ForCES MIB

Status of this Memo

Copyright Notice

Abstract

   This memo defines a Management Information Base (MIB) for use with
   network management protocols in the Internet community. In
   particular, it defines a MIB for the Forwarding and Control Element
   Separation (ForCES) Network Element (NE). The ForCES working group
   is defining a protocol to allow a Control Element (CE) to control the
   behavior of a Forwarding Element (FE).

Conventions used in this document

   The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT",
   "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this
   document are to be interpreted as described in RFC-2119 [RFC2119].

Table of Contents

1. Introduction

   The ForCES MIB is a primarily read-only MIB that captures information
   related to the ForCES protocol. This includes state information about
   the associations between CE(s) and FE(s) in the NE.

   The ForCES MIB does not include information that is specified in
   other MIBs, such as packet counters for interfaces, etc.

   More specifically , the information in the ForCES MIB relative to
   associations includes:

- identifiers of the elements in the association
- state of the association
- configuration parameters of the association
- statistics of the association

The relevant references from the ForCES requirements and architecture
documents are repeated below:

From the ForCES requirements RFC [RFC 3654], Section 4, point 4:

  A NE MUST support the appearance of a single functional device. For
  example, in a router, the TTL of the packet should be decremented
  only once as it traverses the NE regardless of how many FEs through
  which it passes. However, external entities (e.g., FE managers and
  CE managers) MAY have direct access to individual ForCES protocol
  elements for providing information to transition them from the pre-
  association to post-association phase.

And [RFC 3654], Section 4, point 14:

  1. The ability for a management tool (e.g., SNMP) to be used to
  read(but not change) the state of FE SHOULD NOT be precluded.
  2. It MUST NOT be possible for management tools (e.g., SNMP, etc)
  to change the state of a FE in a manner that affects overall NE
  behavior without the CE being notified.

According to the ForCES architecture RFC [RFC 3746], Section 3.3:

  CE managers may be physically and logically separate entities that
  configure the CE with FE information via such mechanisms as COPS-PR
  [7] or SNMP [5].

and [RFC 3746], Section 5.7:

  RFC 1812 [2] also dictates that "Routers MUST be manageable by
  SNMP". In general, for the post-association phase, most external
  management tasks (including SNMP) should be done through
  interaction with the CE in order to support the appearance of a
  single functional device. Therefore, it is recommended that an SNMP
  agent be implemented by CEs and that the SNMP messages received by
  FEs be redirected to their CEs. AgentX framework defined in RFC
  2741 ([6]) may be applied here such that CEs act in the role of
  master agent to process SNMP protocol messages while FEs act in the
  role of subagent to provide access to the MIB objects residing on
  FEs. AgentX protocol messages between the master agent (CE) and
  the subagent (FE) are encapsulated and transported via ForCES, just
  like data packets from any other application layer protocols.

2. Design of ForCES MIB

In an NE composed of one or more FEs and a single CE, the CE is clearly aware of all associations and hence can provide this information in a single ForCES MIB. In contrast, in an NE composed of more than one CE, such association information is distributed and hence more than one ForCES MIB may be necessary, unless this information is aggregated into a single ForCES MIB by some means beyond the scope of this document. Nevertheless, the ForCES MIB design is compatible with both the single-CE and the multiple-CE case.

3. Association State

Association state as shown in the MIB is considered from the CE's point of view:
- An association is in the DOWN state if the CE has not received any message (heartbeat or other protocol message) from the FE within a given time period or if an Association Teardown message has been sent by the CE.
- An association is in the ESTABLISHING state as long as no message has been received from the FE after the CE has sent a positive Association Setup Response message.
- An association is in the UP state in all other cases.

Note that it is left to the implementers to choose how long entries of associations in the DOWN state remain in the MIB until they are removed, if at all.

The ForCES protocol may be used by the CE to query the FE Protocol LFB about some of the configuration parameters. However, such queries may obviously be issued only when the association is in the UP state. Hence any MIB value that corresponds to such a parameter can only be considered valid as long as the association is in the UP state.
[Note: there is no such parameter in the MIB at this time]

[Note: Should the MIB indicate whether associations have been rejected ? Can this be a weakness exploited by DDoS if the MIB lists all such rejected associations ?]

4. ForCES MIB Definition

For each association identified by the pair CE ID and FE ID, the following information is provided by the MIB:

- Current state of the association:

   DOWN: the CE(s) indicated by the CE ID and FE(s) indicated by the
FE ID are not associated.

 ESTABLISHING: transient state until the association has been
established. See [Section 3](#) above for details.

 UP: the CE(s) indicated by the CE ID and FE(s) indicated by the FE
ID are associated.

Association statistics:

- Time when the association attained the UP state.

- Time when the association appeared in the MIB.

- Number of transitions to ESTABLISHING state since the association
appeared in the MIB.

- Number of transitions to UP state since the association appeared in
the MIB.

- Number of ForCES messages sent/received since the association
attained the UP state.


```
    FORCES-MIB DEFINITIONS ::= BEGIN

    IMPORTS
        OBJECT-TYPE, MODULE-IDENTITY,
          Integer32, Counter32, Unsigned32
          FROM SNMPv2-SMI

        TEXTUAL-CONVENTION, RowStatus, TimeInterval, TimeStamp
          FROM SNMPv2-TC;

   forcesMIB MODULE-IDENTITY
        LAST-UPDATED "200601241200Z"  -- Jan 24, 2006
        ORGANIZATION "Forwarding and Control Element Separation
                    (ForCES) Working Group"
        CONTACT-INFO
            "Robert Haas (rha@zurich.ibm.com), IBM"
        DESCRIPTION
            "Initial version, published as RFC yyyy. This MIB
            contains managed object definitions for the ForCES
            Protocol."
-- RFC Ed.: replace yyyy with actual RFC number & remove this note
```

```
        ::= { mib-2 XXX }
-- RFC Ed.: replace XXX with IANA-assigned number & remove this note


--*************************************************************
   ForcesID ::= TEXTUAL-CONVENTION
       STATUS      current
       DESCRIPTION
           "The ForCES identifier is a four octet quantity."
       SYNTAX      OCTET STRING (SIZE (4))

   ForcesAssociationState ::= TEXTUAL-CONVENTION
       STATUS      current
       DESCRIPTION
             "The value down(1) indicates that the current state of
              the association is down. establishing(2) indicates
              that the association is in the process of being set
              up. up(3) indicates that the association is up."
       SYNTAX  INTEGER {
            down(1),
            establishing(2),
            up(3)
         }


   forcesAssociations    OBJECT IDENTIFIER ::= { forcesMIB 1 }

   forcesAssociationTable OBJECT-TYPE
       SYNTAX SEQUENCE OF ForcesAssociationEntry
       MAX-ACCESS not-accessible
       STATUS current
       DESCRIPTION
             "The (conceptual) table of associations."

       ::= { forcesAssociations 1 }

   forcesAssociationEntry OBJECT-TYPE
       SYNTAX ForcesAssociationEntry
       MAX-ACCESS not-accessible
       STATUS current
       DESCRIPTION
             "A (conceptual) entry for one association."
       INDEX { forcesAssociationCEID, forcesAssociationFEID }
       ::= { forcesAssociationTable 1 }

   ForcesAssociationEntry ::= SEQUENCE {
         forcesAssociationCEID            ForcesID,
         forcesAssociationFEID            ForcesID,
```

        forcesAssociationState   ForcesAssociationState,
        forcesAssociationUptime   TimeStamp,
        forcesAssociationCreated   TimeStamp,
        forcesAssociationTransitionsEstablishing Counter32,
        forcesAssociationTransitionsUp   Counter32,
        forcesAssociationMsgSent   Counter32,
        forcesAssociationMsgReceived   Counter32
    }

forcesAssociationCEID OBJECT-TYPE
    SYNTAX ForcesID
    MAX-ACCESS read-only
    STATUS current
    DESCRIPTION
            "The ForCES ID of the CE."
    ::= { forcesAssociationEntry 1 }

forcesAssociationFEID OBJECT-TYPE
    SYNTAX ForcesID
    MAX-ACCESS read-only
    STATUS current
    DESCRIPTION
            "The ForCES ID of the FE."
    ::= { forcesAssociationEntry 2 }

forcesAssociationState OBJECT-TYPE
     SYNTAX ForcesAssociationState
     MAX-ACCESS read-only
     STATUS current
     DESCRIPTION
            "The current operational state of the association
             described by this row of the table."
     ::= { forcesAssociationEntry 3 }

forcesAssociationUptime OBJECT-TYPE
    SYNTAX TimeStamp
    MAX-ACCESS read-only
    STATUS current
    DESCRIPTION
            "The time when this association came up."
    ::= { forcesAssociationEntry 4 }

forcesAssociationCreated OBJECT-TYPE
    SYNTAX TimeStamp
    MAX-ACCESS read-only
    STATUS current
    DESCRIPTION
            "The time when this entry in the table was

created for this association."

```
        ::= { forcesAssociationEntry 5 }

    forcesAssociationTransitionsEstablishing OBJECT-TYPE
        SYNTAX Counter32
        MAX-ACCESS read-only
        STATUS current
        DESCRIPTION
                "A counter of how many times this association
                 state changed from down to establishing."
        ::= { forcesAssociationEntry 6}

    forcesAssociationTransitionsUp OBJECT-TYPE
        SYNTAX Counter32
        MAX-ACCESS read-only
        STATUS current
        DESCRIPTION
                "A counter of how many times this association
                 state changed from establishing to up."
        ::= { forcesAssociationEntry 7}

    forcesAssociationMsgSent OBJECT-TYPE
        SYNTAX Counter32
        MAX-ACCESS read-only
        STATUS current
        DESCRIPTION
                "A counter of how many messages have been sent on
                 this association since it is up."
        ::= { forcesAssociationEntry 8}

    forcesAssociationMsgReceived OBJECT-TYPE
        SYNTAX Counter32
        MAX-ACCESS read-only
        STATUS current
        DESCRIPTION
                "A counter of how many messages have been received on
                 this association since it is up."
        ::= { forcesAssociationEntry 9}


    END
```

Security Considerations

    Some of the readable objects in this MIB module may be considered
    sensitive or vulnerable in some network environment.

    SNMP versions prior to SNMPv3 did not include adequate security.
    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 module.

   It is RECOMMENDED that implementers consider the security features as
   provided by the SNMPv3 framework (see [RFC3410], section 8),
   including full support for the SNMPv3 cryptographic mechanisms (for
   authentication and privacy).

   Further, deployment of SNMP versions prior to SNMPv3 is NOT
   RECOMMENDED. Instead, it is RECOMMENDED to deploy SNMPv3 and to
   enable cryptographic security. It is then a customer/operator
   responsibility to ensure that the SNMP entity giving access to an
   instance of this MIB module 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.

References

   [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate
   Requirements Levels", BCP 14, RFC 2119, March 1997.

   [RFC3654] Khosravi, H,, and Anderson, T., "Requirements for
   Separation of IP Control and Forwarding", RFC 3654, November 2003.

   [RFC3746] Yang, L., Dantu, R., Anderson, T., Gopal, R., "Forwarding
   and Control Element Separation (ForCES) Framework", RFC 3746, April
   2004.

   [RFC3410] Case, J., Mundy, R., Partain, D., and B. Stewart,
   "Introduction and Applicability Statements for Internet- Standard
   Management Framework", RFC 3410, December 2002.

Acknowledgments

   The author wants to acknowledge the comments of the members of the
   ForCES working group.

Author's Addresses

   Robert Haas
   IBM Research
   Zurich Research Lab
   Saeumerstrasse 4
   8803 Rueschlikon
   Switzerland
   Email: rha@zurich.ibm.com