        quick mode exchanges multiplied by the number of SA payloads
        in each of those exchanges."
    ::= { ikeSaEntry 16 }

saDeletedSuites OBJECT-TYPE
    SYNTAX      Counter32
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The total number of SA suites deleted for which this SA
        sent or received SA suite delete notifications. When delete
        notifications are sent or received for more than one IPsec
        SA in an SA suite, this number shall be incremented by one,
        and not by the number IPsec SAs in the suite that were
        deleted."
    ::= { ikeSaEntry 17 }
```

```
saDecryptErrors OBJECT-TYPE
    SYNTAX      Counter32
    UNITS       "packets"
    MAX-ACCESS  read-only
    STATUS      current


    DESCRIPTION
        "The total number of packets inbound to this SA that were
        discarded due to decryption errors."
    ::= { ikeSaEntry 18 }

saHashErrors OBJECT-TYPE
    SYNTAX      Counter32
    UNITS       "packets"
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The total number of packets inbound to this SA that were
        discarded due to hash result errors."
    ::= { ikeSaEntry 19 }

saOtherReceiveErrors OBJECT-TYPE
    SYNTAX      Counter32
    UNITS       "packets"
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The total number of packets inbound to this SA that were
        discarded due to errors other than decryption or hash result
        errors. This may include packets dropped to a lack of
        receive buffer space."
    ::= { ikeSaEntry 20 }

saSendErrors OBJECT-TYPE
    SYNTAX      Counter32
    UNITS       "packets"
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The total number of packets outbound from this SA that were
        discarded due to errors. This may include packets dropped to
        a lack of transmit buffer space."
    ::= { ikeSaEntry 21 }

--
-- the IKE SA By Creators Table
--

saByCreatorsTable OBJECT-TYPE
    SYNTAX      SEQUENCE OF SaByCreatorsEntry
```

```
    MAX-ACCESS  not-accessible
    STATUS      current

    DESCRIPTION
        "The (conceptual) table that sorts the IKE phase 1 SAs by
        the endpoint identifiers.

        The number of rows in this table is the same as the number
        of IKE phase 1 SAs in the entity."
    ::= { ikeTables 3 }

saByCreatorsEntry OBJECT-TYPE
    SYNTAX      SaByCreatorsEntry
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "An entry (conceptual row) referencing a particular IKE
        phase 1 SA.

        A row in this table cannot be created or deleted by SNMP
        operations on columns of the table."
    INDEX
        {
            saByCreatorsLocalEndpoint,
            saByCreatorsRemoteEndpoint,
            saByCreatorsIndex
        }
    ::= { saByCreatorsTable 1 }

SaByCreatorsEntry  ::= SEQUENCE {
-- index
    saByCreatorsLocalEndpoint   Unsigned32,
    saByCreatorsRemoteEndpoint  Unsigned32,
    saByCreatorsIndex           Unsigned32,

-- phase 1 SA reference
    saIkeLocalIpAddressType     InetAddressType,
    saIkeLocalIpAddress         InetAddress,
    saIkeRemoteIpAddressType    InetAddressType,
    saIkeRemoteIpAddress        InetAddress,
    saIkeInitiatorCookie        IsakmpCookie,
    saIkeResponderCookie        IsakmpCookie
}

saByCreatorsLocalEndpoint OBJECT-TYPE
    SYNTAX      Unsigned32
    MAX-ACCESS  not-accessible
    STATUS      current


    DESCRIPTION
```

```
        "The index of the endpoint table row for the local
        endpoint."
    ::= { saByCreatorsEntry 1 }

saByCreatorsRemoteEndpoint OBJECT-TYPE
    SYNTAX      Unsigned32
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "The index of the endpoint table row for the remote
        endpoint."
    ::= { saByCreatorsEntry 2 }

saByCreatorsIndex OBJECT-TYPE
    SYNTAX      Unsigned32 (1..16777215)
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "A unique value, greater than zero, for each IKE phase 1 SA
        that exists between the two endpoints. It is recommended
        that values are assigned contiguously starting from 1."
    ::= { saByCreatorsEntry 3 }

saIkeLocalIpAddressType OBJECT-TYPE
    SYNTAX      InetAddressType
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The value of 'saLocalIpAddressType' of the phase 1 SA for
        this row."
    ::= { saByCreatorsEntry 4 }

saIkeLocalIpAddress OBJECT-TYPE
    SYNTAX      InetAddress (SIZE(4|16|20))
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The value of 'saLocalIpAddress' of the phase 1 SA for this
        row."
    ::= { saByCreatorsEntry 5 }

saIkeRemoteIpAddressType OBJECT-TYPE
    SYNTAX      InetAddressType
    MAX-ACCESS  read-only
    STATUS      current

    DESCRIPTION
        "The value of 'saRemoteIpAddressType' of the phase 1 SA for
        this row."
    ::= { saByCreatorsEntry 6 }
```

```
saIkeRemoteIpAddress OBJECT-TYPE
    SYNTAX      InetAddress (SIZE(4|16|20))
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The value of 'saRemoteIpAddress' of the phase 1 SA for this
        row."
    ::= { saByCreatorsEntry 7 }

saIkeInitiatorCookie OBJECT-TYPE
    SYNTAX      IsakmpCookie
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The value of 'saInitiatorCookie' of the phase 1 SA for this
        row."
    ::= { saByCreatorsEntry 8 }

saIkeResponderCookie OBJECT-TYPE
    SYNTAX      IsakmpCookie
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The value of 'saResponderCookie' of the phase 1 SA for this
        row."
    ::= { saByCreatorsEntry 9 }


-- the Exchange Count MIB-Group
--
-- a collection of objects providing information about the
-- number of exchanges performed using ISAKMP-based SAs
--

exchangeTable OBJECT-TYPE
    SYNTAX      SEQUENCE OF ExchangeEntry
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "The (conceptual) table containing the exchanges used.

        There should be one row for every exchange attempt that has
        occurred using a phase 1 security association that exists in
        the entity. The maximum number of rows is implementation
        dependent."
    ::= { ikeTables 4 }

exchangeEntry OBJECT-TYPE
    SYNTAX      ExchangeEntry
    MAX-ACCESS  not-accessible
    STATUS      current
```

```
    DESCRIPTION
        "An entry (conceptual row) containing the information on a
        particular exchange used in an SA.

        A row in this table cannot be created or deleted by SNMP
        operations on columns of the table."
    INDEX   {
            saLocalIpAddressType,
            saLocalIpAddress,
            saRemoteIpAddressType,
            saRemoteIpAddress,
            saInitiatorCookie,
            saResponderCookie,
            exchangeType
            }
    ::= { exchangeTable 1 }

ExchangeEntry::= SEQUENCE {
-- identification
    exchangeType            IkeExchangeType,

-- the statistics
    exchangesTotalCount     Counter32,
    exchangesInitiatedCount Counter32,
    exchangesRespondedCount Counter32
}

exchangeType OBJECT-TYPE
    SYNTAX      IkeExchangeType
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "The type of the exchange for which the statistics of this
        row apply."
    ::= { exchangeEntry 1 }

exchangesTotalCount OBJECT-TYPE
    SYNTAX      Counter32
    MAX-ACCESS  read-only
    STATUS      current        DESCRIPTION
        "The total number of complete exchanges of the type
        performed using the SA, as either initiator or as responder.

        If there were failed attempts to initiate exchanges, this
        value is not equal to the sum of 'exchangesInitiatedCount'
        and 'exchangesRespondedCount'."
    ::= { exchangeEntry 2 }

exchangesInitiatedCount OBJECT-TYPE
    SYNTAX      Counter32
    MAX-ACCESS  read-only
```

```
    STATUS      current
    DESCRIPTION
        "The total number of exchanges of the type attempted using
        the SA as initiator. This includes exchange that failed or
        were incomplete"
    ::= { exchangeEntry 3 }

exchangesRespondedCount OBJECT-TYPE
    SYNTAX      Counter32
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The total number of complete exchanges of the type
        performed using the SA as responder."
    ::= { exchangeEntry 4 }

--
-- the Suite MIB-Group
--
-- a collection of objects providing information about
-- the phase 2 SA suites
--

suiteTable OBJECT-TYPE
    SYNTAX      SEQUENCE OF SuiteEntry
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "The (conceptual) table containing the phase 2 suites.

        The number of rows in this table is the same as the number
        of suites in the entity. The maximum number of rows is
        implementation dependent."
    ::= { suiteTables 1 }

suiteEntry OBJECT-TYPE
    SYNTAX      SuiteEntry
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "An entry (conceptual row) containing the information on a
        particular phase 2 SA suite.

        A row in this table cannot be created or deleted by SNMP
        operations on columns of the table."
    INDEX   { suiteIndex }
    ::= { suiteTable 1 }

SuiteEntry ::= SEQUENCE {
-- index
    suiteIndex                  Unsigned32,
```

```
-- end points
    suiteLocalAddressType       InetAddressType,
    suiteLocalAddress           InetAddress,
    suiteRemoteAddressType      InetAddressType,
    suiteRemoteAddress          InetAddress,

-- creator ID information
    suitePhase1RemoteEndpoint   Unsigned32,
    suitePhase1LocalEndpoint    Unsigned32,

-- selector
    suiteSelector               Unsigned32,

-- keying material source information
    suiteOakleyGroupDesc        IkeGroupDescription,
    suiteOakleyGroup            OBJECT IDENTIFIER,

-- operating statistics
    suiteLifeSeconds            Counter32,
    suiteInUserOctets           Counter64,
    suiteInPackets              Counter64,
    suiteOutUserOctets          Counter64,
    suiteOutPackets             Counter64,

-- error statistics
    suiteSendErrors             Counter32,
    suiteReceiveErrors          Counter32
}

suiteIndex OBJECT-TYPE
    SYNTAX      Unsigned32 (1..16777215)        MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "A unique value, greater than zero, for each SA suite. It is
        recommended that values are assigned contiguously starting
        from 1."
    ::= { suiteEntry 1 }

suiteLocalAddressType OBJECT-TYPE
    SYNTAX      InetAddressType
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The type of address used by the local entity that
        negotiated the SA suite. "
    ::= { suiteEntry 2 }

suiteLocalAddress OBJECT-TYPE
    SYNTAX      InetAddress (SIZE(4|16|20))
    MAX-ACCESS  read-only
```

```
    STATUS      current
    DESCRIPTION
        "The address used by the local entity that negotiated the SA
        suite. "
    ::= { suiteEntry 3 }

suiteRemoteAddressType OBJECT-TYPE
    SYNTAX      InetAddressType
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The type of address used by the remote entity that
        negotiated the SA suite."
    ::= { suiteEntry 4 }

suiteRemoteAddress OBJECT-TYPE
    SYNTAX      InetAddress (SIZE(4|16|20))
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The address used by the remote entity that negotiated the
        SA suite."
    ::= { suiteEntry 5 }

suitePhase1RemoteEndpoint OBJECT-TYPE
    SYNTAX      Unsigned32
    MAX-ACCESS  read-only
    STATUS      current        DESCRIPTION
        "The index of the endpoint table row for remote entity that
        negotiated this suite. In other words, the value of
        'endpointIndex' for the appropriate row ('ikeEndpointEntry')
        from the 'ikeEndpointTable'."
    ::= { suiteEntry 6 }

suitePhase1LocalEndpoint OBJECT-TYPE
    SYNTAX      Unsigned32
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The index of the endpoint table row for local entity that
        negotiated this suite. In other words, the value of
        'endpointIndex' for the appropriate row ('ikeEndpointEntry')
        from the 'ikeEndpointTable'"
    ::= { suiteEntry 7 }

suiteSelector OBJECT-TYPE
    SYNTAX      Unsigned32
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The index of the selector table row for this suite. In
```

```
        other words, the value of 'selectorIndex' for the
        appropriate row ('SelectorEntry') from the 'selectorTable'"
    ::= { suiteEntry 8 }

suiteOakleyGroupDesc OBJECT-TYPE
    SYNTAX      IkeGroupDescription
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The group number used to generate the Diffie-Hellman key
        pair when setting up the SA, or 0 if none of the well known
        groups was used, or if perfect forward secrecy was not used.

        If this value is 0, the 'suiteOakleyGroup' must not also be
        OBJECT IDENTIFIER { 0 0 }."
    ::= { suiteEntry 9 }

suiteOakleyGroup OBJECT-TYPE
    SYNTAX      OBJECT IDENTIFIER
    MAX-ACCESS  read-only
    STATUS      current


    DESCRIPTION
        "The OID for the Oakley group row that was used if a well-
        known group was not used to generate the Diffie-Hellman key
        pair for this SA.

        If a well-known group was used, or if perfect forward
        secrecy was not used, the value should be set to the OBJECT
        IDENTIFIER { 0 0 }.

        For example, if the group is a MODP group, the value of this
        object is the object identifier of 'modpGroupIndex' of the
        appropriate row ('modpGroupEntry') in 'modpGroupTable'."
    ::= { suiteEntry 10 }

suiteLifeSeconds OBJECT-TYPE
    SYNTAX      Counter32
    UNITS       "seconds"
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The number of seconds that the SA has existed."
    ::= { suiteEntry 11 }

suiteInUserOctets OBJECT-TYPE
    SYNTAX      Counter64
    UNITS       "bytes"
    MAX-ACCESS  read-only
    STATUS      current
```

```
    DESCRIPTION
        "The amount of user level traffic measured in bytes handled
        by the suite in the inbound direction.

        This is the same as the user level traffic of the inner most
        inbound SA in the suite. Note that if the inner-most SA is a
        shared IPcomp SA, then this value may be difficult to
        calculate."
    ::= { suiteEntry 12 }

suiteInPackets OBJECT-TYPE
    SYNTAX      Counter64
    UNITS       "packets"
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The number of inbound packets handled by the suite.


        This is the same as the number of packets handled by any one
        of the inbound SAs in the suite."
    ::= { suiteEntry 13 }

suiteOutUserOctets OBJECT-TYPE
    SYNTAX      Counter64
    UNITS       "bytes"
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The amount of user level traffic measured in bytes handled
        by the suite in the outbound direction.

        This is the same as the user level traffic of the inner most
        outbound SA in the suite. Note that if the inner most SA is
        a shared IPcomp SA, then this value may be difficult to
        calculate."
    ::= { suiteEntry 14 }

suiteOutPackets OBJECT-TYPE
    SYNTAX      Counter64
    UNITS       "packets"
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The number of outbound packets handled by the suite.

        This is the same as the number of packets handled by any one
        of the outbound SAs in the suite."
    ::= { suiteEntry 15 }

suiteSendErrors  OBJECT-TYPE
```

```
    SYNTAX      Counter32
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The number of outbound packets discarded by the suite due
        to any error.

        This is the same as the sum of all errors of all outbound
        SAs in the suite."
    ::= { suiteEntry 16 }

suiteReceiveErrors OBJECT-TYPE
    SYNTAX      Counter32
    UNITS       "packets"
    MAX-ACCESS  read-only
    STATUS      current        DESCRIPTION
        "The number of inbound packets discarded by the suite due to
        any error.

        This is the same as the sum of all errors of all inbound SAs
        in the suite."
    ::= { suiteEntry 17 }


--
-- the Phase 2 SA MIB-Group
--
-- a collection of objects providing information about
-- the phase 2 SAs in SA suites
--

phase2SaTable OBJECT-TYPE
    SYNTAX      SEQUENCE OF Phase2SaEntry
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "The (conceptual) table containing ID information for the
        phase 2 SAs that are part of suites.

        The number of rows in this table is the same as the number
        of phase 2 IPsec SA pairs that are created as part of
        suites. The maximum number of rows is implementation
        dependent."
    ::= { suiteTables 3 }

phase2SaEntry OBJECT-TYPE
    SYNTAX      Phase2SaEntry
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "An entry (conceptual row) containing the information on a
```

```
        particular phase 2 SA within a suite.

        A row in this table cannot be created or deleted by SNMP
        operations on columns of the table."
    INDEX   { suiteIndex, saOrder }
    ::= { phase2SaTable 1 }

Phase2SaEntry  ::= SEQUENCE {
-- additional indexing objects
    saOrder             Unsigned32,

-- SA identifiers
    saProtocol          IpsecDoiTransformIdent,
    saInSpi             Unsigned32,
    saOutSpi            Unsigned32
}

saOrder     OBJECT-TYPE
    SYNTAX      Unsigned32 (1..15)
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "The position within the suite of the pair of SAs indicated
        by this row.

        A value of 1 is used to represent the outer-most SA pair.
        The outer-most SA of any given packet has its header next to
        the outer IP header of the processed packet, while the
        inner-most SA has its header nearest the data of the
        unprocessed packet. (Note that the IPcomp header may be
        missing in actual usage if a particular packet was not
        compressed.)

        This value should be monotonically increasing for every SA
        pair in a suite. The maximum value is implementation
        dependent, but will generally not exceed three."
    ::= { phase2SaEntry 1 }

saProtocol OBJECT-TYPE
    SYNTAX      IpsecDoiTransformIdent
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The protocol of the inbound/outbound SA pair indicated by
        this row of the table."
    ::= { phase2SaEntry 2 }

saInSpi OBJECT-TYPE
    SYNTAX      Unsigned32
    MAX-ACCESS  read-only
    STATUS      current
```

```
    DESCRIPTION
        "The security parameters index of the inbound SA of the
        inbound/outbound SA pair. If the protocol of the SA pair is
        IPcomp, this value is the CPI.

        This value is used with the value of 'suiteLocalAddress'
        from the row indexed by 'suiteIndex' to create a SPI/address
        pair that uniquely identifies the inbound SA used in this SA
        suite. This can then be used to look up the SA in the
        appropriate inbound SA table, based on 'saProtocol'."
    REFERENCE   "RFC 2406 Section 2.1"
    ::= { phase2SaEntry 3 }

saOutSpi OBJECT-TYPE
    SYNTAX      Unsigned32
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The security parameters index of the outbound SA of the
        inbound/outbound SA pair. If the protocol of the SA pair is
        IPcomp, this value is the CPI.

        This value is used with the value of 'suiteRemoteAddress'
        from the row indexed by 'suiteIndex' to create a SPI/address
        pair that uniquely identifies the outbound SA used in this
        SA suite. This can then be used to look up the SA in the
        appropriate outbound SA table, based on 'saProtocol'."
    REFERENCE   "RFC 2406 Section 2.1"
    ::= { phase2SaEntry 4 }


--
-- the Phase 2 Suite By Creators Table
--

suiteByCreatorsTable OBJECT-TYPE
    SYNTAX      SEQUENCE OF SuiteByCreatorsEntry
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "The (conceptual) table that sorts the SA suites by the
        endpoint identifiers.

        The number of rows in this table is the same as the number
        of suites in the entity."
    ::= { suiteTables 4 }

suiteByCreatorsEntry OBJECT-TYPE
    SYNTAX      SuiteByCreatorsEntry
    MAX-ACCESS  not-accessible
    STATUS      current
```

```
    DESCRIPTION
        "An entry (conceptual row) referencing a particular suite.

        A row in this table cannot be created or deleted by SNMP
        operations on columns of the table."        INDEX
        {
            suiteByCreatorsP1LocalEndpoint,
            suiteByCreatorsP1RemoteEndpoint,
            suiteByCreatorsIndex
        }
    ::= { suiteByCreatorsTable 1 }

SuiteByCreatorsEntry    ::= SEQUENCE {
 -- index
    suiteByCreatorsP1LocalEndpoint  Unsigned32,
    suiteByCreatorsP1RemoteEndpoint Unsigned32,
    suiteByCreatorsIndex            Unsigned32,

-- suite reference
    suiteByCreatorsRef              OBJECT IDENTIFIER
}

suiteByCreatorsP1LocalEndpoint OBJECT-TYPE
    SYNTAX      Unsigned32
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "The index of the endpoint table row for the local
        endpoint."
    ::= { suiteByCreatorsEntry 1 }

suiteByCreatorsP1RemoteEndpoint OBJECT-TYPE
    SYNTAX      Unsigned32
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "The index of the endpoint table row for the remote
        endpoint."
    ::= { suiteByCreatorsEntry 2 }

suiteByCreatorsIndex OBJECT-TYPE
    SYNTAX      Unsigned32 (1..16777215)
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "A unique value, greater than zero, for each SA suite that
        is between the two endpoints. It is recommended that values
        are assigned contiguously starting from 1 for each SA suite
        between the two endpoints.

        Note that duplicate entries for the saByCreatorsHash value
```

```
        may also arise due to hash result collisions."
    ::= { suiteByCreatorsEntry 3 }

suiteByCreatorsRef OBJECT-TYPE
    SYNTAX   OBJECT IDENTIFIER
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The object identifier of 'suiteIndex' in the row
        ('suiteEntry') of the 'suiteTable' to which this row
        refers."
    ::= { suiteByCreatorsEntry 4 }


--
-- the Phase 2 Suite By Selector Table
--

suiteBySelectorsTable OBJECT-TYPE
    SYNTAX      SEQUENCE OF SuiteBySelectorsEntry
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "The (conceptual) table that sorts the suites by the
        selectors.

        The number of rows in this table is the same as the number
        of suites in the entity.

        The maximum number of rows in this table is implementation
        dependent."
    ::= { suiteTables 5 }

suiteBySelectorsEntry OBJECT-TYPE
    SYNTAX      SuiteBySelectorsEntry
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "An entry (conceptual row) referencing a particular suite.

        A row in this table cannot be created or deleted by SNMP
        operations on columns of the table."
    INDEX
        {
            selectorIndex,
            suiteBySelectorsIndex
        }
    ::= { suiteBySelectorsTable 1 }
SuiteBySelectorsEntry  ::= SEQUENCE {
-- additional index
    suiteBySelectorsIndex       Unsigned32,
```

```
-- suite reference
    suiteBySelectorsRef         OBJECT IDENTIFIER
}

suiteBySelectorsIndex OBJECT-TYPE
    SYNTAX      Unsigned32 (1..16777215)
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "A unique value, greater than zero, for each SA suite that
        has the same selectors. It is recommended that values are
        assigned contiguously starting from 1."
    ::= { suiteBySelectorsEntry 1 }

suiteBySelectorsRef OBJECT-TYPE
    SYNTAX   OBJECT IDENTIFIER
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The object identifier of 'suiteIndex' in the row
        ('suiteEntry') of the 'suiteTable' to which this row
        refers."
    ::= { suiteBySelectorsEntry 2 }

--
-- the Phase 2 SA to Suite Table
--

ipsecSaInSuiteTable OBJECT-TYPE
    SYNTAX      SEQUENCE OF IpsecSaInSuiteEntry
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "The (conceptual) table that allows determination of which
        suite a particular phase 2 SA is in.

        The number of rows in this table is the same as the number
        of phase 2 SAs in the entity."
    ::= { suiteTables 6 }

ipsecSaInSuiteEntry OBJECT-TYPE
    SYNTAX      IpsecSaInSuiteEntry
    MAX-ACCESS  not-accessible
    STATUS      current        DESCRIPTION
        "An entry (conceptual row) referencing a particular phase 2
        SA.

        A row in this table cannot be created or deleted by SNMP
        operations on columns of the table."
    INDEX
```

```
        {
            ipsecSaInSuiteDestAddrType,
            ipsecSaInSuiteDestAddress,
            ipsecSaInSuiteProtocol,
            ipsecSaInSuiteSpi
        }
    ::= { ipsecSaInSuiteTable 1 }

IpsecSaInSuiteEntry ::= SEQUENCE {
-- index
    ipsecSaInSuiteDestAddrType  InetAddressType,
    ipsecSaInSuiteDestAddress   InetAddress,
    ipsecSaInSuiteProtocol      IpsecDoiSecProtocolId,
    ipsecSaInSuiteSpi           Unsigned32,

-- SA reference
    ipsecSaInSuiteRef           OBJECT IDENTIFIER
}

ipsecSaInSuiteDestAddrType OBJECT-TYPE
    SYNTAX      InetAddressType
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "The type of the destination address of the IPsec phase 2 SA
        to which this row refers."
    ::= { ipsecSaInSuiteEntry 1 }

ipsecSaInSuiteDestAddress OBJECT-TYPE
    SYNTAX      InetAddress (SIZE(4|16|20))
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "The destination address of the IPsec phase 2 SA to which
        this row refers."
    ::= { ipsecSaInSuiteEntry 2 }

ipsecSaInSuiteProtocol OBJECT-TYPE
    SYNTAX      IpsecDoiSecProtocolId
    MAX-ACCESS  not-accessible
    STATUS      current        DESCRIPTION
        "The security protocol of the IPsec phase 2 SA to which this
        row refers."
    ::= { ipsecSaInSuiteEntry 3 }

ipsecSaInSuiteSpi OBJECT-TYPE
    SYNTAX      Unsigned32
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "The SPI value of the IPsec phase 2 SA to which this row
```

```
        refers. If the value of 'ipsecSaInSuiteProtocol' is
        'protoIpcomp(4)', then this is the CPI of the SA."
    REFERENCE   "RFC 2407 Section 4.6.2.1"
    ::= { ipsecSaInSuiteEntry 4 }

ipsecSaInSuiteRef OBJECT-TYPE
    SYNTAX   OBJECT IDENTIFIER
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The object identifier of 'suiteIndex' in the row
        ('suiteEntry') of the 'suiteTable' to which this row refers.

        This is the suite that uses this SA."
    ::= { ipsecSaInSuiteEntry 5 }


-- the Notify Message MIB-Group
--
-- a collection of objects providing information about
-- the occurrences of notify messages

notifyCountTable OBJECT-TYPE
    SYNTAX      SEQUENCE OF NotifyCountEntry
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "The (conceptual) table containing information on IPSec
        notify message counts.

        Rows are created in this table for every notification type
        that has been sent or received by the entity.

        This table MAY be sparsely populated; that is, rows for
        which the count is 0 may be absent."
    ::= { ikeNotifications 1 }
notifyCountEntry OBJECT-TYPE
    SYNTAX      NotifyCountEntry
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "An entry (conceptual row) containing the total number of
        occurrences of a notify message.

        A row in this table cannot be created or deleted by SNMP
        operations on columns of the table."
    INDEX   { notifyProtocol, notifyType }
    ::= { notifyCountTable 1 }

NotifyCountEntry ::= SEQUENCE {
-- identification
```

```
    notifyProtocol          IpsecDoiSecProtocolId,
    notifyType              IkeNotifyMessageType,

-- ocurrences
    notifiesSent            Counter32,
    notifiesReceived        Counter32
}

notifyProtocol OBJECT-TYPE
    SYNTAX      IpsecDoiSecProtocolId
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "The value representing a protocol for which the notify was
        used."
    REFERENCE   "RFC 2408 Section 3.14"
    ::= { notifyCountEntry 1 }

notifyType OBJECT-TYPE
    SYNTAX      IkeNotifyMessageType
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "The value representing a specific ISAKMP notify message, or
        0 if unknown.

        Values are assigned from the set of notify message types as
        defined in Section 3.14.1 of [ISAKMP], and enhanced by the
        IPsec DOI. In addition, the value 0 may be used for this
        object when the object is used as a trap cause, and the
        cause is unknown."
    REFERENCE   "RFC 2408 Section 3.14.1"
    ::= { notifyCountEntry 2 }
notifiesSent    OBJECT-TYPE
    SYNTAX      Counter32
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The total number of times the specific notify message has
        been sent by the entity since system boot."
    ::= { notifyCountEntry 3 }

notifiesReceived    OBJECT-TYPE
    SYNTAX      Counter32
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The total number of times the specific notify message has
        been received by the entity since system boot."
    ::= { notifyCountEntry 4 }
```

```
--
-- the IKE Entity MIB-Group
--
-- a collection of objects providing information about overall IKE
-- status in the entity

--
-- IKE phase 1 SA statistics
--

ikeCurrentSAs OBJECT-TYPE
    SYNTAX      Gauge32
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The current number of IKE SAs in the entity."
    ::= { ikeGlobals 1 }

ikeCurrentInitiatedSAs OBJECT-TYPE
    SYNTAX      Gauge32
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The current number of IKE SAs successfully negotiated in
        the entity that were initiated by the entity."
    ::= { ikeGlobals 2 }

ikeCurrentRespondedSAs OBJECT-TYPE
    SYNTAX      Gauge32
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The current number of IKE SAs successfully negotiated in
        the entity that were initiated by the peer entity."
    ::= { ikeGlobals 3 }

ikeTotalSAs OBJECT-TYPE
    SYNTAX      Counter32
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The total number of IKE SAs successfully negotiated in the
        entity since boot time."
    ::= { ikeGlobals 4 }

ikeTotalInitiatedSAs OBJECT-TYPE
    SYNTAX      Counter32
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
```

```
        "The total number of IKE SAs successfully negotiated in the
        entity since boot time that were initiated by the entity."
    ::= { ikeGlobals 5 }

ikeTotalRespondedSAs OBJECT-TYPE
    SYNTAX      Counter32
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The total number of IKE SAs successfully negotiated in the
        entity since boot time that were initiated by the peer
        entity."
    ::= { ikeGlobals 6 }

ikeTotalAttempts OBJECT-TYPE
    SYNTAX      Counter32
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The total number of IKE SAs negotiation attempts made since
        boot time. This includes successful negotiations."
    ::= { ikeGlobals 7 }

ikeTotalSaInitAttempts OBJECT-TYPE
    SYNTAX      Counter32        MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The total number of IKE SAs negotiation attempts made where
        the entity was the initiator since boot time. This includes
        successful negotiations."
    ::= { ikeGlobals 8 }

ikeTotalSaRespAttempts OBJECT-TYPE
    SYNTAX      Counter32
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The total number of IKE SAs negotiation attempts made where
        the entity was the responder since boot time. This includes
        successful negotiations."
    ::= { ikeGlobals 9 }


--
-- IKE Aggregate Traffic Statistics
--

ikeTotalInPackets OBJECT-TYPE
    SYNTAX      Counter32
    UNITS       "packets"
    MAX-ACCESS  read-only
```

```
    STATUS      current
    DESCRIPTION
        "The total number of IKE packets received by the entity
        since boot time, including re-transmissions and un-encrypted
        packets."
    ::= { ikeTrafStats 1 }

ikeTotalOutPackets OBJECT-TYPE
    SYNTAX      Counter32
    UNITS       "packets"
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The total number of IKE packets sent by the entity since
        boot time, including re-transmissions and un-encrypted
        packets."
    ::= { ikeTrafStats 2 }

ikeTotalInOctets OBJECT-TYPE
    SYNTAX      Counter64
    UNITS       "bytes"        MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The total amount of IKE traffic received by the entity
        since boot time, measured in bytes, including any re-
        transmitted packets received, and including encrypted and
        un-encrypted packets."
    ::= { ikeTrafStats 3 }

ikeTotalOutOctets OBJECT-TYPE
    SYNTAX      Counter64
    UNITS       "bytes"
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The total amount of IKE traffic sent by the entity since
        boot time, measured in bytes, including any re-transmissions
        and including encrypted and un-encrypted packets."
    ::= { ikeTrafStats 4 }


--
-- IKE Phase 1 SA Aggregate Errors
--

ikeTotalInitFailures OBJECT-TYPE
    SYNTAX      Counter32
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The total number of attempts to initiate an IKE phase 1 SA
```

```
        that failed since boot time, when there was a response from
        the peer entity.

        This value may be used to detect clogging or denial-of-
        service attacks."
    ::= { ikeErrors 1 }

ikeTotalInitNoResponses OBJECT-TYPE
    SYNTAX      Counter32
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The total number of attempts to initiate an IKE phase 1 SA
        that failed since boot time, when there was no response from
        the peer entity.

        This should only be incremented if the peer does not repond
        to the first packet of attempted negotiations."
    ::= { ikeErrors 2 }

ikeTotalRespFailures OBJECT-TYPE
    SYNTAX      Counter32
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The total number of attempts to initiate an IKE phase 1 SA
        that failed since boot time, when the initiation attempt
        came for the peer entity."
    ::= { ikeErrors 3 }


--
-- Suite Global Objects
--

totalSuites OBJECT-TYPE
    SYNTAX      Counter32
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The total number of suites created by the entity since
        system boot."
    ::= { suiteGlobals 1 }

currentSuites OBJECT-TYPE
    SYNTAX      Gauge32
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The total number of suites currently in existence in the
        entity."
```

```
    ::= { suiteGlobals 2 }

--
-- Suite Aggregate Traffic Statistics
--

suiteTotalInUserKbytes OBJECT-TYPE
    SYNTAX      Counter64        UNITS       "Kilobytes"
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The total amount of user level traffic carried by all
        suites in the entity since boot time, measured in Kilobytes
        (1024 bytes), in the inbound direction.

        This is the sum of the 'suiteInUserOctets' column for all
        suite rows created since boot time."
    ::= { suiteTrafStats 1 }

suiteTotalInPackets OBJECT-TYPE
    SYNTAX      Counter64
    UNITS       "packets"
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The total number of packets carried by all suites in the
        entity since boot time in the inbound direction.

        This is the sum of the 'suiteInPackets' column for all suite
        rows created since boot time."
    ::= { suiteTrafStats 2 }

suiteTotalOutUserKbytes OBJECT-TYPE
    SYNTAX      Counter64
    UNITS       "Kilobytes"
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The total amount of user level traffic carried by all
        suites in the entity since boot time, measured in Kilobytes
        (1024 bytes), in the outbound direction.

        This is the sum of the 'suiteOutUserOctets' column for all
        suite rows created since boot time."
    ::= { suiteTrafStats 3 }

suiteTotalOutPackets OBJECT-TYPE
    SYNTAX      Counter64
    UNITS       "packets"
    MAX-ACCESS  read-only
    STATUS      current
```

```
    DESCRIPTION
        "The total number of packets carried by all suites in the
        entity since boot time, in the outbound direction.
        This is the sum of the 'suiteOutPackets' column for all
        suite rows created since boot time."
    ::= { suiteTrafStats 4 }

--
-- Suite Aggregate Error Counts
--

suiteInitFailures OBJECT-TYPE
    SYNTAX      Counter32
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The total number of attempts to initiate an suite that
        failed since boot time, when the attempt was initiated
        locally."
    ::= { suiteErrors 1 }

suiteRespondFailures OBJECT-TYPE
    SYNTAX      Counter32
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The total number of attempts to initiate an suite that
        failed since boot time, when the attempt was initiated by
        the peer entity."
    ::= { suiteErrors 2 }


--
-- Trap Objects, Traps and Trap Control
--

ikeLocalEndpoint OBJECT-TYPE
    SYNTAX      Unsigned32
    MAX-ACCESS  accessible-for-notify
    STATUS      current
    DESCRIPTION
        "The index to an endpoint that is the local endpoint in a
        trap."
    ::= { ikeTrapObjects 1 }

ikeRemoteEndpoint OBJECT-TYPE
    SYNTAX      Unsigned32
    MAX-ACCESS  accessible-for-notify
    STATUS      current

    DESCRIPTION
```

```
        "The index to an endpoint that is the remote endpoint in a
        trap."
    ::= { ikeTrapObjects 2 }

ikeSelector OBJECT-TYPE
    SYNTAX      Unsigned32
    MAX-ACCESS  accessible-for-notify
    STATUS      current
    DESCRIPTION
        "The index to a selector that is involved in a trap."
    ::= { ikeTrapObjects 3 }

ikeAuthMethod OBJECT-TYPE
    SYNTAX      IkeAuthMethod
    MAX-ACCESS  accessible-for-notify
    STATUS      current
    DESCRIPTION
        "An authentication method that was used in a trap."
    ::= { ikeTrapObjects 4 }

ikeNegFailureTrapEnable OBJECT-TYPE
    SYNTAX      TruthValue
    MAX-ACCESS  read-write
    STATUS      current
    DESCRIPTION
        "Indicates whether ikeNegFailure traps should be generated."
    DEFVAL { false }
    ::= { ikeTrapControl 1 }

ikeNegFailure NOTIFICATION-TYPE
 OBJECTS   {
        ikeLocalEndpoint,
        ikeRemoteEndpoint,
        localIpAddressType,
        localIpAddress,
        localUdpPort,
        remoteIpAddressType,
        remoteIpAddress,
        remoteUdpPort,
        ikeAuthMethod,
        ikeTotalInitFailures,
        ikeTotalInitNoResponses,
        ikeTotalRespFailures,
        notifiesSent,
        notifiesReceived
    }
    STATUS  current        DESCRIPTION
        "An attempt to negotiate a phase 1 IKE SA failed.

        The notification counts are also sent as part of the trap,
        along with the current value of the total negotiation error
```

```
        counters for ISAKMP."
    ::= { ikeTraps 0 1 }

suiteNegFailureTrapEnable OBJECT-TYPE
    SYNTAX      TruthValue
    MAX-ACCESS  read-write
    STATUS      current
    DESCRIPTION
        "Indicates whether 'suiteNegFailure' traps should be
        generated."
    DEFVAL { false }
    ::= { suiteTrapControl 1 }

suiteNegFailure NOTIFICATION-TYPE
 OBJECTS   {
        ikeSelector,
        suiteInitFailures,
        suiteRespondFailures,
        notifiesSent,
        notifiesReceived
    }
    STATUS  current
    DESCRIPTION
        "An attempt to negotiate a phase 2 SA suite for the
        specified selector failed.

        The current total failure counts are passed as well as the
        notification type counts for the notify involved in the
        failure."
    ::= { suiteTraps 0 1 }


--
-- Units of conformance (Object Groups)
--

--
-- Authors' note: Index objects are commented out, since the current
-- SMI does not allow objects with a MAX-ACCESS clause of
-- 'not-accessible' to be put in groups.
--

oakleyGroup  OBJECT-GROUP
 OBJECTS
    {
        -- modpGroupIndex,
        modpFieldSize, modpPrime, modpGenerator, modpLPF,
        modpStrength,
        -- ecpGroupIndex,
        ecpFieldSize, ecpPrime, ecpGeneratorOne, ecpGeneratorTwo,
        ecpParameterOne, ecpParameterTwo, ecpLPF, ecpOrder,
```

```
        ecpStrength,
        -- ec2nGroupIndex,
        ec2nDegree, ec2nIrrPoly, ec2nGeneratorOne, ec2nGeneratorTwo,
        ec2nParameterOne, ec2nParameterTwo, ec2nLPF, ec2nOrder,
        ec2nStrength
    }
    STATUS  current
    DESCRIPTION
        "A collection of objects that describe the Oakley Groups
        used or known by the entity."
    REFERENCE   "RFC 2412"
    ::= { ikeGroups 1 }

endpointGroup OBJECT-GROUP
 OBJECTS
    {
        -- endpointIndex,
        endpointIdType, endpointIdValue, endpointCertSerialNum,
        endpointCertIssuer, endpointIsLocal, endpointCurrentIkeSAs,
        endpointTotalIkeSAs, endpointCurrentSuites,
        endpointTotalSuites
    }
    STATUS  current
    DESCRIPTION
        "A collection of objects that describe IKE endpoints."
    ::= { ikeGroups 2 }

ikeSaGroup OBJECT-GROUP
 OBJECTS
    {
        saAuthMethod, saPeerEndpoint, saLocalEndpoint, saEncAlg,
        saEncKeyLength, saHashAlg, saHashKeyLength, saPRF,
        saOakleyGroupDesc, saOakleyGroup, saLimitSeconds,
        saLimitKbytes, saLimitKeyUses, saAccKbytes, saKeyUses,
        saCreatedSuites, saDeletedSuites, saDecryptErrors,
        saHashErrors, saOtherReceiveErrors, saSendErrors
    }
    STATUS  current
    DESCRIPTION
        "A collection of objects that describe IKE phase 1 SAs."
    ::= { ikeGroups 3 }

ikeHelpersGroup OBJECT-GROUP
 OBJECTS
    {
        -- saByCreatorsLocalEndpoint, saByCreatorsRemoteEndpoint,
        -- saByCreatorsIndex,
        saIkeLocalIpAddressType, saIkeLocalIpAddress,
        saIkeRemoteIpAddressType, saIkeRemoteIpAddress,
        saIkeInitiatorCookie, saIkeResponderCookie
    }
```

```
    STATUS  current
    DESCRIPTION
        "A collection of objects that help look up IKE phase 1 SAs."
    ::= { ikeGroups 4 }

exchangeGroup OBJECT-GROUP
 OBJECTS
    {
        -- exchangeType,
        exchangesTotalCount, exchangesInitiatedCount,
        exchangesRespondedCount
    }
    STATUS  current
    DESCRIPTION
        "A collection of objects that count exchanges."
    ::= { ikeGroups 5 }

suiteGroup OBJECT-GROUP
 OBJECTS
    {
        -- suiteIndex,
        suiteLocalAddressType, suiteLocalAddress,
        suiteRemoteAddressType, suiteRemoteAddress,
        suitePhase1RemoteEndpoint, suitePhase1LocalEndpoint,
        suiteSelector, suiteOakleyGroupDesc, suiteOakleyGroup,
        suiteLifeSeconds, suiteInUserOctets, suiteInPackets,
        suiteOutUserOctets, suiteOutPackets, suiteSendErrors,
        suiteReceiveErrors
    }
    STATUS  current
    DESCRIPTION
        "A collection of objects that describe phase 2 SA suites."
    ::= { ikeGroups 7 }

phase2SaGroup OBJECT-GROUP
 OBJECTS
    {
        -- saOrder,
        saProtocol, saInSpi, saOutSpi,
        -- ipsecSaInSuiteDestAddrType, ipsecSaInSuiteDestAddress,
        -- ipsecSaInSuiteProtocol, ipsecSaInSuiteSpi,
        ipsecSaInSuiteRef
    }
    STATUS  current
    DESCRIPTION
        "A collection of objects that relate phase 2 SAs to phase 2
        SA suites."
    ::= { ikeGroups 8 }

suiteHelperGroup OBJECT-GROUP
 OBJECTS
```

```
    {
        -- suiteByCreatorsP1LocalEndpoint,
        -- suiteByCreatorsP1RemoteEndpoint, suiteByCreatorsIndex,
        suiteByCreatorsRef,
        -- suiteBySelectorsIndex,
        suiteBySelectorsRef
    }
    STATUS  current
    DESCRIPTION
        "A collection of objects that help look up phase 2 SA
        suites."
    ::= { ikeGroups 9 }

notifyGroup OBJECT-GROUP
 OBJECTS
    {
        -- notifyProtocol, notifyType,
        notifiesSent, notifiesReceived
    }
    STATUS  current
    DESCRIPTION
        "A collection of objects that take statistics for notify
        messages in IKE."
    ::= { ikeGroups 10 }

ikeGlobalsGroup  OBJECT-GROUP
 OBJECTS
    {
        ikeCurrentSAs, ikeCurrentInitiatedSAs,
        ikeCurrentRespondedSAs, ikeTotalSAs, ikeTotalInitiatedSAs,
        ikeTotalRespondedSAs, ikeTotalAttempts,
        ikeTotalSaInitAttempts, ikeTotalSaRespAttempts,
        ikeTotalInPackets, ikeTotalOutPackets, ikeTotalInOctets,
        ikeTotalOutOctets, ikeTotalInitFailures,
        ikeTotalInitNoResponses, ikeTotalRespFailures
    }
    STATUS  current
    DESCRIPTION
        "A collection of objects providing global IKE phase 1 SA
        statistics."
    ::= { ikeGroups 11 }

suiteGlobalsGroup OBJECT-GROUP
 OBJECTS
    {
        totalSuites, currentSuites, suiteTotalInUserKbytes,
        suiteTotalInPackets, suiteTotalOutUserKbytes,
        suiteTotalOutPackets, suiteInitFailures,
        suiteRespondFailures
    }
    STATUS  current
```

```
    DESCRIPTION
        "A collection of objects providing global phase 2 SA suite
        statistics."
    ::= { ikeGroups 12 }

ikeTrapArgumentGroup OBJECT-GROUP
 OBJECTS
    {
        ikeLocalEndpoint, ikeRemoteEndpoint, ikeSelector,
        ikeAuthMethod
    }
    STATUS  current
    DESCRIPTION
        "A collection of objects used only as arguments in traps."
    ::= { ikeGroups 13 }

ikeTrapEnableGroup OBJECT-GROUP
 OBJECTS
    {
        ikeNegFailureTrapEnable, suiteNegFailureTrapEnable
    }
    STATUS  current
    DESCRIPTION
        "A collection of objects providing control over trap
        generation."
    ::= { ikeGroups 14 }

ikeTrapGroup    NOTIFICATION-GROUP
    NOTIFICATIONS
    {
        ikeNegFailure, suiteNegFailure
    }
    STATUS  current
    DESCRIPTION
        "A collection of traps."
    ::= { ikeGroups 15 }

--
-- Compliance statements
--

ikeMonitorCompliance    MODULE-COMPLIANCE
    STATUS      current
    DESCRIPTION
        "The compliance statement for SNMPv2 entities which
        implement the IKE Monitoring MIB."
    MODULE      -- this module
    MANDATORY-GROUPS
        {
            endpointGroup, ikeSaGroup, ikeHelpersGroup,
            exchangeGroup, suiteGroup, phase2SaGroup,
```

```
            suiteHelperGroup, notifyGroup, ikeGlobalsGroup,
            suiteGlobalsGroup, ikeTrapArgumentGroup,
            ikeTrapEnableGroup, ikeTrapGroup
        }

    -- Allow the trap controls to be read-only

    OBJECT  ikeNegFailureTrapEnable
    MIN-ACCESS  read-only
    DESCRIPTION
        "If an implementation cannot properly secure this variable
        against unauthorized write access, it SHOULD implement it as
        read-only, to prevent the security risk of enabling the
        traps. Of course, there must be other means of controlling
        the generation of the associated trap."

    OBJECT  suiteNegFailureTrapEnable
    MIN-ACCESS  read-only
    DESCRIPTION
        "If an implementation cannot properly secure this variable
        against unauthorized write access, it SHOULD implement it as
        read-only, to prevent the security risk of enabling the
        traps. Of course, there must be other means of controlling
        the generation of the associated trap."
      -- don't require support for dns(16) address type

    -- Authors' note: The following statements are commented out,
    -- since the current SMI does not allow objects with a
    -- MAX-ACCESS clause of not-accessible to be put in groups,
    -- and objects that are not in groups cannot be in
    -- compliance statements.

-- OBJECT  saIkeLocalIpAddressType
-- SYNTAX INTEGER { ipv4(1), ipv6(2) }
-- DESCRIPTION
--     "An implementation is only required to support IPv4 and IPv6
--      addresses."

-- OBJECT  saIkeRemoteIpAddressType
-- SYNTAX INTEGER { ipv4(1), ipv6(2) }
-- DESCRIPTION
--     "An implementation is only required to support IPv4 and IPv6
--      addresses."

-- OBJECT  suiteLocalAddressType
-- SYNTAX INTEGER { ipv4(1), ipv6(2) }
-- DESCRIPTION
--     "An implementation is only required to support IPv4 and IPv6
--      addresses."

-- OBJECT  suiteRemoteAddressType
```

```
    -- SYNTAX INTEGER { ipv4(1), ipv6(2) }
    -- DESCRIPTION
    --     "An implementation is only required to support IPv4 and IPv6
    --      addresses."

    -- OBJECT  ipsecSaInSuiteDestAddrType
    -- SYNTAX INTEGER { ipv4(1), ipv6(2) }
    -- DESCRIPTION
    --     "An implementation is only required to support IPv4 and IPv6
    --      addresses."

       ::= { ikeConformance 1 }

   END
```

6. Security Considerations

This MIB contains readable objects whose values provide information related to IPsec SAs. While some of the information is readily available by monitoring the traffic into an entity, other information may provide attackers with more information than an administrator may desire.

Some of the specific concerns are related to the display of the algorithms and key lengths associated with encryption, and the feedback of error counters and traps that enable an attacker to quickly determine the effect of his or her attacks.

Specific examples of this include, but are not limited to:

- Replay counts that tell attackers that replay values are being checked, and what the current window is.
- Specific algorithms and key lengths are displayed, giving attackers a better idea of how to attack.
- Specific traffic counts, giving attackers more information for traffic analysis.

Of particular concern is the ability to disable the transmission of traps. The traps defined in this MIB may appear due to badly configured systems and transient error conditions, but they may also appear due to attacks. If an attacker can disable these traps, they reduce some of the warnings that may be provided to system administrators.

It is thus important to control even GET access to these objects and possibly to even encrypt the values of these object when sending them over the network via SNMP. Not all versions of SNMP provide features for such a secure environment.

SNMPv1 by itself is not a secure environment. Even if the network itself is secure (for example by using IPsec), even then, there is no control as to who on the secure network is allowed to access and GET/SET (read/change/create/delete) the objects in this MIB.

It is recommended that the implementers consider the security features as provided by the SNMPv3 framework. Specifically, the use of the User-based Security Model RFC 2574 [RFC2574] and the View-based Access Control Model RFC 2575 [RFC2575] is recommended.

It is then a customer/user responsibility to ensure that the SNMP entity giving access to an instance of this MIB, is properly configured to give access to the objects only to those principals (users) that have legitimate rights to indeed GET or SET (change/create/delete) them.

7. Acknowledgments

This document was begun and mostly developed by Tim Jenkins and John Shriver. The editor listed for this document (Paul Hoffman) only sheparded the last steps before final publication.

This document is based in part on an earlier proposal titled "draft-ietf-ipsec-mib-xx.txt". That series was abandoned, since it included application specific constructs in addition to the IPsec only objects.

Portions of the original document's origins were based on the working paper "IP Security Management Information Base" by R. Thayer and U. Blumenthal.

Significant contribution to the IPsec MIB series of documents comes from Charles Brooks and Carl Powell, both of GTE Internetworking. Obviously, the IPsec working group made signification contributions, including M. Daniele, T. Kivinen, J. Walker, S. Kelly, J. Leonard, S. Waters, M. Richardson, M. Zallocco and M. Shelor. Thanks also to J. Schoenwaelder and M. Baugher for comments related to indexing of the tables.

8. References

8.1 Normative references

[ADDRMIB] Daniele, M., Haberman, B., Routhier, S., Schoenwaelder, J., "Textual Conventions for Internet Network Addresses", RFC 2851, June, 2000

[IDIMIB]Jenkins, T., Shriver, J., "ISAKMP DOI-Independent Monitoring MIB, draft-ietf-ipsec-isakmp-di-mon-mib, work in progress

[IMMIB] Jenkins, T., Shriver, J., "IPsec Monitoring MIB, draft-ietf-ipsec-monitor-mib, work in progress

[IPSECTC] Shriver, J., "IPSec DOI Textual Conventions MIB, draft-ietf-ipsec-doi-tc-mib, work in progress

[RFC2571] Harrington, D., Presuhn, R., and B. Wijnen, "An Architecture for Describing SNMP Management Frameworks", RFC 2571, April 1999

[RFC1155] Rose, M., and K. McCloghrie, "Structure and Identification of Management Information for TCP/IP-based Internets", STD 16, RFC 1155, May 1990

[RFC1212] Rose, M., and K. McCloghrie, "Concise MIB Definitions", STD 16, RFC 1212, March 1991

[RFC1215] M. Rose, "A Convention for Defining Traps for use with the SNMP", RFC 1215, March 1991

[RFC2578] McCloghrie, K., Perkins, D., Schoenwaelder, J., Case, J., Rose, M., and S. Waldbusser, "Structure of Management Information Version 2 (SMIv2)", STD 58, RFC 2578, April 1999

[RFC2579] McCloghrie, K., Perkins, D., Schoenwaelder, J., Case, J., Rose, M., and S. Waldbusser, "Textual Conventions for SMIv2", STD 58, RFC 2579, April 1999

[RFC2580] McCloghrie, K., Perkins, D., Schoenwaelder, J., Case, J., Rose, M., and S. Waldbusser, "Conformance Statements for SMIv2", STD 58, RFC 2580, April 1999

[RFC1157] Case, J., Fedor, M., Schoffstall, M., and J. Davin, "Simple Network Management Protocol", STD 15, RFC 1157, May 1990.

[RFC1901] Case, J., McCloghrie, K., Rose, M., and S. Waldbusser, "Introduction to Community-based SNMPv2", RFC 1901, January 1996.

[RFC1906] Case, J., McCloghrie, K., Rose, M., and S. Waldbusser, "Transport Mappings for Version 2 of the Simple Network Management Protocol (SNMPv2)", RFC 1906, January 1996.

[RFC2572] Case, J., Harrington D., Presuhn R., and B. Wijnen, "Message Processing and Dispatching for the Simple Network Management Protocol (SNMP)", RFC 2572, April 1999

[RFC2574] Blumenthal, U., and B. Wijnen, "User-based Security Model (USM) for version 3 of the Simple Network Management Protocol (SNMPv3)", RFC 2574, April 1999

[RFC1905] Case, J., McCloghrie, K., Rose, M., and S. Waldbusser, "Protocol Operations for Version 2 of the Simple Network Management Protocol (SNMPv2)", RFC 1905, January 1996.

[RFC2573] Levi, D., Meyer, P., and B. Stewart, "SNMPv3 Applications", RFC 2573, April 1999

[RFC2575] Wijnen, B., Presuhn, R., and K. McCloghrie, "View-based Access Control Model (VACM) for the Simple Network Management Protocol (SNMP)", RFC 2575, April 1999

[RFC2570] Case, J., Mundy, R., Partain, D., and B. Stewart, "Introduction to Version 3 of the Internet-standard Network Management Framework", RFC 2570, April 1999

8.2 Non-normative references

[IKE] Harkins, D., Carrel, D., "The Internet Key Exchange (IKE)", RFC2409, November 1998

[IPCOMP] Shacham, A., Monsour, R., Pereira, R., Thomas, M., "IP Payload Compression Protocol (IPcomp)", RFC3173, September 2001

[IPDOI] Piper, D., "The Internet IP Security Domain of Interpretation for ISAKMP", RFC2407, November 1998

[ISAKMP] Maughan, D., Schertler, M., Schneider, M., and Turner, J., "Internet Security Association and Key Management Protocol (ISAKMP)", RFC2408, November 1998

[OAKLEY] Orman, H., "The OAKLEY Key Determination Protocol", RFC2412, November 1998

[SECARCH] Kent, S., Atkinson, R., "Security Architecture for the Internet Protocol", RFC2401, November 1998

A. Changes from -03 to -04

[[To be removed when published as an RFC]]

- Changed the authors' names to the editor's name.

- Added acknowledgement for the original authors.

- Minor formatting changes.

- Split the references into normative and non-normative.

NOTE: There are still lines that talk about things that need to be changed before release of the RFC (search for "release").