

Internet Engineering Task Force (IETF)
Request for Comments: 7391
Updates: 5810, 7121
Category: Standards Track
ISSN: 2070-1721

J. Hadi Salim
Mojatatu Networks
October 2014

Forwarding and Control Element Separation (ForCES) Protocol Extensions

Abstract

Experience in implementing and deploying the Forwarding and Control Element Separation (ForCES) architecture has demonstrated the need for a few small extensions both to ease programmability and to improve wire efficiency of some transactions. The ForCES protocol is extended with a table range operation and a new extension for error handling. This document updates the semantics in RFCs 5810 and 7121 to achieve that end goal.

Status of This Memo

This is an Internet Standards Track document.

This document is a product of the Internet Engineering Task Force (IETF). It represents the consensus of the IETF community. It has received public review and has been approved for publication by the Internet Engineering Steering Group (IESG). Further information on Internet Standards is available in Section 2 of RFC 5741.

Information about the current status of this document, any errata, and how to provide feedback on it may be obtained at <http://www.rfc-editor.org/info/rfc7391>.

Copyright Notice

Copyright (c) 2014 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (<http://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

| | |
|---|----|
| 1. Introduction | 2 |
| 1.1. Terminology and Conventions | 3 |
| 1.1.1. Requirements Language | 3 |
| 1.1.2. Terminology | 3 |
| 2. Problem Overview | 4 |
| 2.1. Table Ranges | 4 |
| 2.2. Error Codes | 4 |
| 3. Protocol Update | 5 |
| 3.1. Table Ranges | 5 |
| 3.2. Error Codes | 6 |
| 3.2.1. New Codes | 7 |
| 3.2.2. Private Vendor Codes | 8 |
| 3.2.3. Extended Result TLV | 8 |
| 3.2.3.1. Extended Result Backward Compatibility | 9 |
| 3.3. Large Table Dumping | 9 |
| 4. IANA Considerations | 11 |
| 5. Security Considerations | 12 |
| 6. References | 12 |
| 6.1. Normative References | 12 |
| 6.2. Informative References | 12 |
| Appendix A. New FEPO Version | 13 |
| Acknowledgments | 23 |
| Author's Address | 23 |

1. Introduction

Experience in implementing and deploying the ForCES architecture has demonstrated the need for a few small extensions both to ease programmability and to improve wire efficiency of some transactions. This document describes a few extensions to the semantics in the ForCES protocol specification [RFC5810] to achieve that end goal.

This document describes and justifies the need for two small extensions that are backward compatible. This document also clarifies details of how dumping of a large table residing on an FE (Forwarding Element) is achieved. To summarize:

1. A table range operation to allow a controller or control application to request an arbitrary range of table rows is introduced.
2. Additional error codes returned to the controller (or control application) by an FE are introduced. Additionally, a new extension to carry details on error codes is introduced. As a result, this document updates the definition of the FE Protocol Object (FEPO) Logical Functional Block (LFB) in [RFC7121].

3. While already supported, an FE response to a GET request of a large table that does not fit in a single Protocol Layer (PL) message is not described in [RFC5810]. This document clarifies the details.

1.1. Terminology and Conventions

1.1.1. Requirements Language

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 [RFC2119].

1.1.2. Terminology

This document reiterates the terminology defined in several ForCES documents ([RFC3746], [RFC5810], [RFC5811], and [RFC5812]) for the sake of contextual clarity.

Control Element (CE)

Forwarding Element (FE)

FE Model

LFB (Logical Functional Block) Class (or type)

LFB Instance

LFB Model

LFB Metadata

ForCES Component

LFB Component

ForCES Protocol Layer (ForCES PL)

ForCES Protocol Transport Mapping Layer (ForCES TML)

2. Problem Overview

In this section, we present sample use cases to illustrate each challenge being addressed.

2.1. Table Ranges

Consider, for the sake of illustration, an FE table with 1 million reasonably sized table rows that are sparsely populated. Assume, again for the sake of illustration, that there are 2000 table rows sparsely populated between the row indices 23-10023.

Implementation experience has shown that existing approaches for retrieving or deleting a sizable number of table rows are both programmatically tedious and inefficient on utilization of both compute and wire resources.

By definition, ForCES GET and DEL requests sent from a controller (or control application) are prepended with a path to a component and sent to the FE. In the case of indexed tables, the component path can point to either a table or a table row index.

As an example, a control application attempting to retrieve the first 2000 table rows appearing between row indices 23 and 10023 can achieve its goal in one of the following ways:

- o Dump the whole table and filter for the needed 2000 table rows.
- o Send up to 10000 ForCES PL requests, incrementing the index by one each time, and stop when the needed 2000 entries are retrieved.
- o If the application had knowledge of which table rows existed (not unreasonable given the controller is supposed to be aware of state within a Network Element (NE)), then the application could take advantage of ForCES batching to send fewer large messages (each with different path entries for a total of 2000).

As argued, while the above options exist, all are tedious.

2.2. Error Codes

[RFC5810] has defined a generic set of error codes that are to be returned to the CE from an FE. Deployment experience has shown that it would be useful to have more fine-grained error codes. As an example, the error code E_NOT_SUPPORTED could be mapped to many FE error source possibilities that need to then be interpreted by the caller based on some understanding of the nature of the sent request. This makes debugging more time consuming.

3. Protocol Update

This section describes a normative update to the ForCES protocol to address the issues discussed in Section 2.

3.1. Table Ranges

We define a new TLV, TABLERANGE-TLV (type ID 0x0117), that will be associated with the PATH-DATA-TLV in the same manner the KEYINFO-TLV is. Figure 1 shows how this new TLV is constructed.

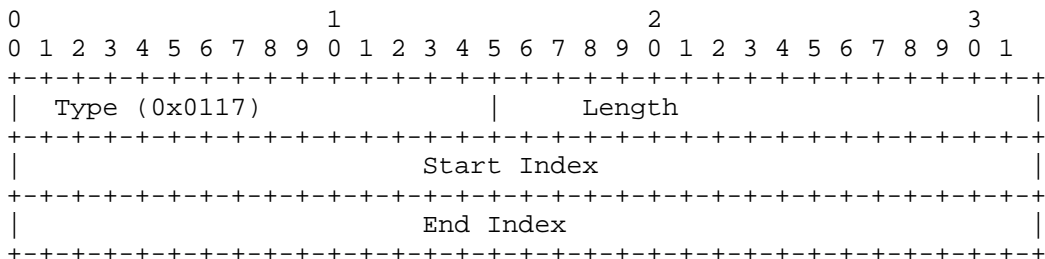


Figure 1: ForCES Table Range Request Layout

Figure 2 illustrates a GET request for a range of rows 11 to 23 of a table with a component path of "1/6".

```

OPER = GET-TLV
PATH-DATA-TLV:
  flags = F_SELTABRANGE, IDCount = 2, IDs = {1,6}
  TABLERANGE-TLV content = {11,23}

```

Figure 2: ForCES Table Range Request Example

The path flag F_SELTABRANGE (0x2, i.e., bit 1, where bit 0 is F_SELKEY as defined in [RFC5810]) MUST be set to indicate the presence of the TABLERANGE-TLV. The path flag bit F_SELTABRANGE can only be used in a GET or DEL and is mutually exclusive with F_SELKEY. The FE MUST enforce the path flag constraints and ensure that the selected path belongs to a defined, indexed table component. Any violation of these constraints MUST be rejected with an error code of E_INVALID_TFLAGS with a description of what the problem is when using extended error reporting (refer to Section 3.2).

It should be noted that there are combinations of path selection mechanisms that should not appear together for the sake of simplicity of operations. These include TABLERANGE-TLV and KEYINFO-TLV as well as multiple nested TABLERANGE-TLVs.

The TABLERANGE-TLV contents constitute:

- o A 32-bit start index. An index of 0 implies the beginning of the table row.
- o A 32-bit end index. A value of 0xFFFFFFFF implies the last entry.

The response for a table range query will either be:

- o The requested table data returned (when at least one referenced row is available); in such a case, a response with a path pointing to the table and whose data content contains the row(s) will be sent to the CE. The data content MUST be encapsulated in a SPARSEDATA-TLV. The SPARSEDATA-TLV content will have the "I" (in Index-Length-Value (ILV)) for each table row indicating the table indices.
- o An EXTENDEDRESULT-TLV (refer to Section 3.2.3) when:
 - * the response is to a range delete request. The result will either be:
 - + a success if any of the rows that were requested are deleted; or
 - + a proper error code if none of the rows that were requested can be deleted.
 - * data is absent and an error code of E_EMPTY with an optional content string describing the nature of the error is used (refer to Section 3.2).
 - * both a path key and path table range were stated on the path flags of the original request. In such a case, an error code of E_INVALID_TFLAGS with an optional content string describing the nature of the error is used (refer to Section 3.2).
 - * other standard ForCES errors (such as Access Control List (ACL) constraints trying to retrieve contents of an unreadable table, accessing unknown components, etc.) occur.

3.2. Error Codes

We define the following:

1. A new set of error codes.
2. Allocation of some reserved codes for private use.

3. A new TLV, EXTENDEDRESULT-TLV (0x0118), that will carry a code (which will be a superset of what is currently specified in [RFC5810]) as well as an optional cause content. This is illustrated in Figure 3.

3.2.1. New Codes

The EXTENDEDRESULT-TLV Result Value is 32 bits and is a superset of the RESULT-TLV Result Value defined in [RFC5810]. The new version code space is 32 bits as opposed to the code size of 8 bits in [RFC5810]. The first 8-bit values (256 codes) are common to both code spaces.

| Code | Mnemonic | Details |
|------|-------------------------|---|
| 0x18 | E_TIMED_OUT | A timeout occurred while processing the message |
| 0x19 | E_INVALID_TFLAGS | Invalid table flags |
| 0x1A | E_INVALID_OP | Requested operation is invalid |
| 0x1B | E_CONGEST_NT | Node congestion notification |
| 0x1C | E_COMPONENT_NOT_A_TABLE | Component not a table |
| 0x1D | E_PERM | Operation not permitted |
| 0x1E | E_BUSY | System is busy |
| 0x1F | E_EMPTY | Table is empty |
| 0x20 | E_UNKNOWN | A generic catch-all error code. Carries a string to further extrapolate what the error implies. |

Table 1: New Codes

3.2.2. Private Vendor Codes

Codes 0x100-0x200 are reserved for use as private codes. Since these are freely available, it is expected that the FE and CE side implementations will both understand/interpret the semantics of any used codes and avoid any conflicts.

3.2.3. Extended Result TLV

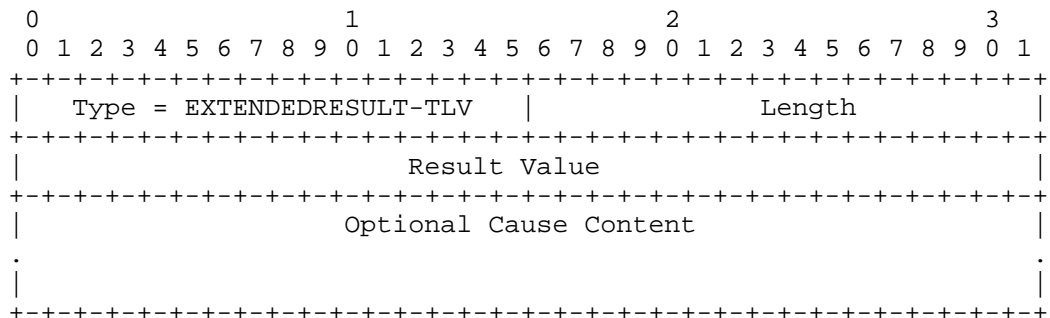


Figure 3: EXTENDEDRESULT-TLV

- o Like all other ForCES TLVs, the EXTENDEDRESULT-TLV is expected to be 32-bit aligned.
- o The EXTENDEDRESULT-TLV Result Value derives and extends from the same current namespace that is used by the RESULT-TLV Result Value as specified in Section 7.1.7 of [RFC5810]. The main difference is that there is now a 32-bit Result Value (as opposed to the old 8-bit).
- o The Optional Cause Content is defined to further disambiguate the Result Value. It is expected that UTF-8 string values will be used. The content Result Value is intended to be consumed by the (human) operator, and implementations may choose to specify different content for the same error code. Additionally, future codes may specify cause content to be of types other than string.
- o It is recommended that the maximum size of the cause string should not exceed 32 bytes. The cause string is not standardized by this document.

3.2.3.1. Extended Result Backward Compatibility

To support backward compatibility, we update the FEPO LFB (in Appendix A) to version 1.2. We also add a new component ID 16 (named EResultAdmin), and a capability component ID 32 (named EResultCapab).

An FE will advertise its capability to support extended TLVs via the EResultCapab table. When an FE is capable of responding with both extended results and older result TLVs, it will have two table rows, one for each supported value. By default, an FE capable of supporting both modes will assume the lowest common denominator (i.e., EResultAdmin will be EResultNotSupported) and will issue responses using RESULT-TLVs. It should be noted that an FE advertising FEPO version 1.2 MUST support EXTENDEDRESULT-TLVs at minimum.

On an FE that supports both RESULT-TLVs and EXTENDEDRESULT-TLVs, a master CE can turn on support for extended results by setting the EResultAdmin value to 2, in which case the FE MUST switch over to sending only EXTENDEDRESULT-TLVs. Likewise, a master CE can turn off extended result responses by writing a 1 to the EResultAdmin. An FE that does not support one mode or the other MUST reject setting EResultAdmin to a value it does not support by responding with an error code of E_NOT_SUPPORTED. It is expected that all CEs participating in a high availability (HA) mode be capable of supporting FEPO version 1.2 whenever EResultAdmin is set to strict support of EXTENDEDRESULT-TLVs. The consensus between CEs in an HA set up to set strict support of EXTENDEDRESULT-TLVs is out of scope for this document.

3.3. Large Table Dumping

Imagine a GET request to a path that is a table, i.e., a table dump. Such a request is sent to the FE with a specific correlator, say X. Imagine this table to have a large number of entries at the FE. For the sake of illustration, let's say millions of rows. This requires that the FE delivers the response over multiple messages, all using the same correlator X.

The ForCES protocol document [RFC5810] does not adequately describe how a large multi-part GET response message is delivered; the text in this section clarifies. We limit the discussion to a table object only.

Implementation experience of dumping large tables shows that we can use transaction flags to indicate that a GET response is the beginning, middle, or end of a multi-part message. In other words, we mirror the effect of an atomic transaction sent by a CE to an FE.

CE PL

FE PL

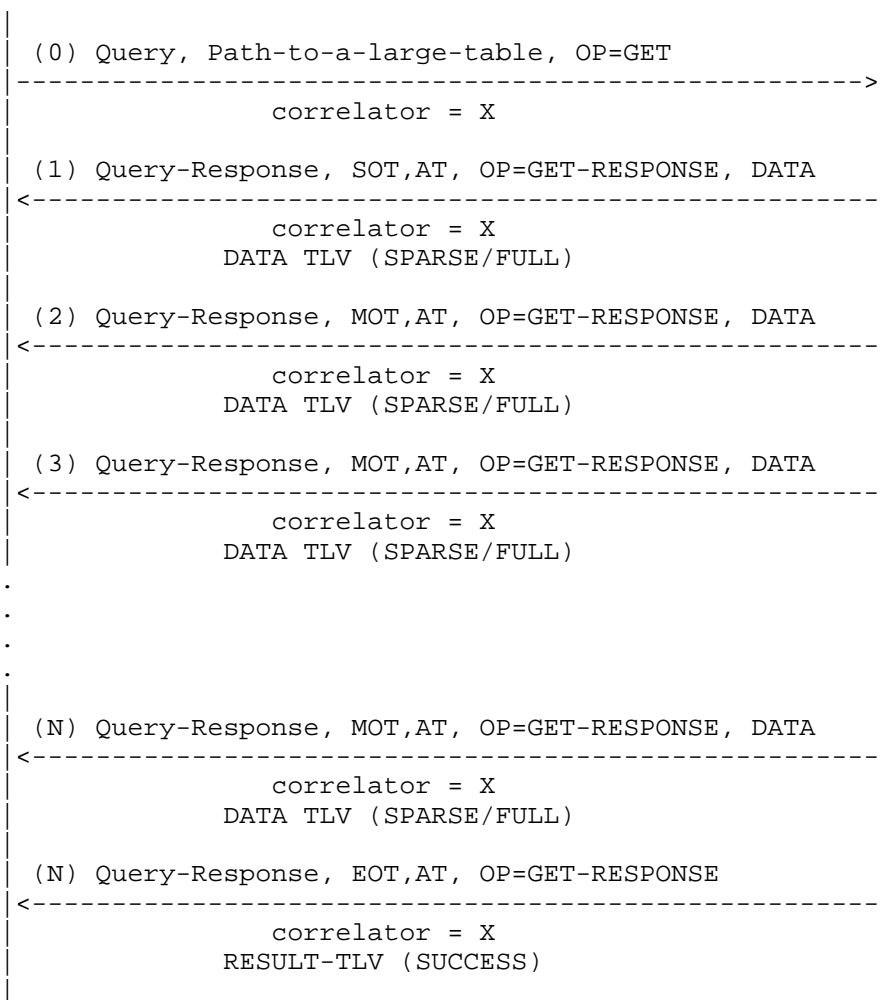


Figure 4: Large Table Dump Time Sequence

The last message to go to the CE, which carries the End Of Transaction (EOT) flag, MUST NOT carry any data. This allows us to mirror ForCES two-phase commit (2PC) messaging [RFC5810] where the last message is an empty commit message. A GET response will carry a RESULT-TLV in such a case.

4. IANA Considerations

This document updates <<https://www.iana.org/assignments/forces>> as follows:

This document registers two new top-level TLVs and two new path flags; it also updates an IANA-registered FE Protocol Object Logical Functional Block (LFB).

Appendix A defines an update to the FE Protocol Object LFB to version 1.2. An entry for FE Protocol Object LFB version 1.2 has been added to the "Logical Functional Block (LFB) Class Names and Class Identifiers" sub-registry.

The following new TLVs have been defined and added to the "TLV Types" sub-registry:

- o TABLERANGE-TLV (type ID 0x0117)
- o EXTENDEDRESULT-TLV (type ID 0x0118)

The "RESULT-TLV Result Values" sub-registry has been updated as follows:

- o Codes 0x21-0xFE are marked as Unassigned.
- o Codes 0x18-0x20 are defined by this document in Section 3.2.1.
- o Codes 0x100-0x200 are reserved for private use.

A new "EXTENDEDRESULT-TLV Result Values" sub-registry has been created. The codes 0x00-0xFF are mirrored from the "RESULT-TLV Result Values" sub-registry. Any future allocations of this code range (in the range 0x21-0xFE) must be made only in the new "EXTENDEDRESULT-TLV Result Values" sub-registry and not in the "RESULT-TLV Result Values" sub-registry. The codes 0x100-0x200 are reserved for private use as described earlier, and the code ranges 0x21-0xFE and 0x201-0xFFFFFFFF are marked as Unassigned with the IANA allocation policy of Specification Required [RFC5226]. The Designated Expert (DE) needs to ensure that existing deployments are not broken by any specified request. The DE should post a given code request to the ForCES WG mailing list (or a successor designated by the Area Director) for comment and review. The DE should then either approve or deny the registration request, publish a notice of the decision to the ForCES WG mailing list or its successor, and inform IANA of his/her decision. A denial notice must be justified by an

explanation and, in the cases where it is possible, concrete suggestions on how the request can be modified so as to become acceptable.

5. Security Considerations

The security considerations described in the ForCES protocol [RFC5810] apply to this document as well.

6. References

6.1. Normative References

- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997, <<http://www.rfc-editor.org/info/rfc2119>>.
- [RFC5226] Narten, T. and H. Alvestrand, "Guidelines for Writing an IANA Considerations Section in RFCs", BCP 26, RFC 5226, May 2008, <<http://www.rfc-editor.org/info/rfc5226>>.
- [RFC5810] Doria, A., Hadi Salim, J., Haas, R., Khosravi, H., Wang, W., Dong, L., Gopal, R., and J. Halpern, "Forwarding and Control Element Separation (ForCES) Protocol Specification", RFC 5810, March 2010, <<http://www.rfc-editor.org/info/rfc5810>>.
- [RFC5811] Hadi Salim, J. and K. Ogawa, "SCTP-Based Transport Mapping Layer (TML) for the Forwarding and Control Element Separation (ForCES) Protocol", RFC 5811, March 2010, <<http://www.rfc-editor.org/info/rfc5811>>.
- [RFC5812] Halpern, J. and J. Hadi Salim, "Forwarding and Control Element Separation (ForCES) Forwarding Element Model", RFC 5812, March 2010, <<http://www.rfc-editor.org/info/rfc5812>>.
- [RFC7121] Ogawa, K., Wang, W., Haleplidis, E., and J. Hadi Salim, "High Availability within a Forwarding and Control Element Separation (ForCES) Network Element", RFC 7121, February 2014, <<http://www.rfc-editor.org/info/rfc7121>>.

6.2. Informative References

- [RFC3746] Yang, L., Dantu, R., Anderson, T., and R. Gopal, "Forwarding and Control Element Separation (ForCES) Framework", RFC 3746, April 2004, <<http://www.rfc-editor.org/info/rfc3746>>.

Appendix A. New FEPO Version

This version of FEPO updates the earlier one given in [RFC7121]. The XML has been validated against the schema defined in [RFC5812].

```
<LFBLibrary xmlns="urn:ietf:params:xml:ns:forces:lfbmodel:1.0"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="lfb-schema.xsd" provides="FEPO">
  <!-- XXX -->
  <dataTypeDefs>
    <dataTypeDef>
      <name>CEHBPolyValues</name>
      <synopsis>
        The possible values of CE heartbeat policy
      </synopsis>
      <atomic>
        <baseType>uchar</baseType>
        <specialValues>
          <specialValue value="0">
            <name>CEHBPoly0</name>
            <synopsis>
              The CE will send heartbeats to the FE
              every CEHDI timeout if no other messages
              have been sent since.
            </synopsis>
          </specialValue>
          <specialValue value="1">
            <name>CEHBPoly1</name>
            <synopsis>
              The CE will not send heartbeats to the FE.
            </synopsis>
          </specialValue>
        </specialValues>
      </atomic>
    </dataTypeDef>
    <dataTypeDef>
      <name>FEHBPolyValues</name>
      <synopsis>
        The possible values of FE heartbeat policy
      </synopsis>
      <atomic>
        <baseType>uchar</baseType>
        <specialValues>
          <specialValue value="0">
            <name>FEHBPoly0</name>
            <synopsis>
              The FE will not generate any heartbeats to the CE.
            </synopsis>
          </specialValue>
        </specialValues>
      </atomic>
    </dataTypeDef>
  </dataTypeDefs>
```

```

    </specialValue>
    <specialValue value="1">
      <name>FEHBPolicy1</name>
      <synopsis>
        The FE generates heartbeats to the CE every
        FEHI if no other
        messages have been sent to the CE.
      </synopsis>
    </specialValue>
  </specialValues>
</atomic>
</dataTypeDef>
<dataTypeDef>
  <name>FERestartPolicyValues</name>
  <synopsis>
    The possible values of FE restart policy
  </synopsis>
  <atomic>
    <baseType>uchar</baseType>
    <specialValues>
      <specialValue value="0">
        <name>FERestartPolicy0</name>
        <synopsis>
          The FE restarts its state from scratch.
        </synopsis>
      </specialValue>
    </specialValues>
  </atomic>
</dataTypeDef>
<dataTypeDef>
  <name>HAModeValues</name>
  <synopsis>
    The possible values of HA modes
  </synopsis>
  <atomic>
    <baseType>uchar</baseType>
    <specialValues>
      <specialValue value="0">
        <name>NoHA</name>
        <synopsis>
          The FE is not running in HA mode.
        </synopsis>
      </specialValue>
      <specialValue value="1">
        <name>ColdStandby</name>
        <synopsis>
          The FE is running in HA mode cold standby.
        </synopsis>
      </specialValue>
    </specialValues>
  </atomic>
</dataTypeDef>

```

```

        </specialValue>
        <specialValue value="2">
            <name>HotStandby</name>
            <synopsis>
                The FE is running in HA mode hot standby.
            </synopsis>
        </specialValue>
    </specialValues>
</atomic>
</dataTypeDef>
<dataTypeDef>
    <name>CEFailoverPolicyValues</name>
    <synopsis>
        The possible values of CE failover policy
    </synopsis>
    <atomic>
        <baseType>uchar</baseType>
        <specialValues>
            <specialValue value="0">
                <name>CEFailoverPolicy0</name>
                <synopsis>
                    The FE should stop functioning immediately
                    and transition to FE OperDisable state.
                </synopsis>
            </specialValue>
            <specialValue value="1">
                <name>CEFailoverPolicy1</name>
                <synopsis>
                    The FE should continue forwarding even
                    without an associated CE for CEFTI. The
                    FE goes to FE OperDisable when the CEFTI
                    expires and there is no association. Requires
                    graceful restart support.
                </synopsis>
            </specialValue>
        </specialValues>
    </atomic>
</dataTypeDef>
<dataTypeDef>
    <name>FEHACapab</name>
    <synopsis>
        The supported HA features
    </synopsis>
    <atomic>
        <baseType>uchar</baseType>
        <specialValues>
            <specialValue value="0">
                <name>GracefullRestart</name>

```

```

        <synopsis>
            The FE supports graceful restart.
        </synopsis>
    </specialValue>
<specialValue value="1">
    <name>HA</name>
    <synopsis>
        The FE supports HA.
    </synopsis>
</specialValue>
</specialValues>
</atomic>
</dataTypeDef>
<dataTypeDef>
    <name>CEStatusType</name>
    <synopsis>Status values.  Status for each CE</synopsis>
    <atomic>
        <baseType>uchar</baseType>
        <specialValues>
            <specialValue value="0">
                <name>Disconnected</name>
                <synopsis>No connection attempt with the CE yet
            </synopsis>
            </specialValue>
            <specialValue value="1">
                <name>Connected</name>
                <synopsis>The FE connection with the CE at the TML
                    has been completed.
            </synopsis>
            </specialValue>
            <specialValue value="2">
                <name>Associated</name>
                <synopsis>The FE has associated with the CE.
            </synopsis>
            </specialValue>
            <specialValue value="3">
                <name>IsMaster</name>
                <synopsis>The CE is the master (and associated).
            </synopsis>
            </specialValue>
            <specialValue value="4">
                <name>LostConnection</name>
                <synopsis>The FE was associated with the CE but
                    lost the connection.
            </synopsis>
            </specialValue>
            <specialValue value="5">
                <name>Unreachable</name>

```



```
        <synopsis>The CE is deemed as unreachable by the FE.
      </synopsis>
    </specialValue>
  </specialValues>
</atomic>
</dataTypeDef>
<dataTypeDef>
  <name>StatisticsType</name>
  <synopsis>Statistics Definition</synopsis>
  <struct>
    <component componentID="1">
      <name>RecvPackets</name>
      <synopsis>Packets received</synopsis>
      <typeRef>uint64</typeRef>
    </component>
    <component componentID="2">
      <name>RecvErrPackets</name>
      <synopsis>Packets received from CE with errors
      </synopsis>
      <typeRef>uint64</typeRef>
    </component>
    <component componentID="3">
      <name>RecvBytes</name>
      <synopsis>Bytes received from CE</synopsis>
      <typeRef>uint64</typeRef>
    </component>
    <component componentID="4">
      <name>RecvErrBytes</name>
      <synopsis>Bytes received from CE in error</synopsis>
      <typeRef>uint64</typeRef>
    </component>
    <component componentID="5">
      <name>TxmitPackets</name>
      <synopsis>Packets transmitted to CE</synopsis>
      <typeRef>uint64</typeRef>
    </component>
    <component componentID="6">
      <name>TxmitErrPackets</name>
      <synopsis>
        Packets transmitted to CE that incurred
        errors
      </synopsis>
      <typeRef>uint64</typeRef>
    </component>
    <component componentID="7">
      <name>TxmitBytes</name>
      <synopsis>Bytes transmitted to CE</synopsis>
      <typeRef>uint64</typeRef>
    </component>
  </struct>
</dataTypeDef>
```

```

    </component>
    <component componentID="8">
      <name>TxmitErrBytes</name>
      <synopsis>Bytes transmitted to CE incurring errors
      </synopsis>
      <typeRef>uint64</typeRef>
    </component>
  </struct>
</dataTypeDef>
<dataTypeDef>
  <name>AllCEType</name>
  <synopsis>Table Type for AllCE component</synopsis>
  <struct>
    <component componentID="1">
      <name>CEID</name>
      <synopsis>ID of the CE</synopsis>
      <typeRef>uint32</typeRef>
    </component>
    <component componentID="2">
      <name>Statistics</name>
      <synopsis>Statistics per CE</synopsis>
      <typeRef>StatisticsType</typeRef>
    </component>
    <component componentID="3">
      <name>CEStatus</name>
      <synopsis>Status of the CE</synopsis>
      <typeRef>CEStatusType</typeRef>
    </component>
  </struct>
</dataTypeDef>
<dataTypeDef>
  <name>ExtendedResultType</name>
  <synopsis>
    Possible extended result support
  </synopsis>
  <atomic>
    <baseType>uchar</baseType>
    <rangeRestriction>
      <allowedRange min="1" max="2"/>
    </rangeRestriction>
    <specialValues>
      <specialValue value="1">
        <name>EResultNotSupported</name>
        <synopsis>
          Extended results are not supported.
        </synopsis>
      </specialValue>
      <specialValue value="2">

```

```

        <name>EResultSupported</name>
        <synopsis>
            Extended results are supported.
        </synopsis>
    </specialValue>
</specialValues>
</atomic>
</dataTypeDef>
</dataTypeDefs>
<LFBClassDefs>
    <LFBClassDef LFBClassID="2">
        <name>FEPO</name>
        <synopsis>
            The FE Protocol Object, with extended result control
        </synopsis>
        <version>1.2</version>
        <components>
            <component componentID="1" access="read-only">
                <name>CurrentRunningVersion</name>
                <synopsis>Currently running ForCES version</synopsis>
                <typeRef>uchar</typeRef>
            </component>
            <component componentID="2" access="read-only">
                <name>FEID</name>
                <synopsis>Unicast FEID</synopsis>
                <typeRef>uint32</typeRef>
            </component>
            <component componentID="3" access="read-write">
                <name>MulticastFEIDs</name>
                <synopsis>
                    The table of all multicast IDs
                </synopsis>
                <array type="variable-size">
                    <typeRef>uint32</typeRef>
                </array>
            </component>
            <component componentID="4" access="read-write">
                <name>CEHBPolicy</name>
                <synopsis>
                    The CE Heartbeat Policy
                </synopsis>
                <typeRef>CEHBPolicyValues</typeRef>
            </component>
            <component componentID="5" access="read-write">
                <name>CEHDI</name>
                <synopsis>
                    The CE Heartbeat Dead Interval in milliseconds
                </synopsis>

```

```

    <typeRef>uint32</typeRef>
  </component>
  <component componentID="6" access="read-write">
    <name>FEHBPolicy</name>
    <synopsis>
      The FE Heartbeat Policy
    </synopsis>
    <typeRef>FEHBPolicyValues</typeRef>
  </component>
  <component componentID="7" access="read-write">
    <name>FEHI</name>
    <synopsis>
      The FE Heartbeat Interval in milliseconds
    </synopsis>
    <typeRef>uint32</typeRef>
  </component>
  <component componentID="8" access="read-write">
    <name>CEID</name>
    <synopsis>
      The Primary CE this FE is associated with
    </synopsis>
    <typeRef>uint32</typeRef>
  </component>
  <component componentID="9" access="read-write">
    <name>BackupCEs</name>
    <synopsis>
      The table of all backup CEs other than the
      primary
    </synopsis>
    <array type="variable-size">
      <typeRef>uint32</typeRef>
    </array>
  </component>
  <component componentID="10" access="read-write">
    <name>CEFailoverPolicy</name>
    <synopsis>
      The CE Failover Policy
    </synopsis>
    <typeRef>CEFailoverPolicyValues</typeRef>
  </component>
  <component componentID="11" access="read-write">
    <name>CEFTI</name>
    <synopsis>
      The CE Failover Timeout Interval in milliseconds
    </synopsis>
    <typeRef>uint32</typeRef>
  </component>
  <component componentID="12" access="read-write">

```

```

    <name>FERestartPolicy</name>
    <synopsis>
      The FE Restart Policy
    </synopsis>
    <typeRef>FERestartPolicyValues</typeRef>
  </component>
  <component componentID="13" access="read-write">
    <name>LastCEID</name>
    <synopsis>
      The Primary CE this FE was last associated
      with
    </synopsis>
    <typeRef>uint32</typeRef>
  </component>
  <component componentID="14" access="read-write">
    <name>HAMode</name>
    <synopsis>
      The HA mode used
    </synopsis>
    <typeRef>HAModeValues</typeRef>
  </component>
  <component componentID="15" access="read-only">
    <name>AllCEs</name>
    <synopsis>The table of all CEs</synopsis>
    <array type="variable-size">
      <typeRef>AllCEType</typeRef>
    </array>
  </component>
  <component componentID="16" access="read-write">
    <name>EResultAdmin</name>
    <synopsis>
      Turn extended results off or on,
      but default to off.
    </synopsis>
    <typeRef>ExtendedResultType</typeRef>
    <defaultValue>1</defaultValue>
  </component>
</components>
<capabilities>
  <capability componentID="30">
    <name>SupportableVersions</name>
    <synopsis>
      The table of ForCES versions that FE supports
    </synopsis>
    <array type="variable-size">
      <typeRef>uchar</typeRef>
    </array>
  </capability>

```

```

<capability componentID="31">
  <name>HACapabilities</name>
  <synopsis>
    The table of HA capabilities the FE supports
  </synopsis>
  <array type="variable-size">
    <typeRef>FEHACapab</typeRef>
  </array>
</capability>
<capability componentID="32">
  <name>EResultCapab</name>
  <synopsis>
    The table of supported result capabilities
  </synopsis>
  <array type="variable-size">
    <typeRef>ExtendedResultType</typeRef>
  </array>
</capability>
</capabilities>
<events baseID="61">
  <event eventID="1">
    <name>PrimaryCEDown</name>
    <synopsis>
      The primary CE has changed.
    </synopsis>
    <eventTarget>
      <eventField>LastCEID</eventField>
    </eventTarget>
    <eventChanged/>
    <eventReports>
      <eventReport>
        <eventField>LastCEID</eventField>
      </eventReport>
    </eventReports>
  </event>
  <event eventID="2">
    <name>PrimaryCEChanged</name>
    <synopsis>A new primary CE has been selected.
    </synopsis>
    <eventTarget>
      <eventField>CEID</eventField>
    </eventTarget>
    <eventChanged/>
    <eventReports>
      <eventReport>
        <eventField>CEID</eventField>
      </eventReport>
    </eventReports>
  </event>
</events>

```

```
        </event>
      </events>
    </LFBClassDef>
  </LFBClassDefs>
</LFBLibrary>
```

Acknowledgments

The author would like to thank Evangelos Haleplidis and Joel Halpern for discussions that made this document better. Adrian Farrel did an excellent AD review of the document, which improved the quality of this document. Tobias Gondrom did the Security Directorate review. Brian Carpenter did the Gen-ART review. Nevil Brownlee performed the Operations Directorate review. S. Moonesamy (SM) worked hard to review our publication process. Pearl Liang caught issues in the IANA text.

The author would like to thank the following IESG members who reviewed and improved this document: Alia Atlas, Barry Leiba, Brian Haberman, Kathleen Moriarty, Richard Barnes, and Spencer Dawkins.

Author's Address

Jamal Hadi Salim
Mojatatu Networks
Suite 400, 303 Moodie Dr.
Ottawa, Ontario K2H 9R4
Canada

EMail: hadi@mojatatu.com