Network Working Group   M. Fine
Internet Draft   K. McCloghrie
Expires January 2001   Cisco Systems
                                          J. Seligson
                                          K. Chan
                                          Nortel Networks
                                          S. Hahn
                                          R. Sahita
                                          Intel
                                          A. Smith
                                          No Affiliation
                                          Francis Reichmeyer
                                          IPHighway

                                          July 14, 2000

Framework Policy Information Base

draft-ietf-rap-frameworkpib-01.txt

Status of this Memo

1. Glossary

```
PRC     Policy Rule Class.        A type of policy data.
PRI     Policy Rule Instance.     An instance of a PRC.
PIB     Policy Information Base.  The database of policy information.
PDP     Policy Decision Point.    See [RAP-FRAMEWORK].
PEP     Policy Enforcement Point. See [RAP-FRAMEWORK].
PRID    Policy Rule Instance Identifier.  Uniquely identifies an
        instance of a PRC.
```

2. Introduction

[SPPI] describes a structure for specifying policy information that
can then be transmitted to a network device for the purpose of
configuring policy at that device. The model underlying this
structure is one of well-defined policy rule classes and instances
of these classes residing in a virtual information store called the
Policy Information Base (PIB).

One way to provision policy is by means of the COPS protocol [COPS]
with the extensions for provisioning [COPS-PR]. This protocol
supports multiple clients, each of which may provision policy for a
specific policy domain such as QoS, virtual private networks, or
security.

As described in [COPS-PR], each client supports a non-overlapping
and independent set of PIB modules. However, some policy rule
classes are common to all subject categories (client-types) and need
to be present in each. This document presents a set of PRCs that
are common to all clients that provision policy using COPS for
Provisioning.

3. General PIB Concepts

3.1. Roles

The policy to apply to an interface may depend on many factors such
as immutable characteristics of the interface (e.g., ethernet or
frame relay), the status of the interface (e.g., half or full
duplex), or user configuration (e.g., branch office or headquarters
interface). Rather than specifying policies explicitly for each
interface of all devices in the network, policies are specified in
terms of interface functionality.

To describe these functionalities of an interface we use the concept
of "roles". A role is simply a string that is associated with an
interface. A given interface may have any number of roles
simultaneously. Policy rule classes have an attribute called a
"role-combination" which is a lexicographically ordered set of

roles. Instances of a given policy rule class are applied to an
interface if and only if the set of roles in the role combination
matches the set of the roles of the interface.

Thus, roles provide a way to bind policy to interfaces without
having to explicitly identify interfaces in a consistent manner
across all network devices. (The SNMP experience with ifIndex has
proved this to be a difficult task.) That is, roles provide a level
of indirection to the application of a set of policies to specific
interfaces. Furthermore, if the same policy is being applied to
several interfaces, that policy need be pushed to the device only
once, rather than once per interface, as long as the interfaces are
configured with the same role combination.

We point out that, in the event that the administrator needs to have
unique policy for each interface, this can be achieved by
configuring each interface with a unique role.

The PEP reports all its role combinations to the PDP in the initial
COPS request (REQ) message and in subsequent request messages
generated in response to COPS state synchronization (SSQ) requests
and local configuration changes.

The comparing of roles (or role combinations) is case sensitive.

By convention, when formatting the role-combination for exchange
within a protocol message, within a PIB/MIB object's value, or as a
printed value, the set is formatted in lexicographical order of the
role's ASCII values; that is, the role that is first is formatted
first. For example, "a+b" and "b+a" are NOT different role-
combinations; rather, they are different formatting of the same
role-combination, and hence for this example:
- "a+b" is the valid formatting of that role-combination,
- "b+a" is an invalid formatting of that role-combination.

The role-combination of interfaces to which no roles have been
assigned is known as the "null" role-combination. (Note the
deliberate use of lower-case letters for "null" so that it avoids
confusion with the ASCII NULL character that has a value of zero but
a length of one.)

In an "install" or an "install-notify" class, the wildcard role-
combination "*" can be used. In addition to providing for interface-
specific roles, it also allows for other optimizations in reducing
the number of role-combinations for which a policy has to be
specified. For example:

Suppose we have three interfaces:

   Roles A, B and R1 are assigned to interface I1
   Roles A, B and R2 are assigned to interface I2
   Roles A, B and R3 are assigned to interface I3

Then, a PRI of the qosIfDscpAssignTable class which has the values:

```
qosIfDscpAssignPrid    = 1
qosIfDscpAssignRoles   = "*+A+B"
qosIfDscpAssignName    = "4queues"
qosIfDscpAssignDscpMap = 1
```

will apply to all three interfaces, because "*" matches with R1, R2 and R3.

Formally,
- The wildcard role is denoted by "*",
- The "*" role is not allowed to be defined as part of the role-combination of an interface as notified by the PEP to the PDP; it is only allowed in policies installed/deleted via COPS-PR from the PDP to the PEP.
- For a policy to apply to an interface when the policy's role-combination is "*+a+b", then the interface's role-combination:
        - Must include "a" and "b", and
        - Can include zero or more other roles.
- The wildcard character "*" is listed before the other roles as "*" is lexicographically before "a"; however, the wildcard matches any zero or more roles, irrespective of lexicographical order. For example: "*+b+e+g" would match "a+b+c+e+f+g"

The concept and usage of roles in this document is consistent with that specified in [POLICY]. Roles are currently under discussion in the IETF's Policy WG; as and when that discussion reaches a conclusion, this PIB will be updated in accordance with that conclusion.

3.1.1. An Example

The functioning of roles might be best understood by an example. Suppose I have a device with three interfaces, with roles as follows:

```
IF1: "finance"
IF2: "finance"
IF3: "manager"
```

Suppose, I also have a PDP with two policies:

```
P1: Packets from finance department (role "finance") get DSCP 5
P2: Packets from managers (role "manager") get DSCP 6
```

To obtain policy, the PEP reports to the PDP that it has some interfaces with role combination "finance" and some with role combination "manager". In response, the PDP downloads policy P1

associated with role combination "finance" and downloads a second
policy P2 associated with role combination "manager".

Now suppose the finance person attached to IF2 is promoted to manager and so the system administrator adds the role "manager" to IF2. The PEP now reports to the PDP that it has three role combinations: some interfaces with role combination "finance", some with role combination "manager" and some with role combination "finance+manager". In response, the PDP downloads an additional third policy associated with the new role combination "finance+manager".

How the PDP determines the policy for this new role combination is entirely the responsibility of the PDP. It could do so algorithmically or by rule. For example, there might be a rule that specifies that manager policy takes preference over department policy. Or there might be a third policy installed in the PDP as follows:

        P3: Packets from finance managers (role "finance" and role
            "manager") get DSCP 7

The point here is that the PDP is required to determine what policy applies to this new role combination and to download a third policy to the PEP for the role combination "finance+manager" even if that policy is the same as one already downloaded. The PEP is not required (or allowed) to construct policy for new role combinations from existing policy.

3.2. Multiple PIB Instances

[COPS-PR] supports multiple, disjoint, independent instances of the PIB to represent multiple instances of configured policy. The intent is to allow for the pre-provisioning of policy that can then be made active by a single, short decision from the PDP.

A COPS context can be defined as an independent COPS request state for a particular subject category (client-type).

With the COPS-PR protocol, each of these states are identified by a unique client handle. The creation and deletion of these PIB instances is controlled by the PDP as described in [COPS-PR].

Although many PIB instances may be configured on a device (the maximum number of these instances being determined by the device itself) only one of them can be active at any given time, the active one being selected by the PDP. To facilitate this selection, the Framework PIB supports an attribute to make a PIB instance the active one and, similarly, to report the active PIB instance to the PDP in a COPS request message. This attribute is in the Incarnation Table described below.

Setting the attribute FrwkPibIncarnationActive to 'true' in one PIB instance MUST ensure that the attribute is 'false' in all other contexts.

3.3. Reporting of Device Capabilities

   Each network device providing policy-based services has its own
   inherent capabilities. These capabilities can be hardware specific,
   e.g., an ethernet interface supporting input classification, or can
   be statically configured, e.g., supported queuing disciplines.
   These capabilities are communicated to the PDP when initial policy
   is requested by the PEP. Knowing device capabilities, the PDP can
   send the policy rule instances (PRIs) relevant to the specific
   device, rather than sending the entire PIB.

   The PIB indicates which capabilities the PEP must report to the PDP
   by means of the PIB-ACCESS clause as described in [SPPI].

3.4. Reporting of Device Limitations

   To facilitate efficient policy installation, it is important to
   understand a device's limitations in relation to the advertised
   device capabilities. Limitations may be class-based, e.g., an
   "install" class is supported as a "notify" or only a limited number
   of class instances may be created, or attribute-based. Attribute
   limitations, such as supporting a restricted set of enumerations or
   requiring related attributes to have certain values, detail
   implementation limitations at a fine level of granularity.

   A PDP can avoid certain installation issues in a proactive fashion
   by taking into account a device's limitations prior to policy
   installation rather than in a reactive mode during installation. As
   with device capabilities, device limitations are communicated to the
   PDP when initial policy is requested.

   Reported device limitations may be accompanied by guidance values
   that can be used by a PDP to determine acceptable values for the
   identified attributes.

4. Summary of the Framework PIB

   The Framework PIB comprises of three groups:

   1. Base PIB classes Group

      This contains PRCs intended to describe the classes supported
      by the PEP, limitations and its current configuration.

      PRC Support Table

         As the technology evolves, we expect devices to be enhanced
         with new PIBs, existing PIBs to add new PRCs and existing PRCs
         to be augmented or extended with new attributes. Also, it is

likely that some existing PRCs or individual attributes of PRCs
will be deprecated. The PRC Support Table describes the PRCs
that the device supports as well as the individual attributes
of each PRC. Using this information the PDP can potentially

   tailor the policy to more closely match the capabilities of the
   device. The PRC Support Table instances are specific to the
   particular Subject Category (Client-Type). That is, the PRC
   Support Table for Subject Category 'A' will not include
   instances for classes supported by the Subject Category 'B'.

PIB Incarnation Table

   This table contains exactly one row (corresponding to one PRI)
   per context. It identifies the PDP that was the last to
   download policy into the device and also contains an identifier
   to identify the version of the policy currently downloaded.
   This identifier, both its syntax and value, is meaningful only
   to the PDPs. It is intended to be a mechanism whereby a PDP,
   on connecting to a PEP, can easily identify a known incarnation
   of policy. The incarnation PRC also includes an attribute to
   indicate which context is the active one at the present time.

Attribute Limitations Table

   Some devices may not be able to implement the full range of
   values for all attributes. In principle, each PRC supports a
   set of errors that the PEP can report to the PDP in the event
   that the specified policy is not implementable. There are two
   problems with this: it may be preferable for the PDP to be
   informed of the device limitations before actually attempting
   to install policy, and while the error can indicate that a
   particular attribute value is unacceptable to the PEP, this
   does not help the PDP ascertain which values would be
   acceptable. To alleviate these limitations, the PEP can report
   some limitations of attribute values in the Attribute
   Limitations Table.

Device Identification Table

   This class contains a single policy rule instance that contains
   device-specific information that is used to facilitate
   efficient policy installation by a PDP. The instance of this
   class is reported to the PDP in a COPS request message so that
   the PDP can take into account certain device characteristics
   during policy installation.

2. Device Capabilities group

   This group contains the PRCs that contain the types of interfaces
   of the device and the Role Combinations assigned to them.

   Interface Capabilities Set Table

The interface types the PEP supports are described by rows in
this table (frwkIfCapSetTable). Each row, or instance of this
class, describes the characteristics of an interface type. The

PEP notifies the PDP of these interface types and then the PDP
configures the interfaces, per role combination.

Interface Capability and Role Combo Table

The Interface Cap Set Table describes the types of interfaces
the PEP supports by their capabilities. Configuration is done
in terms of these interface types and the role combinations
assigned to them; The PDP does not deal with individual
interfaces on the device. Each row of this class is a
<interface type, Role Combo> two-tuple.

3. Classifier group

This group contains the IP and IEEE 802 Classifier elements. The
set of tables consist of a Base Filter table that is extended to
form the IP Filter table and the 802 Filter table. The Filter
Group table forms sets of filters.

5. **The Framework PIB Module**

```
FRAMEWORK-PIB PIB-DEFINITIONS ::= BEGIN

IMPORTS
    Unsigned32, Integer32, MODULE-IDENTITY,
    MODULE-COMPLIANCE, OBJECT-TYPE
            FROM COPS-PR-SPPI
    PolicyInstanceId, PolicyReferenceId, Prid,
    PolicyTagId
            FROM COPS-PR-SPPI-TC
    InetAddress
            FROM INET-ADDRESS-MIB
    TruthValue, TEXTUAL-CONVENTION, PhysAddress
            FROM SNMPv2-TC
    Role, RoleCombination
            FROM POLICY-DEVICE-AUX-MIB
    SnmpAdminString
            FROM SNMP-FRAMEWORK-MIB
    OBJECT-GROUP
            FROM SNMPv2-CONF;

frameworkPib  MODULE-IDENTITY
    SUBJECT-CATEGORY { all }
    LAST-UPDATED "200007141200Z"
    ORGANIZATION "IETF RAP WG"
    CONTACT-INFO "
                    Michael Fine
                    Cisco Systems, Inc.
                    170 West Tasman Drive
                    San Jose, CA  95134-1706 USA
                    Phone: +1 408 527 8218
                    Email: mfine@cisco.com

                    Keith McCloghrie
                    Cisco Systems, Inc.
                    170 West Tasman Drive,
                    San Jose, CA 95134-1706 USA
                    Phone: +1 408 526 5260
                    Email: kzm@cisco.com

                    John Seligson
                    Nortel Networks, Inc.
                    4401 Great America Parkway
                    Santa Clara, CA 95054 USA
                    Phone: +1 408 495 2992
                    Email: jseligso@nortelnetworks.com"
    DESCRIPTION
```

```
        "A PIB module containing the base set of policy
         rule classes that are required for support of
         all policies."

    ::= { tbd }
```

```
     --
     -- The root OID for PRCs in the Framework PIB
     --


     frwkBasePibClasses
                 OBJECT IDENTIFIER ::= { frameworkPib 1 }


     --
     -- Textual Conventions
     --


     --
     -- PRC Support Table
     --


     frwkPrcSupportTable OBJECT-TYPE
         SYNTAX          SEQUENCE OF FrwkPrcSupportEntry
         PIB-ACCESS      notify,5
         STATUS          current
         DESCRIPTION
             "Each instance of this class specifies a PRC that the device
             supports and a bit string to indicate the attributes of the
             class that are supported.  These PRIs are sent to the PDP to
             indicate to the PDP which PRCs, and which attributes of
             these PRCs, the device supports. This table can also be
             downloaded by a network manager when static configuration is
             used.

             All install and install-notify PRCs supported by the device
             must be represented in this table. Notify PRCs may be
             represented for informational purposes."

         ::= { frwkBasePibClasses 1 }


     frwkPrcSupportEntry OBJECT-TYPE
         SYNTAX          FrwkPrcSupportEntry
         STATUS          current
         DESCRIPTION
             "An instance of the frwkPrcSupport class that identifies a
             specific PRC and associated attributes as supported
             by the device."

         INDEX { frwkPrcSupportPrid }
         UNIQUENESS { frwkPrcSupportSupportedPrc }

         ::= { frwkPrcSupportTable 1 }
```

```
FrwkPrcSupportEntry ::= SEQUENCE {
        frwkPrcSupportPrid           PolicyInstanceId,
        frwkPrcSupportSupportedPrc   OBJECT IDENTIFIER,
        frwkPrcSupportSupportedAttrs OCTET STRING,
        frwkPrcSupportMaxPris        Unsigned32
}


frwkPrcSupportPrid OBJECT-TYPE
    SYNTAX         PolicyInstanceId
    STATUS         current
    DESCRIPTION
        "An arbitrary integer index that uniquely identifies an
        instance of the frwkPrcSupport class."

    ::= { frwkPrcSupportEntry 1 }


frwkPrcSupportSupportedPrc OBJECT-TYPE
    SYNTAX         OBJECT IDENTIFIER
    STATUS         current
    DESCRIPTION
        "The object identifier of a supported PRC. There may not
        be more than one instance of the frwkPrcSupport class with
        the same value of frwkPrcSupportSupportedPrc."

    ::= { frwkPrcSupportEntry 2 }


frwkPrcSupportSupportedAttrs OBJECT-TYPE
    SYNTAX         OCTET STRING
    STATUS         current
    DESCRIPTION
        "A bit string representing the supported attributes of the
        class that is identified by the frwkPrcSupportSupportedPrc
        object.

        Each bit of this bit mask corresponds to a class attribute,
        with the most significant bit of the i-th octet of this
        octet string corresponding to the (8*i - 7)-th attribute,
        and the least significant bit of the i-th octet
        corresponding to the (8*i)-th class attribute. Each bit of
        this bit mask specifies whether or not the corresponding
        class attribute is currently supported, with a '1'
        indicating support and a '0' indicating no support. If the
        value of this bit mask is N bits long and there are more
        than N class attributes then the bit mask is logically
        extended with 0's to the required length."
```

```
    ::= { frwkPrcSupportEntry 3 }
```

```
frwkPrcSupportMaxPris OBJECT-TYPE
    SYNTAX          Unsigned32
    STATUS          current
    DESCRIPTION
        "A non-negative value indicating the maximum number of
        policy rule instances that can be installed in the
        identified policy rule class. Note that actual number of
        PRIs that can be installed in a PRC at any given time may be
        less than this value based on the current operational state
        (e.g.,resources currently consumed) of the device."

    ::= { frwkPrcSupportEntry 4 }


--
-- PIB Incarnation Table
--

frwkPibIncarnationTable OBJECT-TYPE
    SYNTAX          SEQUENCE OF FrwkPibIncarnationEntry
    PIB-ACCESS      install-notify,7
    STATUS          current
    DESCRIPTION
        "This class contains a single policy rule instance per
        installed context that identifies the current incarnation
        of the PIB and the PDP or network manager that installed
        this incarnation.  The instance of this class is reported to
        the PDP in the REQ message so that the PDP can (attempt to)
        ascertain the current state of the PIB and the active
        context. A network manager may use the instance to
        determine the state of the device with regard to existing
        NMS interactions."

    ::= { frwkBasePibClasses 2 }


frwkPibIncarnationEntry OBJECT-TYPE
    SYNTAX          FrwkPibIncarnationEntry
    STATUS          current
    DESCRIPTION
        "An instance of the frwkPibIncarnation class. Only
        one instance of this policy class is ever instantiated.
        per context"

    INDEX { frwkPibIncarnationPrid }
    UNIQUENESS { frwkPibIncarnationName }

    ::= { frwkPibIncarnationTable 1 }
```

```
    FrwkPibIncarnationEntry ::= SEQUENCE {
            frwkPibIncarnationPrid                  PolicyInstanceId,
            frwkPibIncarnationName                  SnmpAdminString,
            frwkPibIncarnationId                    OCTET STRING,
            frwkPibIncarnationLongevity             INTEGER,
            frwkPibIncarnationTtl                   Unsigned32,
            frwkPibIncarnationActive                TruthValue
    }


    frwkPibIncarnationPrid OBJECT-TYPE
        SYNTAX          PolicyInstanceId
        STATUS          current
        DESCRIPTION
            "An index to uniquely identify an instance of this
            policy class."

        ::= { frwkPibIncarnationEntry 1 }


    frwkPibIncarnationName OBJECT-TYPE
        SYNTAX          SnmpAdminString
        STATUS          current
        DESCRIPTION
            "The name of the PDP that installed the current incarnation
            of the PIB into the device.  By default, it is the zero
            length string."

        ::= { frwkPibIncarnationEntry 2 }


    frwkPibIncarnationId OBJECT-TYPE
        SYNTAX          OCTET STRING
        STATUS          current
        DESCRIPTION
            "An ID to identify the current incarnation.  It has meaning
            to the PDP/manager that installed the PIB and perhaps its
            standby PDPs/managers. By default, it is the zero-length
            string."

        ::= { frwkPibIncarnationEntry 3 }
```

```
frwkPibIncarnationLongevity OBJECT-TYPE
    SYNTAX          INTEGER {
                        expireNever(1),
                        expireImmediate(2),
                        expireOnTimeout(3)
                    }
    STATUS          current
    DESCRIPTION
        "This attribute controls what the PEP does with the
        downloaded policy on a Client Close message or a loss of
        connection to the PDP.

        If set to expireNever, the PEP continues to operate with the
        installed policy indefinitely.  If set to expireImmediate,
        the PEP immediately expires the policy obtained from the PDP
        and installs policy from local configuration.  If set to
        expireOnTimeout, the PEP continues to operate with the
        policy installed by the PDP for a period of time specified
        by frwkPibIncarnationTtl.  After this time (and it has not
        reconnected to the original or new PDP) the PEP expires this
        policy and reverts to local configuration.

        For all cases, it is the responsibility of the PDP to check
        the incarnation and download new policy, if necessary, on a
        reconnect.

        Policy enforcement timing only applies to policies that have
        been installed dynamically (e.g., by a PDP via COPS)."

    ::= { frwkPibIncarnationEntry 4 }


frwkPibIncarnationTtl OBJECT-TYPE
    SYNTAX          Unsigned32
    STATUS          current
    DESCRIPTION
        "The number of seconds after a Client Close or TCP timeout
        for which the PEP continues to enforce the policy in the
        PIB.
        After this interval, the PIB is considered expired and the
        device no longer enforces the policy installed in the PIB.

        This attribute is only meaningful if
        frwkPibIncarnationLongevity is set to expireOnTimeout."

    ::= { frwkPibIncarnationEntry 5 }
```

```
    frwkPibIncarnationActive OBJECT-TYPE
        SYNTAX          TruthValue
        STATUS          current
        DESCRIPTION
            "If this attribute is set to TRUE, then the PIB instance
            to which this PRI belongs becomes the active PIB instance.
            The previous active instance MUST become inactive and the
            frwkPibIncarnationActive attribute in that PIB instance
            MUST be set to false."

        ::= { frwkPibIncarnationEntry 6 }


    --
    -- Device Identification Table
    --

    -- This table supports the ability to export general
    -- purpose device information to facilitate efficient
    -- communication between the device and a PDP

    frwkDeviceIdTable OBJECT-TYPE
        SYNTAX          SEQUENCE OF FrwkDeviceIdEntry
        PIB-ACCESS      notify,5
        STATUS          current
        DESCRIPTION
            "This class contains a single policy rule instance that
            contains device-specific information that is used to
            facilitate efficient policy installation by a PDP. The
            instance of this class is reported to the PDP in a COPS
            request message so that the PDP can take into account
            certain device characteristics during policy installation."

        ::= { frwkBasePibClasses 3 }


    frwkDeviceIdEntry OBJECT-TYPE
        SYNTAX          FrwkDeviceIdEntry
        STATUS          current
        DESCRIPTION
            "An instance of the frwkDeviceId class. Only one instance of
            this policy class is ever instantiated."

        INDEX { frwkDeviceIdPrid }
        UNIQUENESS { frwkDeviceIdDescr }

        ::= { frwkDeviceIdTable 1 }
```

```
    FrwkDeviceIdEntry ::= SEQUENCE {
            frwkDeviceIdPrid        PolicyInstanceId,
            frwkDeviceIdDescr       SnmpAdminString,
            frwkDeviceIdMaxMsg      Unsigned32,
            frwkDeviceIdMaxContexts Unsigned32
    }


    frwkDeviceIdPrid OBJECT-TYPE
        SYNTAX          PolicyInstanceId
        STATUS          current
        DESCRIPTION
            "An index to uniquely identify an instance of this
            policy class."

        ::= { frwkDeviceIdEntry 1 }


    frwkDeviceIdDescr OBJECT-TYPE
        SYNTAX          SnmpAdminString (SIZE(0..255))
        STATUS          current
        DESCRIPTION
            "A textual description of the PEP. This value should include
            the name and version identification of the PEP's hardware
            and software."

        ::= { frwkDeviceIdEntry 2 }


    frwkDeviceIdMaxMsg OBJECT-TYPE
        SYNTAX          Unsigned32
        STATUS          current
        DESCRIPTION
            "The maximum message size, in octets, that the device
            is capable of processing. Received messages with a
            size in excess of this value must cause the PEP to return an
            error to the PDP containing the global error code
            'maxMsgSizeExceeded'."

        ::= { frwkDeviceIdEntry 3 }


    frwkDeviceIdMaxContexts OBJECT-TYPE
        SYNTAX          Unsigned32
        STATUS          current
        DESCRIPTION
            "The maximum number of unique contexts supported by
             the device."
```

```
   ::= { frwkDeviceIdEntry 4 }
```

```
    --
    -- Component Limitations Table
    --

    -- This table supports the ability to export information
    -- detailing policy class/attribute implementation limitations
    -- to the policy management system.

    frwkCompLimitsTable OBJECT-TYPE
        SYNTAX          SEQUENCE OF FrwkCompLimitsEntry
        PIB-ACCESS      notify,6
        STATUS          current
        DESCRIPTION
            "Each instance of this class identifies a policy class or
            attribute and a limitation related to the implementation of
            the class/attribute in the device. Additional information
            providing guidance related to the limitation may also be
            present. These PRIs are sent to the PDP to indicate which
            PRCs or PRC attributes the device supports in a restricted
            manner."

        ::= { frwkBasePibClasses 4 }


    frwkCompLimitsEntry OBJECT-TYPE
        SYNTAX          FrwkCompLimitsEntry
        STATUS          current
        DESCRIPTION
            "An instance of the frwkCompLimits class that identifies
            a PRC or PRC attribute and a limitation related to the PRC
            or PRC attribute implementation supported by the device.
            All PRIs of this class represent errors that would be
            returned in relation to the identified component for policy
            installation requests that don't abide by the restrictions
            indicated by the limitation type (error code) and, possibly,
            a provided guidance value."

        INDEX { frwkCompLimitsPrid }
        UNIQUENESS { frwkCompLimitsComponent,
                    frwkCompLimitsType,
                    frwkCompLimitsSubType,
                    frwkCompLimitsGuidance }

        ::= { frwkCompLimitsTable 1 }


    FrwkCompLimitsEntry ::= SEQUENCE {
            frwkCompLimitsPrid          PolicyInstanceId,
            frwkCompLimitsComponent     OBJECT IDENTIFIER,
```

```
        frwkCompLimitsType          Integer32,
        frwkCompLimitsSubType       INTEGER,
        frwkCompLimitsGuidance      OCTET STRING
}
```

    frwkCompLimitsPrid OBJECT-TYPE
        SYNTAX   PolicyInstanceId
        STATUS   current
        DESCRIPTION
            "An arbitrary integer index that uniquely identifies an
            instance of the frwkCompLimits class."

        ::= { frwkCompLimitsEntry 1 }

    frwkCompLimitsComponent OBJECT-TYPE
        SYNTAX   OBJECT IDENTIFIER
        STATUS   current
        DESCRIPTION
            "The object identifier of a PRC or PRC attribute that
            is supported in some limited fashion with regard to it's
            definition in the associated PIB module. The same PRC or
            PRC attribute identifier may appear in the table several
            times, once for each implementation limitation
            acknowledged by the device. "

        ::= { frwkCompLimitsEntry 2 }

    frwkCompLimitsType OBJECT-TYPE
        SYNTAX   Integer32
        STATUS   current
        DESCRIPTION
            "A value describing an implementation limitation for the
            device related to the PRC or PRC attribute identified by
            the frwkCompLimitsComponent data in this class instance.
            Values for this object are derived from the defined
            error values associated with the PRC of the identified
            attribute or the PRC itself. All genericPrc and specificPrc
            (defined in a PRC INSTALL-ERRORS clause) error codes
            represent valid limitation type values. The enumeration
            values for generic Class-Specific errors are listed in
            [COPS-PR].

            For example, an implementation of the frwkIpFilter class may
            be limited in several ways, such as address mask, protocol
            and Layer 4 port options. These limitations could be
            exported using this table with the following instances:

            Component   Type

            'frwkIpFilterDstAddrMask' 'attrValueSupLimited'
            'frwkIpFilterSrcAddrMask' 'attrValueSupLimited'
            'frwkIpFilterProtocol'   'attrValueSupLimited'

```
'frwkIpFilterProtocol'     'attrValueSupLimited'
'frwkIpFilterDstL4PortMin' 'invalidDstL4PortData'
'frwkIpFilterDstL4PortMax' 'invalidDstL4PortData'
'frwkBaseFilterPermit'     'attrEnumSupLimited'
```

The above entries describe a number of limitations that
may be in effect for the frwkIpFilter class on a given
device. The limitations include restrictions on acceptable
values for certain attributes and indications of the
relationship between related attributes.

Also, an implementation of a PRC may be limited in the ways
it can be accessed. For instance:
Component   Type

'DscpMapEntry'   'priNotifyOnly'

If the errors defined in the INSTALL-ERRORS section are not
generic Class-Specific errors (in the example,
'invalidDstL4PortData') then the Error code sent should be
'priSpecificError'[COPS-PR] and the Sub-Error code should
contain the enumeration value from the INSTALL-ERRORS
section for the PRC (in the example, the enumeration value
for 'invalidDstL4PortData') [SPPI]."

 ::= { frwkCompLimitsEntry 3 }

frwkCompLimitsSubType OBJECT-TYPE
     SYNTAX   INTEGER {
                         none(1),
                         lengthMin(2),
                         lengthMax(3),
                         rangeMin(4),
                         rangeMax(5),
                         enumMin(6),
                         enumMax(7),
                         enumOnly(8),
                         valueOnly(9),
                         extendsOid(10)
                     }
     STATUS   current
     DESCRIPTION
         "This object indicates the type of guidance related
         to the noted limitation (as indicated by the
         frwkCompLimitsType attribute) that is provided
         in the frwkCompLimitsGuidance attribute.

         A value of 'none(1)' means that no additional
         guidance is provided for the noted limitation type.

         A value of 'lengthMin(2)' means that the guidance
         attribute provides data related to the minimum

acceptable length for the value of the identified
component. A corresponding class instance
specifying the 'lengthMax(3)' value is required
in conjunction with this sub-type.

A value of 'lengthMax(3)' means that the guidance
attribute provides data related to the maximum
acceptable length for the value of the identified
component. A corresponding class instance
specifying the 'lengthMin(2)' value is required
in conjunction with this sub-type.

A value of 'rangeMin(4)' means that the guidance
attribute provides data related to the lower bound
of the range for the value of the identified
component. A corresponding class instance
specifying the 'rangeMax(5)' value is required
in conjunction with this sub-type.

A value of 'rangeMax(5)' means that the guidance
attribute provides data related to the upper bound
of the range for the value of the identified
component. A corresponding class instance
specifying the 'rangeMin(4)' value is required
in conjunction with this sub-type.

A value of 'enumMin(6)' means that the guidance
attribute provides data related to the lowest
enumeration acceptable for the value of the
identified component. A corresponding
class instance specifying the 'enumMax(7)'
value is required in conjunction with this sub-type.

A value of 'enumMin(7)' means that the guidance
attribute provides data related to the largest
enumeration acceptable for the value of the
identified component. A corresponding
class instance specifying the 'enumMin(6)'
value is required in conjunction with this sub-type.

A value of 'enumOnly(8)' means that the guidance
attribute provides data related to a single
enumeration acceptable for the value of the
identified component.

A value of 'valueOnly(9)' means that the guidance
attribute provides data related to a single
value that is acceptable for the identified
component.

A value of 'extendsOid(10)' means that the guidance
attribute provides data related to a PRC that
AUGMENTS or EXTENDS the identified policy class."

```
            ::= { frwkCompLimitsEntry 4 }
```

```
    frwkCompLimitsGuidance OBJECT-TYPE
          SYNTAX          OCTET STRING (SIZE(0..255))
          STATUS          current
          DESCRIPTION
              "A value used to convey additional information related
              to the implementation limitation noted by the
              frwkCompLimitsType and frwkCompLimitsSubType
              attribute. The value of this attribute must be
              interpreted in the context of the frwkCompLimitsType and
              frwkCompLimitsSubType values. Note that a guidance
              value will not necessarily be provided for all exported
              limitations. If a guidance value is not provided, the
              value must be a zero-length string.


              The format of the guidance value, if one is present as
              indicated by the frwkCompLimitsSubType attribute,
              is described by the following table. Note that the
              type of guidance value is dictated by the type of the
              component whose limitation is being exported.

              Base Type       Length  Value
              ---------       ------  -----
              INTEGER         <none>  32-bit value
              OCTET STRING  1 byte   <length> octets of data
              OID           1 byte   <length> 32-bit OID components."

          ::= { frwkCompLimitsEntry 5 }

  --
  -- The device interface capabilities and role combo classes group
  --

  frwkDeviceCapClasses
             OBJECT IDENTIFIER ::= { frameworkPib 2 }

  --
  -- Interface Capability Set Table
  --

  frwkIfCapSetTable OBJECT-TYPE
      SYNTAX          SEQUENCE OF FrwkIfCapSetEntry
      PIB-ACCESS      notify,4
      STATUS          current
      DESCRIPTION
          "Interface type definitions. This class describes the types
          of interfaces that exist on the device. An interface type is
          defined by its name. Associated with each interface type is
          a set of capabilities. These capabilities are used by the
```

PDP to determine policy information to be associated with
        interfaces of this type."

    ::= { frwkDeviceCapClasses 1 }

```
   frwkIfCapSetEntry OBJECT-TYPE
       SYNTAX          FrwkIfCapSetEntry
       STATUS          current
       DESCRIPTION
           "An instance of this class describes the characteristics
           of a type of an interface."

       INDEX { frwkIfCapSetPrid }
       UNIQUENESS { frwkIfCapSetName,
                    frwkIfCapSetCapability }

       ::= { frwkIfCapSetTable 1 }


   FrwkIfCapSetEntry ::= SEQUENCE {
           frwkIfCapSetPrid        PolicyInstanceId,
           frwkIfCapSetName        SnmpAdminString,
           frwkIfCapSetCapability  Prid
   }


   frwkIfCapSetPrid OBJECT-TYPE
       SYNTAX          PolicyInstanceId
       STATUS          current
       DESCRIPTION
           "An arbitrary integer index that uniquely identifies a
           instance of the class."

       ::= { frwkIfCapSetEntry 1 }


   frwkIfCapSetName OBJECT-TYPE
       SYNTAX          SnmpAdminString
       STATUS          current
       DESCRIPTION
           "The name for the capability set.  The capability set name
           is the unique identifier of an interface type."

       ::= { frwkIfCapSetEntry 2 }


   frwkIfCapSetCapability OBJECT-TYPE
       SYNTAX      Prid
       STATUS      current
       DESCRIPTION
           "The complete OID specifying the PRC and the instance of the
           PRC containing a set of capabilities of the interface."

       ::= { frwkIfCapSetEntry 3 }
```

```
    --
    -- Interface Capabilities Set Name and Role Combination Table
    --

    frwkIfCapSetRoleComboTable OBJECT-TYPE
        SYNTAX          SEQUENCE OF FrwkIfCapSetRoleComboEntry
        PIB-ACCESS      notify,4
        STATUS          current
        DESCRIPTION
            "Policy for an interface may depend not only on the type
            of interface but also on its roles.  This table specifies
            all the <interface type, role combination> tuples currently
            on the device."

        ::= { frwkDeviceCapClasses 2 }


    frwkIfCapSetRoleComboEntry OBJECT-TYPE
        SYNTAX          FrwkIfCapSetRoleComboEntry
        STATUS          current
        DESCRIPTION
            "An instance of this class describes a combination of an
            interface type and a role combination."

        INDEX { frwkIfCapSetRoleComboPrid }
        UNIQUENESS { frwkIfCapSetRoleComboName,
                     frwkIfCapSetRoleComboRoles }

        ::= { frwkIfCapSetRoleComboTable 1 }


    FrwkIfCapSetRoleComboEntry ::= SEQUENCE {
            frwkIfCapSetRoleComboPrid   PolicyInstanceId,
            frwkIfCapSetRoleComboName   SnmpAdminString,
            frwkIfCapSetRoleComboRoles  RoleCombination
    }


    frwkIfCapSetRoleComboPrid OBJECT-TYPE
        SYNTAX          PolicyInstanceId
        STATUS          current
        DESCRIPTION
            "An arbitrary integer index that uniquely identifies a
            instance of the class."

        ::= { frwkIfCapSetRoleComboEntry 1 }
```

```
    frwkIfCapSetRoleComboName OBJECT-TYPE
        SYNTAX          SnmpAdminString
        STATUS          current
        DESCRIPTION
            "The name of the interface type.  This name must exist in
            frwkIfCapSetTable."

        ::= { frwkIfCapSetRoleComboEntry 2 }


    frwkIfCapSetRoleComboRoles OBJECT-TYPE
        SYNTAX          RoleCombination
        STATUS          current
        DESCRIPTION
            "A role combination.  The PEP requires policy for interfaces
            with this role combination and of type
            frwkIfCapSetRoleComboName"

        ::= { frwkIfCapSetRoleComboEntry 3 }


    --
    -- The Classification classes group
    --

    frwkClassifierClasses
            OBJECT IDENTIFIER ::= { frameworkPib 3 }

    --
    -- The Base Filter Table
    --

    frwkBaseFilterTable OBJECT-TYPE
        SYNTAX          SEQUENCE OF FrwkBaseFilterEntry
        PIB-ACCESS      install,3
        STATUS          current
        DESCRIPTION
            "The Base Filter class.  A packet has to match all
            fields in an Filter.  Wildcards may be specified for those
            fields that are not relevant."

    ::= { frwkClassifierClasses 1 }


    frwkBaseFilterEntry OBJECT-TYPE
        SYNTAX          FrwkBaseFilterEntry
        STATUS          current
        DESCRIPTION
            "An instance of the frwkBaseFilter class."
```

```
    INDEX { frwkBaseFilterPrid }

    ::= { frwkBaseFilterTable 1 }
```

```
    FrwkBaseFilterEntry ::= SEQUENCE {
            frwkBaseFilterPrid        PolicyInstanceId,
            frwkBaseFilterPermit      TruthValue
    }


    frwkBaseFilterPrid OBJECT-TYPE
        SYNTAX        PolicyInstanceId
        STATUS        current
        DESCRIPTION
            "An integer index to uniquely identify this Filter among all
            the Filters."

        ::= { frwkBaseFilterEntry 1 }


    frwkBaseFilterPermit OBJECT-TYPE
        SYNTAX        TruthValue
        STATUS        current
        DESCRIPTION
            "If the packet matches this filter and the value of this
            attribute is true, then the matching process terminates
            and the action associated with this filter (indirectly
            through the filter group) is applied to the packet.  If the
            value of this attribute is false, then no more filters in
            the filter group are compared to this packet and matching
            continues with the first filter of the next filter group."

        ::= { frwkBaseFilterEntry 2 }


    --
    -- The IP Filter Table
    --

    frwkIpFilterTable OBJECT-TYPE
        SYNTAX        SEQUENCE OF FrwkIpFilterEntry
        PIB-ACCESS    install,11
        STATUS        current
        DESCRIPTION
            "Filter definitions.  A packet has to match all fields in a
            filter.  Wildcards may be specified for those fields that
            are not relevant."

        INSTALL-ERRORS {
            invalidDstL4PortData(1),
            invalidSrcL4PortData(2)
            }
        ::= { frwkClassifierClasses 2 }
```

```
frwkIpFilterEntry OBJECT-TYPE
    SYNTAX          FrwkIpFilterEntry
    STATUS          current
    DESCRIPTION
        "An instance of the frwkIpFilter class."

    EXTENDS { frwkBaseFilterEntry }
    UNIQUENESS { frwkIpFilterDstAddr,
                 frwkIpFilterDstAddrMask,
                 frwkIpFilterSrcAddr,
                 frwkIpFilterSrcAddrMask,
                 frwkIpFilterDscp,
                 frwkIpFilterProtocol,
                 frwkIpFilterDstL4PortMin,
                 frwkIpFilterDstL4PortMax,
                 frwkIpFilterSrcL4PortMin,
                 frwkIpFilterSrcL4PortMax }

    ::= { frwkIpFilterTable 1 }


FrwkIpFilterEntry ::= SEQUENCE {
        frwkIpFilterDstAddr      InetAddress,
        frwkIpFilterDstAddrMask  InetAddress,
        frwkIpFilterSrcAddr      InetAddress,
        frwkIpFilterSrcAddrMask  InetAddress,
        frwkIpFilterDscp         Integer32,
        frwkIpFilterProtocol     INTEGER,
        frwkIpFilterDstL4PortMin INTEGER,
        frwkIpFilterDstL4PortMax INTEGER,
        frwkIpFilterSrcL4PortMin INTEGER,
        frwkIpFilterSrcL4PortMax INTEGER
}


frwkIpFilterDstAddr OBJECT-TYPE

    SYNTAX          InetAddress
    STATUS          current
    DESCRIPTION
        "The IP address to match against the packet's destination IP
        address."

    ::= { frwkIpFilterEntry 1 }
```

    frwkIpFilterDstAddrMask OBJECT-TYPE
        SYNTAX   InetAddress
        STATUS   current
        DESCRIPTION
            "A mask for the matching of the destination IP address.
            A zero bit in the mask means that the corresponding bit in
            the address always matches."

        ::= { frwkIpFilterEntry 2 }

    frwkIpFilterSrcAddr OBJECT-TYPE
        SYNTAX   InetAddress
        STATUS   current
        DESCRIPTION
            "The IP address to match against the packet's source IP
            address."

        ::= { frwkIpFilterEntry 3 }

    frwkIpFilterSrcAddrMask OBJECT-TYPE
        SYNTAX   InetAddress
        STATUS   current
        DESCRIPTION
            "A mask for the matching of the source IP address."

        ::= { frwkIpFilterEntry 4 }

    frwkIpFilterDscp OBJECT-TYPE
        SYNTAX   Integer32 (-1 | 0..63)
        STATUS   current
        DESCRIPTION
            "The value that the DSCP in the packet can have and
            match this filter. A value of -1 indicates that a specific
            DSCP value has not been defined and thus all DSCP values
            are considered a match."

        ::= { frwkIpFilterEntry 5 }

    frwkIpFilterProtocol OBJECT-TYPE
        SYNTAX   INTEGER (0..255)
        STATUS   current
        DESCRIPTION
            "The IP protocol to match against the packet's protocol.
            A value of zero means match all."

```
       ::= { frwkIpFilterEntry 6 }
```

```
   frwkIpFilterDstL4PortMin OBJECT-TYPE
       SYNTAX         INTEGER (0..65535)
       STATUS         current
       DESCRIPTION
           "The minimum value that the packet's layer 4 destination
           port number can have and match this filter."

       ::= { frwkIpFilterEntry 7 }


   frwkIpFilterDstL4PortMax OBJECT-TYPE
       SYNTAX         INTEGER (0..65535)
       STATUS         current
       DESCRIPTION
           "The maximum value that the packet's layer 4 destination
           port number can have and match this filter. This value must
           be equal to or greater that the value specified for this
           filter in frwkIpFilterDstL4PortMin."

       ::= { frwkIpFilterEntry 8 }


   frwkIpFilterSrcL4PortMin OBJECT-TYPE
       SYNTAX         INTEGER (0..65535)
       STATUS         current
       DESCRIPTION
           "The minimum value that the packet's layer 4 source port
           number can have and match this filter."

       ::= { frwkIpFilterEntry 9 }


   frwkIpFilterSrcL4PortMax OBJECT-TYPE
       SYNTAX         INTEGER (0..65535)
       STATUS         current
       DESCRIPTION
           "The maximum value that the packet's layer 4 source port
           number can have and match this filter.  This value must be
           equal to or greater that the value specified for this filter
           in frwkIpFilterSrcL4PortMin."

       ::= { frwkIpFilterEntry 10 }
```

```
    --
    -- The IEEE 802 Filter Table
    --

    -- The IEEE 802 Filter Table supports the specification of IEEE
    -- 802-based (e.g., 802.3) information that is used to perform
    -- traffic classification.
    --

    frwk802FilterTable OBJECT-TYPE
        SYNTAX          SEQUENCE OF Frwk802FilterEntry
        PIB-ACCESS      install,9
        STATUS          current
        DESCRIPTION
            "IEEE 802-based filter definitions. A class that contains
            attributes of IEEE 802 (e.g., 802.3) traffic that form
            filters that are used to perform traffic classification."

        ::= { frwkClassifierClasses 3 }


    frwk802FilterEntry OBJECT-TYPE
        SYNTAX          Frwk802FilterEntry
        STATUS          current
        DESCRIPTION
            "IEEE 802-based filter definitions.  An entry specifies
            (potentially) several distinct matching components. Each
            component is tested against the data in a frame
            individually. An overall match occurs when all of the
            individual components match the data they are compared
            against in the frame being processed. A failure of any
            one test causes the overall match to fail.

            Wildcards may be specified for those fields that are not
            relevant."

        EXTENDS { frwkBaseFilterEntry }
        UNIQUENESS { frwk802FilterDstAddr,
                    frwk802FilterDstAddrMask,
                    frwk802FilterSrcAddr,
                    frwk802FilterSrcAddrMask,
                    frwk802FilterVlanId,
                    frwk802FilterVlanTagRequired,
                    frwk802FilterEtherType,
                    frwk802FilterUserPriority }

        ::= { frwk802FilterTable 1 }
```

```
Frwk802FilterEntry ::= SEQUENCE {
        frwk802FilterDstAddr        PhysAddress,
        frwk802FilterDstAddrMask    PhysAddress,
        frwk802FilterSrcAddr        PhysAddress,
        frwk802FilterSrcAddrMask    PhysAddress,
        frwk802FilterVlanId         Integer32,
        frwk802FilterVlanTagRequired INTEGER,
        frwk802FilterEtherType      Integer32,
        frwk802FilterUserPriority   BITS
}
```

```
frwk802FilterDstAddr OBJECT-TYPE
    SYNTAX          PhysAddress
    STATUS          current

    DESCRIPTION
        "The 802 address against which the 802 DA of incoming
        traffic streams will be compared. Frames whose 802 DA
        matches the physical address specified by this object,
        taking into account address wildcarding as specified by the
        frwk802FilterDstAddrMask object, are potentially subject to
        the processing guidelines that are associated with this
        entry through the related action class."

    ::= { frwk802FilterEntry 1 }
```

```
frwk802FilterDstAddrMask OBJECT-TYPE
    SYNTAX          PhysAddress
    STATUS          current
    DESCRIPTION
        "This object specifies the bits in a 802 destination address
        that should be considered when performing a 802 DA
        comparison against the address specified in the
        frwk802FilterDstAddr object.

        The value of this object represents a mask that is logically
        and'ed with the 802 DA in received frames to derive the
        value to be compared against the frwk802FilterDstAddr
        address. A zero bit in the mask thus means that the
        corresponding bit in the address always matches. The
        frwk802FilterDstAddr value must also be masked using this
        value prior to any comparisons.

        The length of this object in octets must equal the length in
        octets of the frwk802FilterDstAddr. Note that a mask with no
        bits set (i.e., all zeroes) effectively wildcards the
        frwk802FilterDstAddr object."
```

```
      ::= { frwk802FilterEntry 2 }
```

    frwk802FilterSrcAddr OBJECT-TYPE
        SYNTAX   PhysAddress
        STATUS   current
        DESCRIPTION
            "The 802 MAC address against which the 802 MAC SA of
            incoming traffic streams will be compared. Frames whose 802
            MAC SA matches the physical address specified by this
            object, taking into account address wildcarding as specified
            by the frwk802FilterSrcAddrMask object, are potentially
            subject to the processing guidelines that are associated
            with this entry through the related action class."

        ::= { frwk802FilterEntry 3 }

    frwk802FilterSrcAddrMask OBJECT-TYPE
        SYNTAX   PhysAddress
        STATUS   current
        DESCRIPTION
            "This object specifies the bits in a 802 MAC source address
            that should be considered when performing a 802 MAC SA
            comparison against the address specified in the
            frwk802FilterSrcAddr object.

            The value of this object represents a mask that is logically
            and'ed with the 802 MAC SA in received frames to derive the
            value to be compared against the frwk802FilterSrcAddr
            address. A zero bit in the mask thus means that the
            corresponding bit in the address always matches. The
            frwk802FilterSrcAddr value must also be masked using this
            value prior to any comparisons.

            The length of this object in octets must equal the length in
            octets of the frwk802FilterSrcAddr. Note that a mask with no
            bits set (i.e., all zeroes) effectively wildcards the
            frwk802FilterSrcAddr object."

        ::= { frwk802FilterEntry 4 }

    frwk802FilterVlanId OBJECT-TYPE
        SYNTAX   Integer32 (-1 | 1..4094)
        STATUS   current
        DESCRIPTION
            "The VLAN ID (VID) that uniquely identifies a VLAN
            within the device. This VLAN may be known or unknown
            (i.e., traffic associated with this VID has not yet
            been seen by the device) at the time this entry
            is instantiated.

Setting the frwk802FilterVlanId object to -1 indicates that
VLAN data should not be considered during traffic
classification."

::= { frwk802FilterEntry 5 }

```
    frwk802FilterVlanTagRequired OBJECT-TYPE
        SYNTAX          INTEGER {
                            taggedOnly(1),
                            priorityTaggedPlus(2),
                            untaggedOnly(3),
                            ignoreTag(4)
                        }
        STATUS          current
        DESCRIPTION
            "This object indicates whether the presence of an
            IEEE 802.1Q VLAN tag in data link layer frames must
            be considered when determining if a given frame
            matches this 802 filter entry.

            A value of 'taggedOnly(1)' means that only frames
            containing a VLAN tag with a non-Null VID (i.e., a
            VID in the range 1..4094) will be considered a match.

            A value of 'priorityTaggedPlus(2)' means that only
            frames containing a VLAN tag, regardless of the value
            of the VID, will be considered a match.

            A value of 'untaggedOnly(3)' indicates that only
            untagged frames will match this filter component.

            The presence of a VLAN tag is not taken into
            consideration in terms of a match if the value is
            'ignoreTag(4)'."

        ::= { frwk802FilterEntry 6 }


    frwk802FilterEtherType OBJECT-TYPE
        SYNTAX          Integer32 (-1 | 0..'ffff'h)
        STATUS          current
        DESCRIPTION
            "This object specifies the value that will be compared
            against the value contained in the EtherType field of an
            IEEE 802 frame. Example settings would include 'IP'
            (0x0800), 'ARP' (0x0806) and 'IPX' (0x8137).

            Setting the frwk802FilterEtherTypeMin object to -1 indicates
            that EtherType data should not be considered during traffic
            classification.

            Note that the position of the EtherType field depends on
            the underlying frame format. For Ethernet-II encapsulation,
            the EtherType field follows the 802 MAC source address. For
```

802.2 LLC/SNAP encapsulation, the EtherType value follows
the Organization Code field in the 802.2 SNAP header. The
value that is tested with regard to this filter component
therefore depends on the data link layer frame format being

```
        used. If this 802 filter component is active when there is
        no EtherType field in a frame (e.g., 802.2 LLC), a match is
        implied."

    ::= { frwk802FilterEntry 7 }


frwk802FilterUserPriority OBJECT-TYPE
    SYNTAX          BITS {
                        matchPriority0(0),
                        matchPriority1(1),
                        matchPriority2(2),
                        matchPriority3(3),
                        matchPriority4(4),
                        matchPriority5(5),
                        matchPriority6(6),
                        matchPriority7(7)
                    }
    STATUS          current
    DESCRIPTION
        "The set of values, representing the potential range
        of user priority values, against which the value contained
        in the user priority field of a tagged 802.1 frame is
        compared. A test for equality is performed when determining
        if a match exists between the data in a data link layer
        frame and the value of this 802 filter component. Multiple
        values may be set at one time such that potentially several
        different user priority values may match this 802 filter
        component.

        Setting all of the bits that are associated with this
        object causes all user priority values to match this
        attribute. This essentially makes any comparisons
        with regard to user priority values unnecessary. Untagged
        frames are treated as an implicit match."

    ::= { frwk802FilterEntry 8 }


--
-- The Filter Group Definition Table
--

frwkFilterGroupDefnTable OBJECT-TYPE
    SYNTAX          SEQUENCE OF FrwkFilterGroupDefnEntry
    PIB-ACCESS      install,5
    STATUS          current
    DESCRIPTION
        "A class that defines Filter Groups. Each Group being an
```

ordered list of filters. Each instance of this class
identifies one filter of a group and the precedence order of
that filter with respect to other filters in the same
group."

```
      INSTALL-ERRORS {
          priPrecedenceConflict(1) -- precedence conflict detected
          }

      ::= { frwkClassifierClasses 4 }


   frwkFilterGroupDefnEntry OBJECT-TYPE
      SYNTAX          FrwkFilterGroupDefnEntry
      STATUS          current
      DESCRIPTION
          "An instance of the frwkFilterGroupDefn class."

      INDEX { frwkFilterGroupDefnPrid }

      UNIQUENESS { frwkFilterGroupDefnId,
                   frwkFilterGroupDefnFilterId }

      ::= { frwkFilterGroupDefnTable 1 }


   FrwkFilterGroupDefnEntry ::= SEQUENCE {
          frwkFilterGroupDefnPrid            PolicyInstanceId,
          frwkFilterGroupDefnId             PolicyTagId,
          frwkFilterGroupDefnFilterId       PolicyReferenceId,
          frwkFilterGroupDefnFilterPrecedence Unsigned32
   }


   frwkFilterGroupDefnPrid OBJECT-TYPE
      SYNTAX          PolicyInstanceId
      STATUS          current
      DESCRIPTION
          "Unique index of this policy rule instance."

      ::= { frwkFilterGroupDefnEntry 1 }


   frwkFilterGroupDefnId OBJECT-TYPE
      SYNTAX          PolicyTagId
      STATUS          current
      DESCRIPTION
          "An ID for this Filter Group.  There will be one instance of
          the class frwkFilterGroupDefn with this ID for each
          instance of the Base filter class in the Filter Group per
          role combination.

          Note that this identifier is used in instances of the
```

Class that associate a Filter Group with an interface
set and specific actions. An active Filter Group-Target
association prohibits the deletion of all of the
frwkFilterGroupDefn instances with a given

        frwkFilterGroupDefnId (i.e., at
        least one entry for the specific frwkFilterGroupDefnId
        must be present in this table) until the Filter Group-Target
        association is terminated."

    ::= { frwkFilterGroupDefnEntry 2 }

  frwkFilterGroupDefnFilterId OBJECT-TYPE
     SYNTAX   PolicyReferenceId
     PIB-REFERENCES {frwkBaseFilterEntry}
     STATUS   current
     DESCRIPTION
         "This attribute specifies the filter in the
         frwkBaseFilterTable that is in the Filter Group specified by
         frwkFilterGroupDefnId at the position specified by the
         FilterPrecedence attribute.

         Attempting to specify an unknown class instance will result
         in an appropriate error indication being returned to the
         entity that is attempting to install the conflicting entry.
         For example, a 'priUnknown(2)' error indication is returned
         to the policy server in this situation."

    ::= { frwkFilterGroupDefnEntry 3 }

  frwkFilterGroupDefnFilterPrecedence OBJECT-TYPE
     SYNTAX   Unsigned32
     STATUS   current
     DESCRIPTION
         "The precedence order of this filter. The precedence order
         determines the position of this filter in the Filter Group.
         A filter with a given precedence order is positioned in the
         Filter group before one with a higher-valued
         precedence order.

         Precedence values within a group must be unique otherwise
         instance installation will be prohibited and an error
         value will be returned."

    ::= { frwkFilterGroupDefnEntry 4 }

```
       --
       -- Conformance Section
       --

       frwkBasePibConformance
                      OBJECT IDENTIFIER ::= { frameworkPib 4 }

       frwkBasePibCompliances
                      OBJECT IDENTIFIER ::= { frwkBasePibConformance 1 }

       frwkBasePibGroups
                      OBJECT IDENTIFIER ::= { frwkBasePibConformance 2 }


       frwkBasePibCompliance MODULE-COMPLIANCE
           STATUS   current
           DESCRIPTION
                   "Describes the requirements for conformance to the
                   Framework PIB."

           MODULE  -- this module
               MANDATORY-GROUPS { frwkPrcSupportGroup,
                                  frwkPibIncarnationGroup,
                                  frwkDeviceIdGroup,
                                  frwkCompLimitsGroup,
                                  frwkIfCapSetGroup,
                                  frwkIfCapSetRoleComboGroup }

               OBJECT          frwkPibIncarnationLongevity
               PIB-MIN-ACCESS  notify
               DESCRIPTION     "Install support is not required."

               OBJECT          frwkPibIncarnationTtl
               PIB-MIN-ACCESS  notify
               DESCRIPTION     "Install support is not required."

               OBJECT          frwkPibIncarnationActive
               PIB-MIN-ACCESS  notify
               DESCRIPTION     "Install support is not required."

           GROUP   frwkBaseFilterGroup
               DESCRIPTION
                   "The frwkBaseFilterGroup is mandatory if filtering
                    based on traffic components is supported."

           GROUP   frwkIpFilterGroup
               DESCRIPTION
                   "The frwkIpFilterGroup is mandatory if filtering
                    based on IP traffic components is supported."
```

```
        GROUP   frwk802FilterGroup
            DESCRIPTION
                "The frwk802FilterGroup is mandatory if filtering
                based on 802 traffic criteria is supported."

        GROUP   frwkFilterGroupDefnGroup
            DESCRIPTION
                "The frwkFilterGroupDefnGroup is mandatory if
                filtering based on IP traffic components is
                supported."

        ::= { frwkBasePibCompliances 1 }


    frwkPrcSupportGroup OBJECT-GROUP
        OBJECTS {
                frwkPrcSupportSupportedPrc,
                frwkPrcSupportSupportedAttrs,
                frwkPrcSupportMaxPris
        }
        STATUS  current
        DESCRIPTION
                "Objects from the frwkPrcSupportTable."

        ::= { frwkBasePibGroups 1 }


    frwkPibIncarnationGroup OBJECT-GROUP
        OBJECTS {
                frwkPibIncarnationName,
                frwkPibIncarnationId,
                frwkPibIncarnationLongevity,
                frwkPibIncarnationTtl,
                frwkPibIncarnationActive
        }
        STATUS  current
        DESCRIPTION
                "Objects from the frwkDevicePibIncarnationTable."

        ::= { frwkBasePibGroups 2 }

    frwkDeviceIdGroup OBJECT-GROUP
        OBJECTS {
                frwkDeviceIdDescr,
                frwkDeviceIdMaxMsg,
                frwkDeviceIdMaxContexts }
        STATUS  current
        DESCRIPTION
                "Objects from the frwkDeviceIdTable."
```

```
     ::= { frwkBasePibGroups 3 }
```

```
frwkCompLimitsGroup OBJECT-GROUP
    OBJECTS {
            frwkCompLimitsComponent,
            frwkCompLimitsType,
            frwkCompLimitsGuidance,
            frwkCompLimitsSubType }
    STATUS  current
    DESCRIPTION
            "Objects from the frwkCompLimitsTable."

    ::= { frwkBasePibGroups 4 }



frwkIfCapSetGroup OBJECT-GROUP
    OBJECTS {
            frwkIfCapSetName,
            frwkIfCapSetCapability
    }
    STATUS  current
    DESCRIPTION
            "Objects from the frwkIfCapSetTable."

    ::= { frwkBasePibGroups 5 }



frwkIfCapSetRoleComboGroup OBJECT-GROUP
    OBJECTS {
            frwkIfCapSetRoleComboName,
            frwkIfCapSetRoleComboRoles
    }
    STATUS  current
    DESCRIPTION
            "Objects from the frwkIfCapSetRoleComboTable."

    ::= { frwkBasePibGroups 6 }



frwkBaseFilterGroup OBJECT-GROUP
    OBJECTS {
            frwkBaseFilterPermit
    }
    STATUS  current
    DESCRIPTION
            "Objects from the frwkBaseFilterTable."

    ::= { frwkBasePibGroups 7 }
```

```
    frwkIpFilterGroup OBJECT-GROUP
        OBJECTS {
                 frwkIpFilterDstAddr,
                 frwkIpFilterDstAddrMask,
                 frwkIpFilterSrcAddr,
                 frwkIpFilterSrcAddrMask,
                 frwkIpFilterDscp,
                 frwkIpFilterProtocol,
                 frwkIpFilterDstL4PortMin,
                 frwkIpFilterDstL4PortMax,
                 frwkIpFilterSrcL4PortMin,
                 frwkIpFilterSrcL4PortMax
        }
        STATUS  current
        DESCRIPTION
                 "Objects from the frwkIpFilterTable."

        ::= { frwkBasePibGroups 8 }


    frwk802FilterGroup OBJECT-GROUP
        OBJECTS {
                 frwk802FilterDstAddr,
                 frwk802FilterDstAddrMask,
                 frwk802FilterSrcAddr,
                 frwk802FilterSrcAddrMask,
                 frwk802FilterVlanId,
                 frwk802FilterVlanTagRequired,
                 frwk802FilterEtherType,
                 frwk802FilterUserPriority
        }
        STATUS  current
        DESCRIPTION
                 "Objects from the frwk802FilterTable."

        ::= { frwkBasePibGroups 9 }


    frwkFilterGroupDefnGroup OBJECT-GROUP
        OBJECTS {
                 frwkFilterGroupDefnId,
                 frwkFilterGroupDefnFilterId,
                 frwkFilterGroupDefnFilterPrecedence
        }
        STATUS  current
        DESCRIPTION
                 "Objects from the frwkFilterGroupDefnTable."
```

```
    ::= { frwkBasePibGroups 10 }

END
```

6. Security Considerations

   The information contained in a PIB when transported by the COPS
   protocol [COPS-PR] may be sensitive, and its function of
   provisioning a PEP requires that only authorized communication take
   place. The use of IPSEC between PDP and PEP, as described in
   [COPS], provides the necessary protection against these threats.

7. Intellectual Property Considerations

   The IETF is being notified of intellectual property rights claimed
   in regard to some or all of the specification contained in this
   document. For more information consult the online list of claimed
   rights.

8. Author Information and Acknowledgments

        Michael Fine
        Cisco Systems, Inc.
        170 West Tasman Drive
        San Jose, CA 95134-1706 USA
        Phone: +1 408 527 8218
        Email: mfine@cisco.com

        Keith McCloghrie
        Cisco Systems, Inc.
        170 West Tasman Drive
        San Jose, CA 95134-1706 USA
        Phone: +1 408 526 5260
        Email: kzm@cisco.com

        John Seligson
        Nortel Networks, Inc.
        4401 Great America Parkway
        Santa Clara, CA 95054 USA
        Phone: +1 408 495 2992
        Email: jseligso@nortelnetworks.com

        Kwok Ho Chan
        Nortel Networks, Inc.
        600 Technology Park Drive
        Billerica, MA 01821 USA
        Phone: +1 978 288 8175
        Email: khchan@nortelnetworks.com

        Scott Hahn
        Intel Corp.
        2111 NE 25th Avenue
        Hillsboro, OR 97124 USA
        Phone: +1 503 264 8231
        Email: scott.hahn@intel.com

        Ravi Sahita
        Intel Corp.
        2111 NE 25th Avenue
        Hillsboro, OR 97124 USA
        Phone: +1 503 712 1554
        Email: ravi.sahita@intel.com

        Andrew Smith
        Fax: +1 415 345 1827
        Email: ah_smith@pacbell.net

        Francis Reichmeyer
        IPHighway Inc.
        Parker Plaza, 16th Floor
        400 Kelby St.
        Fort-Lee, NJ 07024
        Phone: (201) 585-0800
        Email: FranR@iphighway.com

9. References

    [COPS]
        Boyle, J., Cohen, R., Durham, D., Herzog, S., Rajan, R., and
        A. Sastry, "The COPS (Common Open Policy Service) Protocol"
        RFC 2748, January 2000.

    [COPS-PR]
        K. Chan, D. Durham, S. Gai, S. Herzog, K. McCloghrie,
        F. Reichmeyer, J. Seligson, A. Smith, R. Yavatkar, "COPS Usage
        for Policy Provisioning," draft-ietf-rap-pr-03.txt,
        July 2000.

    [SPPI]
        K. McCloghrie, et.al., "Structure of Policy Provisioning
        Information," draft-ietf-rap-sppi-01.txt, July 2000.

    [POLICY]
        M. Stevens, W. Weiss H. Mahon, B. Moore, J. Strassner,

G. Waters, A. Westerinen, J. Wheeler, "Policy Framework",
draft-ietf-policy-framework-00.txt, September 1999.

   [RAP-FRAMEWORK]
        R. Yavatkar, D. Pendarakis, "A Framework for Policy-based
        Admission Control", RFC 2753, January 2000.

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

Table of Contents