Internet Engineering Task Force (IETF)
Request for Comments: 6984
Updates: 6053
Category: Informational
ISSN: 2070-1721

W. Wang
Zhejiang Gongshang University
K. Ogawa
NTT Corporation
E. Haleplidis
University of Patras
M. Gao
Hangzhou BAUD Networks
J. Hadi Salim
Mojatatu Networks

August 2013

Interoperability Report for Forwarding and Control Element Separation (ForCES)

Abstract

This document captures the results of the second Forwarding and Control Element Separation (ForCES) interoperability test that took place on February 24-25, 2011, in the Internet Technology Lab (ITL) at Zhejiang Gongshang University, China. The results of the first ForCES interoperability test were reported in RFC 6053, and this document updates RFC 6053 by providing further interoperability results.

Status of This Memo

This document is not an Internet Standards Track specification; it is published for informational purposes.

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). Not all documents approved by the IESG are a candidate for any level of Internet Standard; see 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/rfc6984.

Table of Contents

1. Introduction

This document captures the results of the second interoperability test of the Forwarding and Control Element Separation (ForCES) that took place on February 24-25, 2011, in the Internet Technology Lab (ITL) at Zhejiang Gongshang University, China. The test involved protocol elements described in several documents, namely:

- ForCES Protocol [RFC5810]

- ForCES Forwarding Element (FE) Model [RFC5812]

- ForCES Transport Mapping Layer (TML) [RFC5811]

The test also involved protocol elements described in the then-current versions of two Internet-Drafts. Although these documents have subsequently been revised and advanced, it is important to understand which versions of the work were used during this test. The then-current Internet-Drafts are:

- "ForCES Logical Function Block (LFB) Library" (December 2010) [LFB-LIB]

- "ForCES Intra-NE High Availability" (October 2010) [CEHA]

Note: The ForCES Logical Function Block (LFB) Library document was published as [RFC6956].

Three independent ForCES implementations participated in the test.

Scenarios of ForCES LFB Operation, TML with IPsec, Control Element High Availability (CEHA), and Packet Forwarding were constructed. Series of testing items for every scenario were carried out and interoperability results were achieved. The popular packet analyzers Ethereal/Wireshark [Ethereal] and Tcpdump [Tcpdump] were used to verify the wire results.

This document is an update to [RFC6053], which captured the results of the first ForCES interoperability test. The first test on ForCES was held in July 2008 at the University of Patras, Greece. That test focused on validating the basic semantics of the ForCES protocol and ForCES Forwarding Element (FE) model.

1.1. ForCES Protocol

The ForCES protocol works in a master-slave mode in which FEs are slaves and Control Elements (CEs) are masters. The protocol includes commands for transport of Logical Function Block (LFB) configuration

information, association setup, status, event notifications, etc. The reader is encouraged to read the ForCES protocol specification [RFC5810] for further information.

1.2. ForCES FE Model

The ForCES FE model [RFC5812] presents a formal way to define FE 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.

1.3. Transport Mapping Layer

The ForCES Transport Mapping Layer (TML) transports the ForCES protocol layer messages. The TML is where the issues of how to achieve transport-level reliability, congestion control, multicast, ordering, etc., are handled. It is expected that more than one TML will be standardized. RFC 5811 specifies a TML that is based on the Stream Control Transmission Protocol (SCTP) and is a mandated TML for ForCES. See RFC 5811 for more details.

1.4. Definitions

This document follows the terminology defined by ForCES-related documents, including [RFC3654], [RFC3746], [RFC5810], [RFC5811], [RFC5812], [RFC5813], etc.

2. Overview

2.1. Date, Location, and Participants

The second ForCES interoperability test meeting was held by the IETF ForCES Working Group on February 24-25, 2011, and was chaired by Jamal Hadi Salim. Three independent ForCES implementations participated in the test:

- Zhejiang Gongshang University/Hangzhou BAUD Corporation of Information and Networks Technology (Hangzhou BAUD Networks), China. This implementation is referred to as "ZJSU" or "Z" in this document for the sake of brevity.

- NTT Corporation, Japan. This implementation is referred to as "NTT" or "N" in this document for the sake of brevity.

- The University of Patras, Greece. This implementation is referred to as "UoP" or "P" in this document for the sake of brevity.

Two other organizations, Mojatatu Networks and Hangzhou BAUD Networks Corporation, which independently extended two different well-known public domain protocol analyzers, Ethereal/Wireshark [Ethereal] and Tcpdump [Tcpdump], also participated in the interoperability test. During the test, the two protocol analyzers were used to verify the validity (and in some cases, the semantics) of ForCES protocol messages.

Some issues related to interoperability among implementations were discovered. Most of the issues were solved on site during the test. The most contentious issue found was on the format of encapsulation for the protocol TLVs (refer to Section 5.1).

Some errata related to the ForCES document were found by the interoperability test. The errata found in related RFCs have also been reported.

At times, interoperability testing was exercised between two instead of all three representative implementations because the third one lacked a specific feature; however, in ensuing discussions, all implementers mentioned they would be implementing any missing features in the future.

2.2. Testbed Configuration

2.2.1. Participants' Access

NTT and ZJSU were physically present for the testing at the Internet Technology Lab (ITL) at Zhejiang Gongshang University in China. The implementation team from the University of Patras joined remotely from Greece. The chair, Jamal Hadi Salim, joined remotely from Canada by using TeamViewer as the monitoring tool [TeamViewer]. The approach was as shown in Figure 1. In the figure, FE/CE refers to the FE or CE that the implementer may act as alternatively.

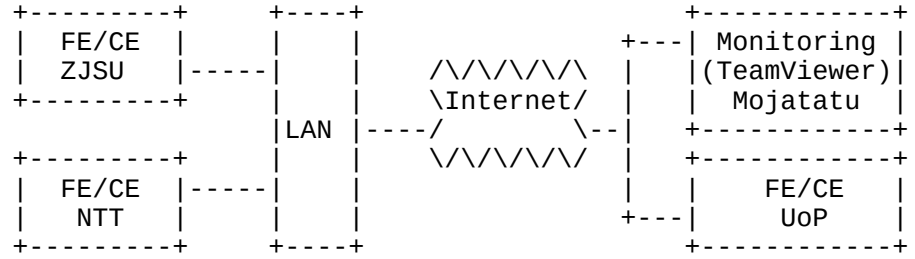

Figure 1: Access for Participants

As specified in [RFC5811], all CEs and FEs implemented IPsec in the TML.

On the Internet boundary, gateways used must allow for IPsec, the SCTP protocol, and SCTP ports as defined in the ForCES SCTP-based TML document [RFC5811].

2.2.2. Testbed Configuration

The CEs and FEs from ZJSU's and NTT's implementations were physically located within the ITL Lab at Zhejiang Gongshang University and connected together using Ethernet switches. The configuration can be seen in Figure 2. In the figure, SmartBits [SmartBits] is a third-party routing protocol testing machine that acts as a router running Open Shortest Path First (OSPF) and RIP, and exchanges routing protocol messages with ForCES routers in the network. Connection to the Internet was via an Asymmetric Digital Subscriber Line (ADSL) channel.

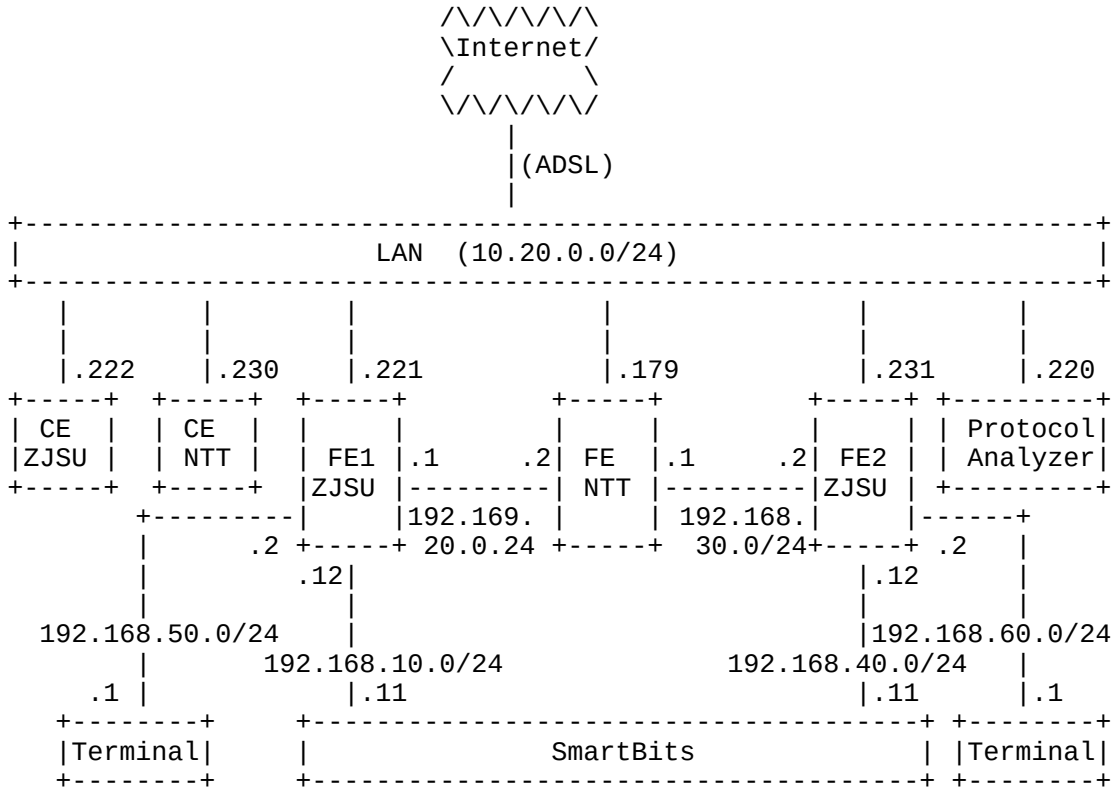

Figure 2: Testbed Configuration Located in the ITL Lab, China

The CE and FE from the UoP implementation were located within the University of Patras, Greece, and were connected together using LAN, as shown in Figure 3. Connection to the Internet was via a VPN channel.

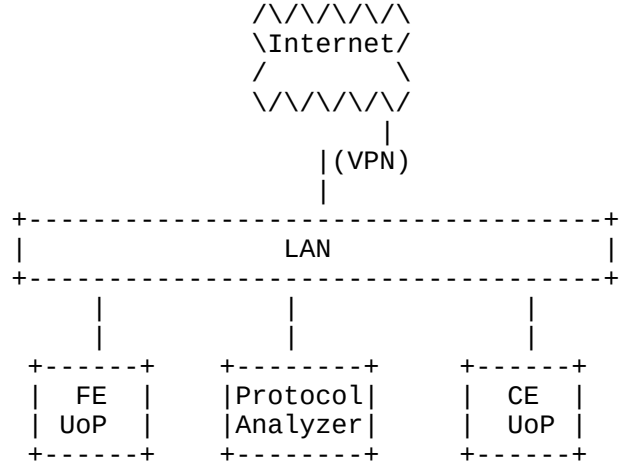

Figure 3: Testbed Configuration
Located in the University of Patras, Greece

The testbeds above were then able to satisfy the requirements of all interoperability test scenarios in this document.

3. Scenarios

3.1. Scenario 1 - LFB Operation

This scenario was designed to test the interoperability of LFB operations among the participants. The connection diagram for the participants is shown in Figure 4.

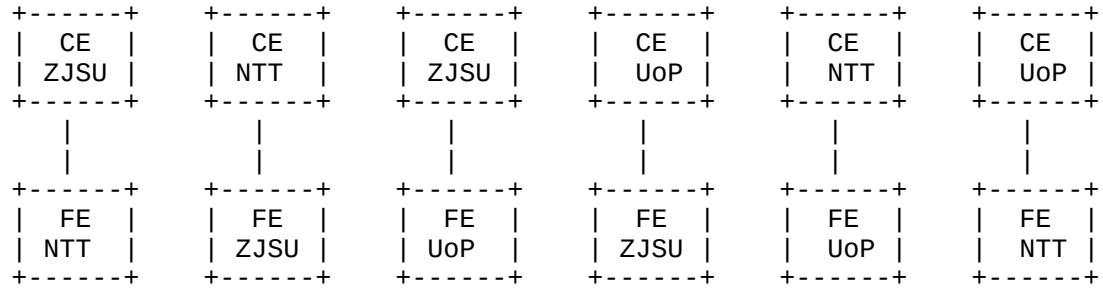

Figure 4: Scenario for LFB Operation

In order to make interoperability more credible, the three implementers were required to carry out the test acting as a CE or FE alternatively. As a result, every LFB operation was combined with six scenarios, as shown by Figure 4.

The test scenario was designed with the following purposes.

Firstly, the scenario was designed to verify all kinds of protocol messages with their complex data formats, which were defined in [RFC5810]. Specifically, we tried to verify the data format of a PATH-DATA-TLV with nested PATH-DATA-TLVs, and the operation (SET, GET, and DEL) of an array or an array with a nested array.

Secondly, the scenario was designed to verify the definition of ForCES LFB Library [LFB-LIB], which defined a base set of ForCES LFB classes for typical router functions. Successful tests under this scenario would help the validity of the LFB definitions.

3.2. Scenario 2 - TML with IPsec

This scenario was designed to implement a TML with IPsec, which was the requirement defined by RFC 5811. TML with IPsec was not implemented and tested in the first ForCES interoperability test, as reported in RFC 6053. For this reason, in this interoperability test, we specifically designed the test scenario to verify the TML over IPsec channel.

In this scenario, tests on LFB operations for Scenario 1 were repeated with the difference that TML was secured via IPsec. This setup scenario allowed us to verify whether all interactions between the CE and FE could be made correctly under an IPsec TML environment.

The connection diagram for this scenario is shown in Figure 5. Because an unfortunate problem with the test system in the UoP prevented the deployment of IPsec over TML, this test only took place between the test systems in ZJSU and NTT.

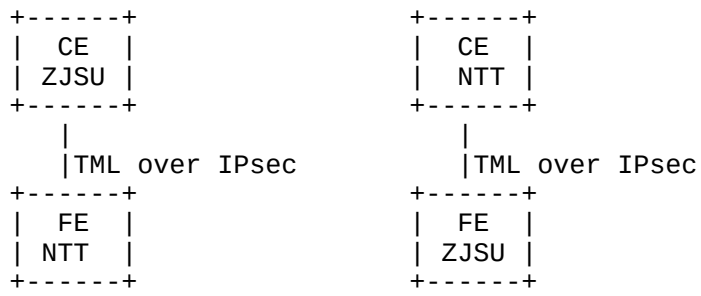

Figure 5: Scenario for LFB Operation with TML over IPsec

In this scenario, ForCES TML was run over the IPsec channel. Implementers joined in this interoperability test using the same third-party software 'Racoon' [Racoon] to establish the IPsec channel.

The Racoon in NetBSD is an Internet Key Exchange (IKE) daemon that performs the key exchange with the peers. Both IKEv1 and IKEv2 are supported by Racoon in Linux 2.6, and IKEv2 was adopted in the interop test. The Security Association Database (SAD) and Security Policy Database (SPD) were necessary for the test, setups of which were in the Racoon configuration file. The Encapsulating Security Payload (ESP) was specified in the SAD and SPD in the Racoon configuration file.

ZJSU and NTT conducted a successful test with the scenario, and the IPsec requirement items in [RFC5812] were realized.

3.3. Scenario 3 - CE High Availability

CE High Availability (CEHA) was tested based on the ForCES CEHA document [CEHA].

The design of the setup and the scenario for the CEHA were simplified so as to focus mostly on the mechanics of the CEHA, which were:

- Associating with more than one CE.
- Switching to a backup CE on a master CE failure.

The connection diagram for the scenario is shown in Figure 6.

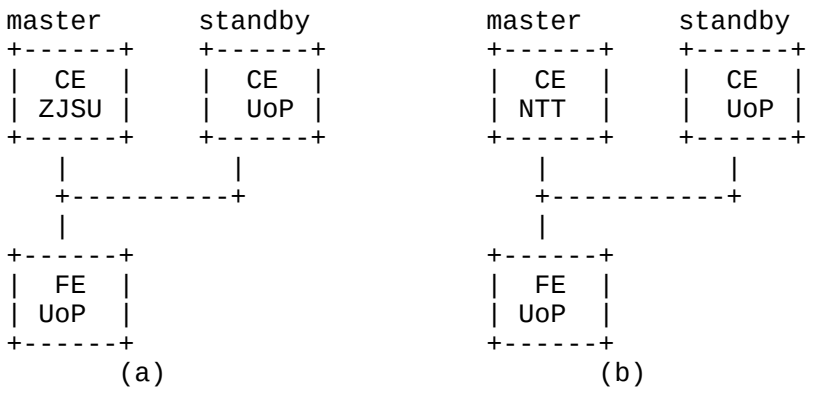

Figure 6: Scenario for CE High Availability

In this scenario, one FE was connected and associated to a master CE and a backup CE. In the pre-association phase, the FE would be configured to have ZJSU's or NTT's CE as the master CE and the UoP's CE as the standby CE. The CEFailoverPolicy component of the FE Protocol Object LFB that specified whether the FE was in High Availability mode (value 2 or 3) would be set either in the pre-association phase by the FE interface or in the post-association phase by the master CE.

If the CEFailoverPolicy value was set to 2 or 3, the FE (in the post-association phase) would attempt to connect and associate with the standby CE.

When the master CE was deemed disconnected, either by a TearDown, Loss of Heartbeats, or physically disconnected, the FE would assume that the standby CE was now the master CE. The FE would then send an Event Notification, Primary CE Down, to all associated CEs (only the standby CE in this case) with the value of the new master Control Element ID (CEID). The standby CE would then respond by sending a configuration message to the CEID component of the FE Protocol Object with its own ID to confirm that the CE considered itself the master as well.

The steps of the CEHA test scenario were as follows:

1. In the pre-association phase, the FE is set up with the master CE and the backup CE.

2. The FE connects and associates with the master CE.

3. When CEFailoverPolicy is set to 2 or 3, the FE connects and associates with the backup CE.

4. Once the master CE is considered disconnected, then the FE chooses the first associated backup CE.

5. It sends an Event Notification that specifies the master CE is down and identifies the new master CE.

6. The new master CE sends a SET Configuration message to the FE; the FE then sets the CEID value to the new master CE completing the switch.

3.4. Scenario 4 - Packet Forwarding

This test scenario was conducted to verify LFBs like RedirectIn, RedirectOut, IPv4NextHop, and IPv4UcastLPM, which were defined by the ForCES LFB library document [LFB-LIB], and more importantly, to verify the combination of the LFBs to implement IP packet forwarding.

The connection diagram for this scenario is shown in Figure 7.

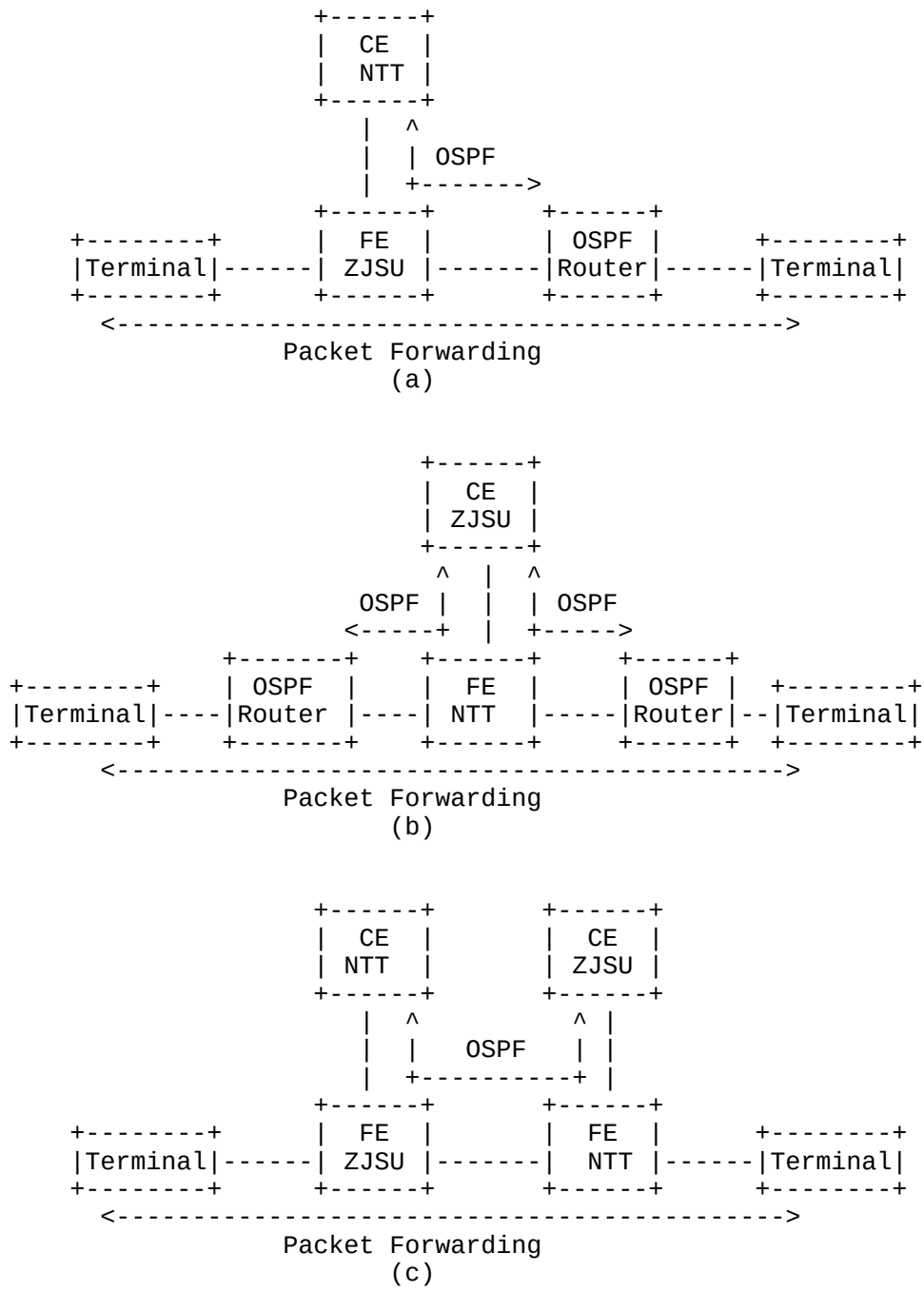

Figure 7: Scenario for IP Packet Forwarding

In case (a), NTT's CE was connected to ZJSU's FE to form a ForCES router. A SmartBits [SmartBits] test machine equipped with routing protocol software was used to simulate an OSPF router and was connected with the ForCES router to try to exchange OSPF Hello packets and Link State Advertisement (LSA) packets among them. Terminals were simulated by SmartBits to send and receive packets. As a result, the CE in the ForCES router needed to be configured to run and support the OSPF routing protocol.

In case (b), ZJSU'S CE was connected to NTT'S FE to form a ForCES router. Two routers running OSPF were simulated and connected to the ForCES router to test if the ForCES router could support the OSPF protocol and support packet forwarding.

In case (c), two ForCES routers were constructed; one was with NTT's CE and ZJSU's FE, and the other was with NTT's FE and ZJSU's CE. OSPF and packet forwarding were tested in the environment.

The testing process for this scenario is shown below:

1. Boot terminals and routers, and set the IP addresses of their interfaces.

2. Boot the CE and FE.

3. Establish an association between the CE and FE, and set the IP addresses of the FE interfaces.

4. Start OSPF among the CE and routers, and set the Forwarding Information Base (FIB) on the FE.

5. Send packets between terminals.

4. Test Results

4.1. Test of LFB Operation

The test results are reported in Figure 8. As mentioned earlier, for convenience, the following abbreviations are used in the table: "Z" for the implementation from ZJSU, "N" for the implementation from NTT, and "P" for the implementation from the UoP.

Test#	CE	FE(s)	Oper	LFB	Component /Capability	Result
1	Z	N	GET	FEObject	LFBTopology	Success
	N	Z				Success
	P	Z				Success
	N	P				Success
	P	N				Success
2	Z	N	GET	FEObject	LFBSelector	Success
	N	Z				Success
	Z	P				Success
	P	Z				Success
	N	P				Success
	P	N				Success
3	Z	N	GET	EtherPHYCop	PHYPortID	Success
	N	Z				Success
	Z	P				Success
	P	Z				Success
	N	P				Success
	P	N				Success
4	Z	N	GET	EtherPHYCop	AdminStatus	Success
	N	Z				Success
	Z	P				Success
	P	Z				Success
	N	P				Success
	P	N				Success
5	Z	N	GET	EtherPHYCop	OperStatus	Success
	N	Z				Success
	Z	P				Success
	P	Z				Success
	N	P				Success
	P	N				Success

6	Z	N	GET	EtherPHYCop	AdminLinkSpeed	Success
	N	Z				Success
	Z	P				Success
	P	Z				Success
	N	P				Success
	P	N				Success
7	Z	N	GET	EtherPHYCop	OperLinkSpeed	Success
	N	Z				Success
	Z	P				Success
	P	Z				Success
	N	P				Success
	P	N				Success
8	Z	N	GET	EtherPHYCop	AdminDuplexSpeed	Success
	N	Z				Success
	Z	P				Success
	P	Z				Success
	N	P				Success
	P	N				Success
9	Z	N	GET	EtherPHYCop	OperDuplexSpeed	Success
	N	Z				Success
	Z	P				Success
	P	Z				Success
	N	P				Success
	P	N				Success
10	Z	N	GET	EtherPHYCop	CarrierStatus	Success
	N	Z				Success
	Z	P				Success
	P	Z				Success
	N	P				Success
	P	N				Success
11	Z	N	GET	EtherMACIn	AdminStatus	Success
	N	Z				Success
	Z	P				Success
	P	Z				Success
	N	P				Success
	P	N				Success
12	Z	N	GET	EtherMACIn	LocalMacAddresses	Success
	N	Z				Success
	Z	P				Success
	P	Z				Success
	N	P				Success
	P	N				Success

13	Z	N	GET	EtherMACIn	L2Bridging PathEnable	Success
	N	Z				Success
	Z	P				Success
	P	Z				Success
	N	P				Success
	P	N				Success
14	Z	N	GET	EtherMACIn	PromiscuousMode	Success
	N	Z				Success
	Z	P				Success
	P	Z				Success
	N	P				Success
	P	N				Success
15	Z	N	GET	EtherMACIn	TxFlowControl	Success
	N	Z				Success
	Z	P				Success
	P	Z				Success
	N	P				Success
	P	N				Success
16	Z	N	GET	EtherMACIn	RxFlowControl	Success
	N	Z				Success
	Z	P				Success
	P	Z				Success
	N	P				Success
	P	N				Success
17	Z	N	GET	EtherMACIn	MACInStats	Success
	N	Z				Success
	Z	P				Success
	P	Z				Success
	N	P				Success
	P	N				Success
18	Z	N	GET	EtherMACOut	AdminStatus	Success
	N	Z				Success
	Z	P				Success
	P	Z				Success
	N	P				Success
	P	N				Success

19	Z	N	GET	EtherMACOut	MTU	Success
	N	Z				Success
	Z	P				Success
	P	Z				Success
	N	P				Success
	P	N				Success
20	Z	N	GET	EtherMACOut	TxFlowControl	Success
	N	Z				Success
	Z	P				Success
	P	Z				Success
	N	P				Success
	P	N				Success
21	Z	N	GET	EtherMACOut	TxFlowControl	Success
	N	Z				Success
	Z	P				Success
	P	Z				Success
	N	P				Success
	P	N				Success
22	Z	N	GET	EtherMACOut	MACOutStats	Success
	N	Z				Success
	Z	P				Success
	P	Z				Success
	N	P				Success
	P	N				Success
23	Z	N	GET	ARP	PortV4AddrInfoTable	Success
	N	Z				Success
	Z	P				Success
	P	Z				Success
	N	P				Success
	P	N				Success
24	Z	N	SET	ARP	PortV4AddrInfoTable	Success
	N	Z				Success
	Z	P				Success
	P	Z				Success
	N	P				Success
	P	N				Success
25	Z	N	DEL	ARP	PortV4AddrInfoTable	Success
	N	Z				Success
	Z	P				Success
	P	Z				Success
	N	P				Success
	P	N				Success

26	Z	N	SET	EtherMACIn	LocalMACAddresses	Success
	N	Z				Success
	Z	P				Success
	P	Z				Success
	N	P				Success
	P	N				Success
27	Z	N	SET	EtherMACIn	MTU	Success
	N	Z				Success
	Z	P				Success
	P	Z				Success
	N	P				Success
	P	N				Success
28	Z	N	SET	IPv4NextHop	IPv4NextHopTable	Success
	N	Z				Success
	Z	P				Success
	P	Z				Success
	N	P				Success
	P	N				Success
29	Z	N	SET	IPv4UcastLPM	IPv4PrefixTable	Success
	N	Z				Success
	Z	P				Success
	P	Z				Success
	N	P				Success
	P	N				Success
30	Z	N	DEL	IPv4NextHop	IPv4NextHopTable	Success
	N	Z				Success
	Z	P				Success
	P	Z				Success
	N	P				Success
	P	N				Success
31	Z	N	DEL	IPv4UcastLPM	IPv4PrefixTable	Success
	N	Z				Success
	Z	P				Success
	P	Z				Success
	N	P				Success
	P	N				Success

32	Z	N	SET	EtherPHYCop	AdminStatus	Success
	N	Z				Success
	Z	P				Success
	P	Z				Success
	N	P				Success
	P	N				Success
33	Z	N	SET	Ether Classifier	VlanInputTable	Success
	N	Z				Success
	Z	P				Success
	P	Z				Success
	N	P				Success
	P	N				Success
34	Z	N	DEL	Ether Classifier	VlanInputTable	Success
	N	Z				Success
	Z	P				Success
	P	Z				Success
	N	P				Success
	P	N				Success
35	Z	N	SET	Ether Encapsulator	VlanOutputTable	Success
	N	Z				Success
	Z	P				Success
	P	Z				Success
	N	P				Success
	P	N				Success
36	Z	N	DEL	Ether Encapsulator	VlanOutputTable	Success
	N	Z				Success
	Z	P				Success
	P	Z				Success
	N	P				Success
	P	N				Success

Figure 8: Test Results for LFB Operation

Note on tests #1 and #2:

On the wire format of encapsulation on array, only the case of FULLDATA-TLV vs. SPARSEDATA-TLV was tested.

When we use the ForCES protocol, it is very common for the CE to use the FEobject LFB to get information on LFBs and their topology in the FE. Hence, the two tests were specifically made.

4.2. Test of TML with IPsec

In this scenario, the ForCES TML was run over IPsec. Implementers joined this interoperability test and used the same third-party tool software 'Racoon' [Racoon] to establish the IPsec channel. Typical LFB operation tests as in Scenario 1 were repeated with the IPsec-enabled TML.

As mentioned, this scenario only took place between implementers from ZJSU and NTT.

The TML with IPsec test results are reported in Figure 9.

Test#	CE	FE(s)	Oper	LFB	Component/ Capability	Result
1	Z	N	GET	FEObject	LFBTopology	Success
	N	Z				Success
2	Z	N	GET	FEObject	LFBSelectors	Success
	N	Z				Success
3	Z	N	SET	Ether Classifier	VlanInputTable	Success
	N	Z				Success
4	Z	N	DEL	Ether Classifier	VlanInputTable	Success
	N	Z				Success

Figure 9: Test Results for TML with IPsec

4.3. Test of CE High Availability

In this scenario, one FE connected and associated with a master CE and a backup CE. When the master CE was deemed disconnected, the FE attempted to find another associated CE to become the master CE.

The CEHA scenario, as described in Scenario 3, was completed successfully for both setups.

Due to a bug in one of the FEs, an interesting issue was caught: it was observed that the buggy FE took up to a second to failover. It was eventually found that the issue was due to the FE's prioritization of the different CEs. All messages from the backup CE were being ignored unless the master CE was disconnected.

While the bug was fixed and the CEHA scenario was completed successfully, the authors felt it was important to capture the implementation issue in this document. The recommended approach is the following:

- The FE should receive and handle messages first from the master CE on all priority channels to maintain proper functionality and then receive and handle messages from the backup CEs.

- Only when the FE is attempting to associate with the backup CEs should the FE receive and handle messages per priority channel from all CEs. When all backup CEs are associated with or deemed unreachable, then the FE should return to receiving and handling messages first from the master CE.

4.4. Test of Packet Forwarding

As described in the ForCES LFB library [LFB-LIB], packet forwarding is implemented by a set of LFB classes that compose a processing path for packets. In this test scenario, as shown in Figure 7, a ForCES router running the OSPF protocol was constructed. In addition, a set of LFBs including RedirectIn, RedirectOut, IPv4UcastLPM, and IPv4NextHop were used. RedirectIn and RedirectOut LFBs redirected OSPF Hello and LSA packets from and to the CE. A SmartBits [SmartBits] test machine was used to simulate an OSPF router and exchange the OSPF Hello and LSA packets with the CE in the ForCES router.

In Figure 7, cases (a) and (b) both need a RedirectIn LFB to send OSPF packets generated by the CE to the FE by use of ForCES packet redirect messages. The OSPF packets were further sent to an outside OSPF router by the FE via forwarding LFBs, including IPv4NextHop and IPv4UcastLPM. A RedirectOut LFB in the FE was used to send OSPF packets received from outside the OSPF router to the CE by ForCES packet redirect messages.

By running OSPF, the CE in the ForCES router could generate new routes and load them to the routing table in the FE. The FE was then able to forward packets according to the routing table.

The test results are shown in Figure 10.

Test#	CE	FE(s)	Item	LFBs Related	Result
1	N	Z	IPv4NextHopTable SET	IPv4NextHop	Success
2	N	Z	IPv4PrefixTable SET	IPv4UcastLPM	Success
3	N	Z	Redirect OSPF packet from CE to SmartBits	RedirectIn	Success
4	N	Z	Redirect OSPF packet from SmartBits to CE	RedirectOut	Success
5	N	Z	Metadata in redirect message	RedirectOut RedirectIn	Success
6	N	Z	OSPF neighbor discovery	RedirectOut RedirectIn	Success
7	N	Z	OSPF DD exchange	RedirectOut RedirectIn IPv4NextHop	Success
8	N	Z	OSPF LSA exchange	RedirectOut RedirectIn IPv4NextHop IPv4UcastLPM	Success
9	N	Z	Data Forwarding	RedirectOut RedirectIn IPv4NextHop IPv4UcastLPM	Success
10	Z	N	IPv4NextHopTable SET	IPv4NextHop	Success
11	Z	N	IPv4PrefixTable SET	IPv4UcastLPM	Success
12	Z	N	Redirect OSPF packet from CE to other OSPF router	RedirectIn	Success
13	Z	N	Redirect OSPF packet from other OSPF router to CE	RedirectOut	Success
14	Z	N	Metadata in redirect message	RedirectOut RedirectIn	Success
15	Z	N	OSPF neighbor discovery	RedirectOut RedirectIn	Success

16	Z	N	OSPF DD exchange	RedirectOut RedirectIn IPv4NextHop	Failure	
17	Z	N	OSPF LSA exchange	RedirectOut RedirectIn IPv4NextHop IPv4UcastLPM	Failure	

Figure 10: Test Results for Packet Forwarding

Note on tests #3 to #9:

During the test, OSPF packets received from the CE were found by Ethereal/Wireshark to have checksum errors in the FE. Because the test time was quite limited, the implementer of the CE did not make an effort to find and solve the checksum error; instead, the FE had tried to correct the checksum in order to not let the SmartBits drop the packets. Note that such a solution does not affect the test results.

Comment on tests #16 and #17:

The two test items failed. Note that tests #7 and #8 were identical to tests #16 and #17, only with CE and FE implementers being exchanged. Moreover, tests #12 and #13 showed that the redirect channel worked well. Therefore, it can be reasonably inferred that the problem caused by the failure was from the implementations, rather than from the ForCES protocol itself or the misunderstanding of implementations on the protocol specification. Although the failure made the OSPF interoperability test incomplete, it did not show an interoperability problem. More test work is needed to verify the OSPF interoperability.

5. Discussions

5.1. On Data Encapsulation Format

On the first day of the test, it was found that the LFB interoperations pertaining to tables all failed. It was eventually found that the failure occurred because different data encapsulation methods for ForCES protocol messages were used by different implementations. The issue is described in detail below.

Assuming that an LFB has two components, one is a struct with ID=1 and the other is an array with ID=2; in addition, both have two components of u32 inside, as shown below:

```
struct1: type struct, ID=1
        components are:
        a, type u32, ID=1
        b, type u32, ID=2

table1: type array, ID=2
        components for each row are (a struct of):
        x, type u32, ID=1
        y, type u32, ID=2
```

1. On Response of PATH-DATA-TLV Format

When a CE sends a config/query ForCES protocol message to an FE from a different implementer, the CE probably receives a response from the FE with a different PATH-DATA-TLV encapsulation format. For example, if a CE sends a query message with a path of 1 to a third-party FE to manipulate struct1 as defined above, it is probable that the FE will generate a response with two different PATH-DATA-TLV encapsulation formats: one is the value with FULLDATA-TLV/SPARSEDATA-TLV and the other is the value with many parallel PATH-DATA-TLVs and nested PATH-DATA-TLVs, as shown below:

```
format 1:
    OPER = GET-RESPONSE-TLV
        PATH-DATA-TLV:
            IDs=1
            FULLDATA-TLV containing valueof(a),valueof(b)
format 2:
    OPER = GET-RESPONSE-TLV
        PATH-DATA-TLV:
            IDs=1
            PATH-DATA-TLV:
                IDs=1
                FULLDATA-TLV containing valueof(a)
            PATH-DATA-TLV:
                IDs=2
                FULLDATA-TLV containing valueof(b)
```

The interoperability testers witnessed that a ForCES element (CE or FE) sender is free to choose whatever data structure that IETF ForCES documents define and best suits the element, while a ForCES element (CE or FE) should be able to accept and process information (requests and responses) that use any legitimate structure defined by IETF ForCES documents. While in the case where a ForCES element is free

to choose any legitimate data structure as a response, it is preferred that the ForCES element responds in the same format that the request was made, as it is most likely the data structure that the request sender looks to receive.

2. On Operation to Array

An array operation may also have several different data encapsulation formats. For instance, if a CE sends a config message to table1 with a path of (2.1), which refers to the component with ID=2 (an array), and the second ID is the row, then row 1 may be encapsulated with three formats as shown below:

```
format 1:
    OPER = SET-TLV
        PATH-DATA-TLV:
            IDs=2.1
            FULLDATA-TLV containing valueof(x),valueof(y)
format 2:
    OPER = SET-TLV
        PATH-DATA-TLV:
            IDs=2.1
            PATH-DATA-TLV:
                IDs=1
                FULLDATA-TLV containing valueof(x)
            PATH-DATA-TLV
                IDs=2
                FULLDATA-TLV containing valueof(y)
```

Moreover, if the CE is targeting the whole array, for example, if the array is empty and the CE wants to add the first row to the table, it could also adopt another format:

```
format 3:
    OPER = SET-TLV
        PATH-DATA-TLV:
            IDs=2
            FULLDATA-TLV containing rowindex=1,valueof(x),valueof(y)
```

The interoperability test experience has shown that formats 1 and 3, which take full advantage of the multiple data elements description in one TLV of FULLDATA-TLV, are more efficient, although format 2 can also achieve the same operating goal.

6. Security Considerations

Developers of ForCES FEs and CEs must take the security considerations of the ForCES Framework [RFC3746] and ForCES Protocol Specification [RFC5810] into account. Also, as specified in the security considerations of SCTP-Based TML for the ForCES Protocol [RFC5811], the transport-level security has to be ensured by IPsec. Test results of TML with IPsec supported have been shown in Section 4.2 in this document.

The tests described in this document used only simple password security mode. Testing using more sophisticated security is for future study.

Further testing using key agility is encouraged. The tests reported here used SCTP TML running over an IPsec tunnel, which was established by Racoon. Key negotiation formed part of this process, but we believe that the SCTP TML used does not include key agility or renegotiation.

7. References

7.1. Normative References

[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.

[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.

[RFC5812] Halpern, J. and J. Hadi Salim, "Forwarding and Control Element Separation (ForCES) Forwarding Element Model", RFC 5812, March 2010.

[RFC5813] Haas, R., "Forwarding and Control Element Separation (ForCES) MIB", RFC 5813, March 2010.

7.2. Informative References

[CEHA] Ogawa, K., Wang, W., Haleplidis, E., and J. Salim, "ForCES Intra-NE High Availability", Work in Progress, October 2010.

[Ethereal] Fenggen, J., "Subject: Release of a test version of ForCES dissector based on Ethereal 0.99.0", message to the IETF forces mailing list, 11 June 2009, <http://www.ietf.org/mail-archive/web/forces/current/msg03687.html>.

[LFB-LIB] Wang, W., Haleplidis, E., Ogawa, K., Li, C., and J. Halpern, "ForCES Logical Function Block (LFB) Library", Work in Progress, December 2010.

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

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

[RFC6053] Haleplidis, E., Ogawa, K., Wang, W., and J. Hadi Salim, "Implementation Report for Forwarding and Control Element Separation (ForCES)", RFC 6053, November 2010.

[RFC6956] Wang, W., Haleplidis, E., Ogawa, K., Li, C., and J. Halpern, "Forwarding and Control Element Separation (ForCES) Logical Function Block (LFB) Library", RFC 6956, June 2013.

[Racoon] The NetBSD Foundation, "How to build a remote user access VPN with Racoon", <http://www.netbsd.org/docs/network/ipsec/rasvpn.html>.

[SmartBits] Spirent Inc., "The Highly-Scalable Router Performance Tester: TeraRouting Tester", 2005, <http://www.spirent.com/~/media/Datasheets/Broadband/Obsolete_SMB-TM/TeraRouting%20Tester.pdf>.

[Tcpdump] Hadi Salim, J., "Subject: tcpdump 4.1.1", message to the IETF forces mailing list, 20 May 2010, <http://www.ietf.org/mail-archive/web/forces/current/msg03811.html>.

[TeamViewer] TeamViewer Inc., "TeamViewer - the All-In-One Software for Remote Support and Online Meetings", <http://www.teamviewer.com/>.

Appendix A. Acknowledgements

The authors thank the following test participants:

Chuanhuang Li, Hangzhou BAUD Networks
Ligang Dong, Zhejiang Gongshang University
Bin Zhuge, Zhejiang Gongshang University
Jingjing Zhou, Zhejiang Gongshang University
Liaoyuan Ke, Hangzhou BAUD Networks
Kelei Jin, Hangzhou BAUD Networks

The authors also thank very much Adrian Farrel, Joel Halpern, Ben Campbell, Nevil Brownlee, and Sean Turner for their important help in the document publication process.

Appendix B. Contributors

Contributors who have made major contributions to the interoperability test are listed below.

Hirofumi Yamazaki
NTT Corporation
Tokyo
Japan
EMail: yamazaki.horofumi@lab.ntt.co.jp

Rong Jin
Zhejiang Gongshang University
Hangzhou
P.R. China
EMail: jinrong@zjsu.edu.cn

Yuta Watanabe
NTT Corporation
Tokyo
Japan
EMail: yuta.watanabe@ntt-at.co.jp

Xiaochun Wu
Zhejiang Gongshang University
Hangzhou
P.R. China
EMail: spring-403@zjsu.edu.cn

Authors' Addresses

Weiming Wang
Zhejiang Gongshang University
18 Xuezheng Str., Xiasha University Town
Hangzhou 310018
P.R. China

Phone: +86-571-28877721
EMail: wmwang@zjsu.edu.cn

Kentaro Ogawa
NTT Corporation
Tokyo
Japan

EMail: ogawa.kentaro@lab.ntt.co.jp

Evangelos Haleplidis
University of Patras
Department of Electrical & Computer Engineering
Patras 26500
Greece

EMail: ehalep@ece.upatras.gr

Ming Gao
Hangzhou BAUD Networks
408 Wen-San Road
Hangzhou 310012
P.R. China

EMail: gaoming@mail.zjgsu.edu.cn

Jamal Hadi Salim
Mojatatu Networks
Ottawa
Canada

EMail: hadi@mojatatu.com