| Internet Engineering Task Force | E. Haleplidis | |
| --- | --- | --- |
| Internet-Draft | University of Patras | |
| Intended status: Informational | K. Ogawa | |
| Expires: August 29, 2010 | NTT Corporation | |
| | W. Wang | |
| | Zhejiang Gongshang University | |
| | J. Hadi Salim | |
| | Mojatatu Networks | |
| | February 25, 2010 | |

**Implementation Report for ForCES**
**draft-ietf-forces-implementation-report-01**

**Abstract**

Forwarding and Control Element Separation (ForCES) defines an architectural framework and associated protocols to standardize information exchange between the control plane and the forwarding plane in a ForCES Network Element (ForCES NE). RFC3654 has defined the ForCES requirements, and RFC3746 has defined the ForCES framework.
This document is an implementation report of the ForCES Protocol, Model and SCTP-TML, including the report on interoperability testing and the current state of ForCES implementations.

**Status of this Memo**

---

**Table of Contents**

---

## 1.  Terminology and Conventions

---

## 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] (Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels," March 1997.).

---

## 1.2.  Definitions

This document follows the terminology defined by the ForCES Requirements in [RFC3654] (Khosravi, H. and T. Anderson, "Requirements for Separation of IP Control and Forwarding," November 2003.) and by the ForCES framework in [RFC3746] (Yang, L., Dantu, R., Anderson, T., and R. Gopal, "Forwarding and Control Element Separation (ForCES) Framework," April 2004.). The definitions below are repeated below for clarity.

> Control Element (CE) - A logical entity that implements the ForCES protocol and uses it to instruct one or more FEs on how to

process packets. CEs handle functionality such as the execution of control and signaling protocols.

Forwarding Element (FE) - A logical entity that implements the ForCES protocol. FEs use the underlying hardware to provide per-packet processing and handling as directed/controlled by one or more CEs via the ForCES protocol.

LFB (Logical Function Block) - The basic building block that is operated on by the ForCES protocol. The LFB is a well defined, logically separable functional block that resides in an FE and is controlled by the CE via ForCES protocol. The LFB may reside at the FE's datapath and process packets or may be purely an FE control or configuration entity that is operated on by the CE. Note that the LFB is a functionally accurate abstraction of the FE's processing capabilities, but not a hardware-accurate representation of the FE implementation.

LFB Class and LFB Instance - LFBs are categorized by LFB Classes. An LFB Instance represents an LFB Class (or Type) existence. There may be multiple instances of the same LFB Class (or Type) in an FE. An LFB Class is represented by an LFB Class ID, and an LFB Instance is represented by an LFB Instance ID. As a result, an LFB Class ID associated with an LFB Instance ID uniquely specifies an LFB existence.

LFB Metadata - Metadata is used to communicate per-packet state from one LFB to another, but is not sent across the network. The FE model defines how such metadata is identified, produced and consumed by the LFBs. It defines the functionality but not how metadata is encoded within an implementation.

LFB Components - Operational parameters of the LFBs that must be visible to the CEs are conceptualized in the FE model as the LFB components. The LFB components include, for example, flags, single parameter arguments, complex arguments, and tables that the CE can read and/or Components write via the ForCES protocol (see below).

ForCES Protocol - While there may be multiple protocols used within the overall ForCES architecture, the term "ForCES protocol" and "protocol" refer to the Fp reference points in the ForCES Framework in [RFC3746]. This protocol does not apply to CE-to-CE communication, FE-to-FE communication, or to communication between FE and CE managers. Basically, the ForCES protocol works in a master- slave mode in which FEs are slaves

> and CEs are masters. This document defines the specifications for
> this ForCES protocol.
>
> ForCES Protocol Transport Mapping Layer (ForCES TML) - A layer in
> ForCES protocol architecture that uses the capabilities of
> existing transport protocols to specifically address protocol
> message transportation issues, such as how the protocol messages
> are mapped to different transport media (like TCP, IP, ATM,
> Ethernet, etc), and how to achieve and implement reliability,
> multicast, ordering, etc. The ForCES TML specifications are
> detailed in separate ForCES documents, one for each TML.

---

## 2.  Introduction

This is an implementation report for the ForCES protocol, model and
SCTP-TML documents and includes an interoperability report.
It follows the outline suggested by [RFC5657] (Dusseault, L. and R.
Sparks, "Guidance on Interoperation and Implementation Reports for
Advancement to Draft Standard," September 2009.).
Forwarding and Control Element Separation (ForCES) defines an
architectural framework and associated protocols to standardize
information exchange between the control plane and the forwarding plane
in a ForCES Network Element (ForCES NE). [RFC3654] (Khosravi, H. and T.
Anderson, "Requirements for Separation of IP Control and Forwarding,"
November 2003.) has defined the ForCES requirements, and [RFC3746]
(Yang, L., Dantu, R., Anderson, T., and R. Gopal, "Forwarding and
Control Element Separation (ForCES) Framework," April 2004.) has
defined the ForCES framework.

---

## 2.1.  ForCES Protocol

The ForCES protocol works in a master-slave mode in which FEs are
slaves and CEs are masters. The protocol includes commands for
transport of Logical Function Block (LFB) configuration information,
association setup, status, and event notifications, etc. The reader is
encouraged to read FE-protocol (Dong, L., Doria, A., Gopal, R., HAAS,
R., Salim, J., Khosravi, H., and W. Wang, "ForCES Protocol
Specification," March 2009.) [I-D.ietf-forces-protocol] for further
information.

---

## 2.2. ForCES Model

The FE-MODEL (Halpern, J. and J. Salim, "ForCES Forwarding Element Model," October 2008.) [I-D.ietf-forces-model] presents a formal way to define FE Logical Function Blocks (LFBs) using XML. LFB configuration components, capabilities, and associated events are defined when the LFB is formally created. The LFBs within the FE are accordingly controlled in a standardized way by the ForCES protocol.

---

## 2.3. Transport mapping layer

The TML transports the PL messages. The TML is where the issues of how to achieve transport level reliability, congestion control, multicast, ordering, etc. are handled. All ForCES Protocol Layer implementations MUST be portable across all TMLs. Although more than one TML may be standardized for the ForCES Protocol, all implementations MUST IMPLEMENT the SCTP TML (Salim, J. and K. Ogawa, "SCTP based TML (Transport Mapping Layer) for ForCES protocol," January 2010.) [I-D.ietf-forces-sctptml].

---

## 3. Summary

The authors attest that the ForCES Protocol, Model and SCTP-TML meet the requirements for Draft Standard.
Three independent implementations, NTT Japan, University of Patras and Zhejiang Gongshang University, were surveyed and found to already implement all the major features. All implementors mentioned they will be implementing all missing features in the future.
An interop test was conducted in July/2009 for all three implementations. Two other organizations, Mojatatu Networks and Hangzhou Baud Information and Networks Technology Corporation, which independently extended two different well known public domain protocol analyzers, Ethereal/Wireshark and tcpdump, also participated in the interop for a total of five independent organizations implementing. The two protocol analyzers were used to verify validity of ForCEs protocol messages (and in some cases semantics).
There were no notable difficulties in the interoperability test and almost all issues were code bugs that were dealt with mostly on site and tests repeated successfully.

---

## 4.  Methodology

This report has both an implementation experience survey as well as the results of the interoperability test.
The survey information was gathered after implementors answered a brief questionnaire with all ForCES Protocol, Model and SCTP-TML features.
The results can be seen in ForCES Features (Implementation Experience)
The interoperability results were part of the interoperability test.
Extended Ethereal and extended Tcpdump were used to verify the results.
The results can be seen in Interoperability Report (Interoperability Report)

## 5.  Exceptions

The core features of the ForCES Protocol, Model and SCTP-TML have been implemented and tested in an interop in July, 2009. The intention of the interop testing was to validate that all the main features of the 3 core documents were inter-operable amongst different implementations.
The tested features can be seen in Tested Features (Tested Features).
Different organizations surveyed have implemented certain features but not others. This approach is driven by presence of different LFBs the different organizations have currently implemented. All organizations surveyed have indicated intention to implement all outstanding features in due time. The implemented features can be seen in Section 6.1 (Implementation Experience).
The mandated TML security requirement, IPSec, was not validated during the interop and is not discussed in this document. Since IPSec is well known and is widely deployed not testing in presence of IPSec does not invalidate the tests done.
Although the SCTP priority ports have been changed since the interoperability test with the latest SCTP-TML draft, the change has no impact in the validity of the interoperability test.

## 6.  Detail Section

## 6.1.  Implementation Experience

Three different organizations have implemented the ForCES Protocol, Model and SCTP-TML and answered a questionnaire. These are:

*NTT Japan.

*University of Patras.

*Zhejiang Gongshang University.

Also, not actual implementations, but extensions on protocol analyzers capable of understanding ForCES protocol messages, also are considered part of an implementation as they can offer validation of exchanged protocol messages. Two such extensions have been created:

*Extension to Ethereal/Wireshark [ethereal] (, "Ethereal is a protocol analyzer. The specific ethereal that was used is an updated Ethereal, by Fenggen Jia, that can analyze and decode the ForCES protocol messages.," .).

*Extension to Tcpdump [tcpdump] (, "Tcpdump is a linux protocol analyzer. The specific tcpdump that was used is a modified tcpdump, by Jamal Hadi Salim, that can analyze and decode the ForCES protocol messages.," .).

All implementors were asked regarding the ForCES features they have implemented. For every item listed the respondents indicated whether they had implemented, will implement, or won't implement at all.

---

### 6.1.1.  ForCES Protocol Features

---

### 6.1.1.1.  Protocol Messages

| Protocol Message | NTT Japan | University of Patras | Zhejiang Gongshang University |
|---|---|---|---|
| Association Setup | Implemented | Implemented | Implemented |
| Association Setup Response | Implemented | Implemented | Implemented |
| Association TearDown | Implemented | Implemented | Implemented |
| Configuration | Implemented | Implemented | Implemented |
| Configuration Response | Implemented | Implemented | Implemented |

| | | | |
|---|---|---|---|
| Query | Implemented | Implemented | Implemented |
| Query Response | Implemented | Implemented | Implemented |
| Event Notification | Implemented | Will Implement | Implemented |
| Packet Redirect | Implemented | Will Implement | Implemented |
| HeartBeat | Implemented | Implemented | Implemented |

**ForCES Protocol Message**

### 6.1.1.2.  MainHeader Handling

| Header Field | NTT Japan | University of Patras | Zhejiang Gongshang University |
|---|---|---|---|
| Correlator | Implemented | Implemented | Implemented |
| Acknowledge Flag | Implemented | Implemented | Implemented |
| Priority Flag | Will Implement | Implemented | Implemented |
| Execution Mode Flag | Will Implement | Will Implement | Implemented |
| Atomic Flag | Will Implement | Will Implement | Implemented |
| Transaction Flag | Will Implement | Will Implement | Implemented |

**MainHeader Handling**

### 6.1.1.3.  TLV Handling

| TLV | NTT Japan | University of Patras | Zhejiang Gongshang University |
|---|---|---|---|

| | | | |
|---|---|---|---|
| Redirect TLV | Implemented | Will Implement | Implemented |
| Association Setup Result TLV | Implemented | Implemented | Implemented |
| Association TearDown Reason TLV | Implemented | Implemented | Implemented |
| LFBSelector TLV | Implemented | Implemented | Implemented |
| Operation TLV | Implemented | Implemented | Implemented |
| PathData TLV | Implemented | Implemented | Implemented |
| KeyInfo TLV | Will Implement | Will Implement | Implemented |
| FullData TLV | Implemented | Implemented | Implemented |
| SparseData TLV | Will Implement | Will Implement | Implemented |
| ILV | Will Implement | Will Implement | Implemented |
| Metadata TLV | Will Implement | Will Implement | Implemented |
| Result TLV | Implemented | Implemented | Implemented |
| Redirect Data TLV | Implemented | Will Implement | Implemented |

**TLVs Supported**

---

**6.1.1.4.  Operation Types Supported**

| Operation | NTT Japan | University of Patras | Zhejiang Gongshang University |
|---|---|---|---|
| Set | Implemented | Implemented | Implemented |
| Set Prop | Will Implement | Will Implement | Implemented |
| Set Response | Implemented | Implemented | Implemented |
| Set Prop Response | Will Implement | Will Implement | Implemented |
| Del | Implemented | Will Implement | Implemented |
| Del Response | Implemented | Will Implement | Implemented |
| Get | Implemented | Implemented | Implemented |

| | | | |
|---|---|---|---|
| Get Prop | Will Implement | Will Implement | Implemented |
| Get Response | Implemented | Implemented | Implemented |
| Get Prop Response | Will Implement | Will Implement | Implemented |
| Report | Implemented | Implemented | Implemented |
| Commit | Will Implement | Will Implement | Implemented |
| Commit Response | Will Implement | Will Implement | Implemented |
| TRComp | Will Implement | Will Implement | Implemented |

**Operation Type Supported**

6.1.1.5.  ForCES Protocol Advanced Features

| Feature | NTT Japan | University of Patras | Zhejiang Gongshang University |
|---|---|---|---|
| Execute Mode | Will Implement | Will Implement | Implemented |
| Transaction | Will Implement | Will Implement | Implemented |
| Batching | Will Implement | Implemented | Implemented |
| Command Pipelining | Will Implement | Will Implement | Will Implement |
| HeartBeats | Implemented | Implemented | Implemented |

**ForCES Protocol Advanced Features**

## 6.1.2.  ForCES Model Features

### 6.1.2.1.  Basic Atomic Types Supported

| Atomic Type | NTT Japan | University of Patras | Zhejiang Gongshang University |
|---|---|---|---|
| char | Implemented | Implemented | Will Implement |
| uchar | Implemented | Implemented | Implemented |
| int16 | Implemented | Implemented | Will Implement |
| uint16 | Implemented | Implemented | Will Implement |
| int32 | Implemented | Implemented | Implemented |
| uint32 | Implemented | Implemented | Implemented |
| int16 | Implemented | Implemented | Will Implement |
| uint64 | Implemented | Implemented | Will Implement |
| boolean | Implemented | Implemented | Implemented |
| string[N] | Implemented | Implemented | Implemented |
| string | Implemented | Implemented | Implemented |
| byte[N] | Implemented | Implemented | Implemented |
| octetstring[N] | Implemented | Implemented | Will Implement |
| float32 | Implemented | Implemented | Will Implement |
| float64 | Implemented | Implemented | Will Implement |

**Basic Atomic Types Supported**

### 6.1.2.2.  Compound Types Supported

| Compound Type | NTT Japan | University of Patras | Zhejiang Gongshang University |
|---|---|---|---|
| structs | Implemented | Implemented | Implemented |

| | | | |
|---|---|---|---|
| arrays | Implemented | Implemented | Implemented |

**Compound Types Supported**

---

### 6.1.2.3. LFBs Supported

---

### 6.1.2.3.1. FE Protocol LFB

---

| Protocol DataTypes | NTT Japan | University of Patras | Zhejiang Gongshang University |
|---|---|---|---|
| CEHBPolicy | Implemented | Implemented | Implemented |
| FEHIBPolicy | Implemented | Implemented | Implemented |
| FERestarPolicy | Implemented | Implemented | Implemented |
| CEFailoverPolicy | Implemented | Implemented | Implemented |
| FEHACapab | Implemented | Implemented | Will Implement |

**FE Protocol LFB Datatypes**

---

| Protocol Components | NTT Japan | University of Patras | Zhejiang Gongshang University |
|---|---|---|---|
| CurrentRunningVersion | Implemented | Implemented | Implemented |
| FEID | Implemented | Implemented | Implemented |
| MulticastFEIDs | Implemented | Implemented | Implemented |
| CEHBPolicy | Implemented | Implemented | Implemented |
| CEHDI | Implemented | Implemented | Implemented |
| FEHBPolicy | Implemented | Implemented | Implemented |
| FEHI | Implemented | Implemented | Implemented |
| CEID | Implemented | Implemented | Implemented |

| | | | |
|---|---|---|---|
| BackupCEs | Implemented | Will Implement | Will Implement |
| CEFailoverPolicy | Implemented | Implemented | Implemented |
| CEFTI | Implemented | Implemented | Implemented |
| FERestartPolicy | Implemented | Implemented | Will Implement |
| LastCEID | Implemented | Implemented | Will Implement |

**FE Protocol LFB Components**

| Capabilities | NTT Japan | University of Patras | Zhejiang Gongshang University |
|---|---|---|---|
| SupportableVersions | Implemented | Implemented | Implemented |
| HACapabilities | Implemented | Implemented | Will Implement |

**Capabilities Supported**

| Events | NTT Japan | University of Patras | Zhejiang Gongshang University |
|---|---|---|---|
| PrimaryCEDown | Will Implement | Will Implement | Will Implement |

**Events Supported**

**6.1.2.3.2.  FE Object LFB**

| Object DataTypes | NTT Japan | University of Patras | Zhejiang Gongshang University |
|---|---|---|---|
| LFBAdjacencyLimit | Implemented | Implemented | Implemented |
| PortGroupLimitType | Implemented | Implemented | Implemented |

| | | | |
|---|---|---|---|
| SupportedLFBType | Implemented | Implemented | Implemented |
| FEStateValues | Implemented | Implemented | Implemented |
| FEConfiguredeighborType | Implemented | Implemented | Implemented |
| FEConfiguredeighborType | Implemented | Implemented | Implemented |
| LFBSelectorType | Implemented | Implemented | Implemented |
| LFBLinkType | Implemented | Implemented | Implemented |

**FE Object LFB Datatypes**

| Object Components | NTT Japan | University of Patras | Zhejiang Gongshang University |
|---|---|---|---|
| LFBTopology | Implemented | Implemented | Implemented |
| LFBSelectors | Implemented | Implemented | Implemented |
| FEName | Implemented | Implemented | Implemented |
| FEID | Implemented | Implemented | Implemented |
| FEVendor | Implemented | Implemented | Implemented |
| FEModel | Implemented | Implemented | Implemented |
| FEState | Implemented | Implemented | Implemented |
| FENeighbors | Implemented | Implemented | Implemented |

**FE Object LFB Components**

| Capabilities | NTT Japan | University of Patras | Zhejiang Gongshang University |
|---|---|---|---|
| ModifiableLFBTopology | Implemented | Implemented | Implemented |
| SupportedLFBs | Implemented | Implemented | Implemented |

**Capabilities Supported**

### 6.1.3.  ForCES SCTP-TML Features

---

### 6.1.3.1.  TML Priority Ports

| Port | NTT Japan | University of Patras | Zhejiang Gongshang University |
|---|---|---|---|
| High priority (6700) | Implemented | Implemented | Implemented |
| Medium priority (6701) | Implemented | Implemented | Implemented |
| Low priority (6702) | Implemented | Implemented | Implemented |

**Priority Ports**

---

### 6.1.3.2.  Message Handling at specific priorities

| ForCES Message | NTT Japan | University of Patras | Zhejiang Gongshang University |
|---|---|---|---|
| Association Setup | Implemented | Implemented | Implemented |
| Association Setup Response | Implemented | Implemented | Implemented |
| Association Teardown | Implemented | Implemented | Implemented |
| Config | Implemented | Implemented | Implemented |
| Config Response | Implemented | Implemented | Implemented |
| Query | Implemented | Implemented | Implemented |
| Query Response | Implemented | Implemented | Implemented |

**Message Handling at High priority (6700) Port**

| ForCES Message | NTT Japan | University of Patras | Zhejiang Gongshang University |
|:---:|:---:|:---:|:---:|
| Event Notification | Implemented | Implemented | Implemented |

**Message Handling at Medium priority (6701) Port**

| ForCES Message | NTT Japan | University of Patras | Zhejiang Gongshang University |
|:---:|:---:|:---:|:---:|
| Packet Redirect | Implemented | Implemented | Implemented |
| Heartbeats | Implemented | Implemented | Implemented |

**Message Handling at Low priority (6702) Port**

6.1.3.3.  TML Security Feature

| Security Feature | NTT Japan | University of Patras | Zhejiang Gongshang University |
|:---:|:---:|:---:|:---:|
| IPSec | Will Implement | Will Implement | Will Implement |

**Security Feature Support**

## 6.2. Interoperability Report

The interoperability test took place at the University of Patras, in the Department of Electrical and Computer Engineering.
There were two options to participate in the interoperability test.
1. Locally at the University of Patras premises.
2. Remotely via internet.
Implementations from NTT and University of Patras, were present locally at the University of Patras premises in Greece, while the implementation from Zhejiang Gongshang University, which was behind a NAT, connected remotely from China.
The interoperability test, tested the basic functionality of the ForCES protocol, mainly message exchanging and handling.
The following scenarios were tested.

---

### 6.2.1. Scenarios

The main goal of the interoperability test was to test the basic protocol functionality, the test parameters were limited.

1. In the Association Setup Message, all report messages were ignored.

2. In the Association Setup Phase, the messages, FEO OperEnable Event (FE to CE), Config FEO Adminup (CE to FE) and FEO Config-Resp (FE to CE) were ignored. The CEs assumed that the FEs were enabled once the LFBSelectors had been queried.

3. Only FullDataTLVs were used and not SparseData TLVs.

4. There were no transaction operations.

5. Each message had only one LFBSelector TLV, one Operation TLV and one PathDataTLV per message when these were used.

---

### 6.2.1.1. Scenario 1 - Pre-association Setup

While the Pre-association setup is not in the ForCES current scope it is an essential step before CEs and FEs communicate. As the first part in a successful CE-FE connection the participating CEs and FEs had to be able to be configured.
In the Pre-association Phase the following configuration items were setup regarding the CEs:

*The CE ID.

*The FE IDs that were connected to this CE

*The IP of the FEs that connected

*The TML priority ports.

In the Pre-association Phase the following configuration items were setup regarding the FEs:

*The FE ID.

*The CE ID that this FE were connecting to.

*The IP of the CE that connected to

*The TML priority ports.

---

**6.2.1.2.  Scenario 2 - TML priority channels connection**

For the interoperability test, the SCTP was used as TML. The TML connection with the associating element was needed for the scenario 2 to be successful.
Although the latest SCTP-TML document (Salim, J. and K. Ogawa, "SCTP based TML (Transport Mapping Layer) for ForCES protocol," January 2010.) [I-D.ietf-forces-sctptml] defines 3 priority channels, with specific ports:

*High priority - Port number: 6704

*Medium priority - Port number: 6705

*Lower priority - Port number: 6706

At the time of the interoperability test, the sctp ports of the three priority channels were the following:

*High priority - Port number: 6700

*Medium priority - Port number: 6701

*Lower priority - Port number: 6702

As specified in the exceptions section, this does not invalidate the results of the interoperability test.

### 6.2.1.3.  Scenario 3 - Association Setup - Association Complete

Once the Pre-association phase had been complete in the previous 2
scenarios, CEs and FEs would be ready to communicate using the ForCES
protocol, and enter the Association Setup stage. In this stage the FEs
would attempt to join the NE. The following ForCES protocol messages
would be exchanged for each CE-FE pair in the specified order:

  *Association Setup Message (from FE to CE)

  *Association Setup Response Message (from CE to FE)

  *Query Message: FEO LFBSelectors(from CE to FE)

  *Query Response: FEO LFBSelectors response (from FE to CE)

### 6.2.1.4.  Scenario 4 - CE query

Once the Association Phase stage has been complete, the FEs and CEs
would enter the Established stage. In this stage the FE will be
continuously updated or queried. The CE should query the FE a specific
value from the FE Object LFB and from the FE Protocol LFB. An example
from the FE Protocol LFB is the HeartBeat Timer (FEHI) and from the FE
Object LFB is the State of the LFB (FEState)
The following ForCES protocol messages were exchanged:

  *Query Message

  *Query Response Message

### 6.2.1.5.  Scenario 5 - Heartbeat monitoring

The Heartbeat (HB) Message is used for one ForCES element (FE or CE) to
asynchronously notify one or more other ForCES elements in the same
ForCES NE on its liveness. The default configuration of the Heartbeat
Policy of the FE is set to 0 which means, that the FE should not
generate any Heartbeat messages. the CE is responsible for checking FE
liveness by setting the PL header ACK flag of the message it sends to

AlwaysACK. In this Scenario the CE will send a Heartbeat message with
the ACK flag set to AlwaysACK and the FE should respond.
The following ForCES protocol messages were exchanged:

> *Heartbeat Message

---

### 6.2.1.6.  Scenario 6 - Simple Config Command

A config message is sent by the CE to the FE to configure LFB
components in the FE. A simple config command easily visible and
metered would be to change the Heartbeat configuration. This was done
in two steps:

> 1. Change the FE Heartbeat Policy (FEHBPolicy) to value 1, to
>    force the FE to send heartbeats.
>
> 2. After some heartbeats from the FE, the FE Heartbeat Interval
>    (FEHI) was changed.

The following ForCES protocol messages were exchanged:

> *Config Message
>
> *Config Response Message

---

### 6.2.1.7.  Scenario 7 - Association Teardown

In the end, the association must be terminated. There were three
scenarios by which the association was terminated:

> 1. Normal tear down by exchanging Association Teardown Message
>
> 2. Irregular tear down by stopping heartbeats from a FE or a CE.
>
> 3. Irregular tear down by externally shutting down/rebooting a FE
>    or a CE.

All scenarios were tested in the interoperability test.
The following ForCES protocol messages were exchanged:

> *Association Teardown Message

### 6.2.2. Tested Features

The features that were tested are:

### 6.2.2.1. ForCES Protocol Features

### 6.2.2.1.1. Protocol Messages

| Protocol Message |
| :---: |
| Association Setup |
| Association Setup Response |
| Association TearDown |
| Configuration |
| Configuration Response |
| Query |
| Query Response |
| HeartBeat |

**ForCES Protocol Message**

*PASS: All implementations handled the protocol messages and all
 protocol analyzers captured them.

### 6.2.2.1.2. MainHeader Handling

| Header Field |
| :---: |
| Correlator |

| Acknowledge Flag |
| Priority Flag |

**MainHeader Handling**

---

*PASS: All implementations handled these main header flags and all protocol analyzers captured them.

---

### 6.2.2.1.3. TLV Handling

---

| TLV |
| :---: |
| Association Setup Result TLV |
| Association TearDown Reason TLV |
| LFBSelector TLV |
| Operation TLV |
| PathData TLV |
| FullData TLV |
| Result TLV |

**TLVs Supported**

---

*PASS: All implementations handled these TLVs and all protocol analyzers captured them.

---

### 6.2.2.1.4. Operation Types Supported

---

| Operation |
| :---: |
| Set |
| Set Response |

| Get |
|-----|
| Get Response |
| Report |

**Operation Type Supported**

---

*PASS: All implementations handled these Operations and all
 protocol analyzers captured them.

---

**6.2.2.1.5.  ForCES Protocol Advanced Features**

---

| Feature |
|---------|
| Batching |
| HeartBeats |

**ForCES Protocol Advanced Features**

---

Although Batching was not initially designed to be tested, it was
tested during the interoperability test.

*PASS: Two implementations handled batching and all handled
 Heartbeats. The protocol analyzers captured both.

---

**6.2.2.2.  ForCES Model Features**

---

**6.2.2.2.1.  Basic Atomic Types Supported**

---

| Atomic Type |
|-------------|

| |
|---|
| uchar |
| uint32 |

**Basic Atomic Types Supported**

---

   *PASS: All implementations handled these basic atomic types.

---

### 6.2.2.2.2.  Compound Types Supported

---

| Compound Type |
|---|
| structs |
| arrays |

**Compound Types Supported**

---

   *PASS: All implementations handled these compound types.

---

### 6.2.2.2.3.  LFBs Supported

---

### 6.2.2.2.3.1.  FE Protocol LFB

---

| Protocol DataTypes |
|---|
| CEHBPolicy |
| FEHIBPolicy |

**FE Protocol LFB Datatypes**

*PASS: All implementations handled these FE Protocol LFB
 Datatypes.

| Protocol Components |
|:---:|
| FEID |
| CEHBPolicy |
| CEHDI |
| FEHBPolicy |
| FEHI |
| CEID |

**FE Protocol LFB Components**

*PASS: All implementations handled these FE Protocol LFB
 Components.

**6.2.2.2.3.2.  FE Object LFB**

| Object DataTypes |
|:---:|
| FEStateValues |
| LFBSelectorType |

**FE Object LFB Datatypes**

*PASS: All implementations handled these FE Object LFB Datatypes.

| Object Components |
|:---:|
| LFBSelectors |
| FEState |

**FE Object LFB Components**

---

  *PASS: All implementations handled these FE Object LFB Components.

---

**6.2.2.3.  ForCES SCTP-TML Features**

---

**6.2.2.3.1.  TML Priority Ports**

---

| Port |
|------|
| High priority (6700) |
| Medium priority (6701) |
| Low priority (6702) |

**Priority Ports**

---

  *PASS: All implementations opened and connected to all the SCTP
   priority ports. The protocol analyzers captured all ports and
   corresponding priority.

---

**6.2.2.3.2.  Message Handling at specific priorities**

---

| ForCES Message |
|----------------|
| Association Setup |
| Association Setup Response |
| Association Teardown |
| Config |

| Config Response |
| Query |
| Query Response |

**Message Handling at High priority (6700) Port**

---

*PASS: All implementations handled these messages at this SCTP
priority port. The protocol analyzers captured these messages at
these priority ports.

---

| **ForCES Message** |
| Heartbeats |

**Message Handling at Low priority (6702) Port**

---

*PASS: All implementations handled these messages at this SCTP
priority port. The protocol analyzers captured these messages at
these priority ports.

---

6.2.3.  Interoperability Results

All implementations were found to be interoperable with each other.
All scenarios were tested successfully.
The following issues were found and dealt with.

1. Some messages were sent to the wrong priority channels. There
   was some ambiguities on the SCTP-TML document on what must be
   the correct response from the CE and FE. Should it respond in
   the same channel, should it respond in the correct channel, or
   should it drop the packet? This has been corrected by changing
   the word recommended to MUST in the SCTP-TML document regarding
   priority channel messaging .

2. At some point, a CE sent a TearDown message to the FE. The CE
   expected the FE to shut down the connection, and the FE waited
   the CE to shut down the connection and were caught in a
   deadlock. This was a code bug and was fixed.

3. Sometimes the association setup message, only on the remote
   connection test, although sent, was not received by the other
   end and made impossible the association. This was caused by
   network problems.

4. An implementation did not take into account that the padding in
   TLVs MUST NOT be included in the length of the TLV. This was a
   code bug and was fixed.

5. EM Flag was set to reserved by a CE and was not ignored by the
   FE. This was a code bug and was fixed.

6. After the FEHBPolicy was set to 1 the FE didn't send any
   HeartBeats. This was a code bug and was fixed.

7. Some FEs sent HeartBeats with the ACK flag with a value other
   than NoACK. The CE responded. This was a code bug and was
   fixed.

8. When a cable was disconnected, all TML implementation didn't
   detect it. The association was eventually dropped due to
   heartbeats, this was a success, but this is an implementation
   issue implementers should keep in mind. This is a SCTP options
   issue. Nothing was needed to be done.

9. A CE crashed due to unknown LFBSelector values. This was a code
   bug and was fixed.

10. With the remote connection from China, which was behind a NAT,
    to Greece there were a lot of ForCES packet retransmission. The
    problem is that packets like Heartbeats were retransmitted.
    This was a SCTP issue. SCTP-PR was needed to be used.

The interoperability test went so well that an additional extended test
was added to test for batching messages. This test was also done
successfully.

---

## 7. Acknowledgements [TOC]

---

## 8.  IANA Considerations

This memo includes no request to IANA.

---

## 9.  Security Considerations

For Security considerations please see [I-D.ietf-forces-protocol] (Dong, L., Doria, A., Gopal, R., HAAS, R., Salim, J., Khosravi, H., and W. Wang, "ForCES Protocol Specification," March 2009.) and [I-D.ietf-forces-sctptml] (Salim, J. and K. Ogawa, "SCTP based TML (Transport Mapping Layer) for ForCES protocol," January 2010.)

---

## 10.  References

---

## 10.1. Normative References

| | |
|---|---|
| [I-D.ietf-forces-model] | Halpern, J. and J. Salim, "ForCES Forwarding Element Model," draft-ietf-forces-model-16 (work in progress), October 2008 (TXT). |
| [I-D.ietf-forces-protocol] | Dong, L., Doria, A., Gopal, R., HAAS, R., Salim, J., Khosravi, H., and W. Wang, "ForCES Protocol Specification," draft-ietf-forces-protocol-22 (work in progress), March 2009 (TXT). |
| [I-D.ietf-forces-sctptml] | Salim, J. and K. Ogawa, "SCTP based TML (Transport Mapping Layer) for ForCES protocol," draft-ietf-forces-sctptml-08 (work in progress), January 2010 (TXT). |

## 10.2. Informative References

| [RFC2119] | Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels," BCP 14, RFC 2119, March 1997 (TXT, HTML, XML). |
|---|---|
| [RFC2629] | Rose, M., "Writing I-Ds and RFCs using XML," RFC 2629, June 1999 (TXT, HTML, XML). |
| [RFC3552] | Rescorla, E. and B. Korver, "Guidelines for Writing RFC Text on Security Considerations," BCP 72, RFC 3552, July 2003 (TXT). |
| [RFC3654] | Khosravi, H. and T. Anderson, "Requirements for Separation of IP Control and Forwarding," RFC 3654, November 2003 (TXT). |
| [RFC3746] | Yang, L., Dantu, R., Anderson, T., and R. Gopal, "Forwarding and Control Element Separation (ForCES) Framework," RFC 3746, April 2004 (TXT). |
| [RFC5226] | Narten, T. and H. Alvestrand, "Guidelines for Writing an IANA Considerations Section in RFCs," BCP 26, RFC 5226, May 2008 (TXT). |
| [RFC5657] | Dusseault, L. and R. Sparks, "Guidance on Interoperation and Implementation Reports for Advancement to Draft Standard," BCP 9, RFC 5657, September 2009 (TXT). |
| [ethereal] | "Ethereal is a protocol analyzer. The specific ethereal that was used is an updated Ethereal, by Fenggen Jia, that can analyze and decode the ForCES protocol messages.." |
| [tcpdump] | "Tcpdump is a linux protocol analyzer. The specific tcpdump that was used is a modified tcpdump, by Jamal Hadi Salim, that can analyze and decode the ForCES protocol messages.." |

## Authors' Addresses

| | |
|---|---|
| | Evangelos Haleplidis |
| | University of Patras |
| | Patras, |
| | Greece |
| Email: | ehalep@ece.upatras.gr |
| | |
| | Kentaro Ogawa |
| | NTT Corporation |
| | Tokyo, |
| | Japan |
| Email: | ogawa.kentaro@lab.ntt.co.jp |
| | |

| | |
|---|---|
| | Weiming Wang |
| | Zhejiang Gongshang University |
| | 18, Xuezheng Str., Xiasha University Town |
| | Hangzhou, 310018 |
| | P.R.China |
| Phone: | +86-571-28877721 |
| Email: | wmwang@mail.zjgsu.edu.cn |
| | |
| | Jamal Hadi Salim |
| | Mojatatu Networks |
| | Ottawa, Ontario, |
| | Canada |
| Phone: | |
| Email: | hadi@mojatatu.com |