Network Working Group	J. Hadi Salim
Internet-Draft	Mojatatu Networks
Expires: January 2, 2010	K. Ogawa
	NTT Corporation
	July 01, 2009

TOC

SCTP based TML (Transport Mapping Layer) for ForCES protocol
draft-ietf-forces-sctptml-04

Status of this Memo

Copyright Notice

Abstract

This document defines the SCTP based TML (Transport Mapping Layer) for the ForCES protocol. It explains the rationale for choosing the SCTP (Stream Control Transmission Protocol) [RFC4960] (Stewart, R., “Stream Control Transmission Protocol,” September 2007.) and also describes how this TML addresses all the requirements described in [RFC3654] (Khosravi, H. and T. Anderson, “Requirements for Separation of IP Control and Forwarding,” November 2003.) and the ForCES protocol

[FE-PROTO] (Doria (Ed.), A., Haas (Ed.), R., Hadi Salim (Ed.), J., Khosravi (Ed.), H., M. Wang (Ed.), W., Dong, L., and R. Gopal, “ForCES Protocol Specification,” November 2008.) draft.

Table of Contents

---

1. Definitions

TOC

The following definitions are taken from [RFC3654] (Khosravi, H. and T. Anderson, “Requirements for Separation of IP Control and Forwarding,” November 2003.)and [RFC3746] (Yang, L., Dantu, R., Anderson, T., and R. Gopal, “Forwarding and Control Element Separation (ForCES) Framework,” April 2004.):

ForCES Protocol -- The protocol used at the Fp reference point in the ForCES Framework in [RFC3746] (Yang, L., Dantu, R., Anderson, T., and R. Gopal, “Forwarding and Control Element Separation (ForCES) Framework,” April 2004.).

ForCES Protocol Layer (ForCES PL) -- A layer in ForCES protocol architecture that defines the ForCES protocol architecture and the state transfer mechanisms as defined in [FE-PROTO] (Doria (Ed.), A., Haas (Ed.), R., Hadi Salim (Ed.), J., Khosravi (Ed.), H., M. Wang (Ed.), W., Dong, L., and R. Gopal, “ForCES Protocol Specification,” November 2008.).

ForCES Protocol Transport Mapping Layer (ForCES TML) -- A layer in ForCES protocol architecture that specifically addresses the protocol message transportation issues, such as how the protocol messages are mapped to different transport media (like SCTP, IP, ATM, Ethernet, etc), and how to achieve and implement reliability, security, etc.

2. Introduction

TOC

The ForCES (Forwarding and Control Element Separation) working group in the IETF defines the architecture and protocol for separation of Control Elements(CE) and Forwarding Elements(FE) in Network Elements(NE) such as routers. [RFC3654] (Khosravi, H. and T. Anderson, “Requirements for Separation of IP Control and Forwarding,” November 2003.) and [RFC3746] (Yang, L., Dantu, R., Anderson, T., and R. Gopal, “Forwarding and Control Element Separation (ForCES) Framework,” April 2004.) respectively define architectural and protocol requirements for the communication between CE and FE. The ForCES protocol layer specification [FE-PROTO] (Doria (Ed.), A., Haas (Ed.), R., Hadi Salim (Ed.), J., Khosravi (Ed.), H., M. Wang (Ed.), W., Dong, L., and R. Gopal, “ForCES Protocol Specification,” November 2008.) describes the protocol semantics and workings. The ForCES protocol layer operates on top of an inter-connect hiding layer known as the TML. The relationship is illustrated in Figure 1 (Message exchange between CE and FE to establish an NE association).

This document defines the SCTP based TML for the ForCES protocol layer. It also addresses all the requirements for the TML including security, reliability, etc as defined in [FE-PROTO] (Doria (Ed.), A., Haas (Ed.), R., Hadi Salim (Ed.), J., Khosravi (Ed.), H., M. Wang (Ed.), W., Dong, L., “ForCES Protocol Specification,” November 2008.).

TOC

3. Protocol Framework Overview

The reader is referred to the Framework document [RFC3746] (Yang, L., Dantu, R., Anderson, T., and R. Gopal, “Forwarding and Control Element Separation (ForCES) Framework,” April 2004.), and in particular sections 3 and 4, for an architectural overview and explanation of where and how the ForCES protocol fits in.
There is some content overlap between the ForCES protocol draft [FE-PROTO] (Doria (Ed.), A., Haas (Ed.), R., Hadi Salim (Ed.), J., Khosravi (Ed.), H., M. Wang (Ed.), W., Dong, L., and R. Gopal, “ForCES Protocol Specification,” November 2008.) and this section (Section 3 (Protocol Framework Overview)) in order to provide basic context to the reader of this document.
The ForCES protocol layering constitutes two pieces: the PL and TML layer. This is depicted in Figure 1 (Message exchange between CE and FE to establish an NE association).

```
+-----------------------------------------------+
|                    CE PL                      |
+-----------------------------------------------+
|                    CE TML                     |
+-----------------------------------------------+
                       ^
                       |
            ForCES PL  | messages
                       |
                       v
+------------------------------------------------+
|                   FE TML                       |
+------------------------------------------------+
|                   FE PL                        |
+------------------------------------------------+
```

Figure 1: Message exchange between CE and FE to establish an NE association

The PL is in charge of the ForCES protocol. Its semantics and message layout are defined in [FE-PROTO] (Doria (Ed.), A., Haas (Ed.), R., Hadi Salim (Ed.), J., Khosravi (Ed.), H., M. Wang (Ed.), W., Dong, L., and R. Gopal, “ForCES Protocol Specification,” November 2008.). The TML is necessary to connect two ForCES end-points as shown in Figure 1 (Message exchange between CE and FE to establish an NE association).

Both the PL and TML are standardized by the IETF. While only one PL is defined, different TMLs are expected to be standardized. The TML at each of the nodes (CE and FE) is expected to be of the same definition in order to inter-operate.
When transmitting from a ForCES end-point, the PL delivers its messages to the TML. The TML then delivers the PL message to the destination TML(s).
On reception of a message, the TML delivers the message to its destination PL level (as described in the ForCES header).

3.1. The PL

TOC

The PL is common to all implementations of ForCES and is standardized by the IETF [FE-PROTO] (Doria (Ed.), A., Haas (Ed.), R., Hadi Salim (Ed.), J., Khosravi (Ed.), H., M. Wang (Ed.), W., Dong, L., and R. Gopal, “ForCES Protocol Specification,” November 2008.). The PL level is responsible for associating an FE or CE to an NE. It is also responsible for tearing down such associations.
An FE may use the PL level to asynchronously send packets to the CE. The FE may redirect via the PL (from outside the NE) various control protocol packets (e.g. OSPF, etc) to the CE. Additionally, the FE delivers various events that CE has subscribed-to via PL [FE-MODEL] (Halpern, J. and J. Hadi Salim, “ForCES Forwarding Element Model,” October 2008.).
The CE and FE may interact synchronously via the PL. The CE issues status requests to the FE and receives responses via the PL. The CE also configures the associated FE's LFBs' components using the PL [FE-MODEL] (Halpern, J. and J. Hadi Salim, “ForCES Forwarding Element Model,” October 2008.).

3.2. The TML

TOC

The TML level is responsible for transport of the PL level messages. [FE-PROTO] (Doria (Ed.), A., Haas (Ed.), R., Hadi Salim (Ed.), J., Khosravi (Ed.), H., M. Wang (Ed.), W., Dong, L., and R. Gopal, “ForCES Protocol Specification,” November 2008.) section 5 defines the requirements that need to be met by a TML specification. The SCTP TML specified in this document meets all the requirements specified in [FE-PROTO] (Doria (Ed.), A., Haas (Ed.), R., Hadi Salim (Ed.), J., Khosravi (Ed.), H., M. Wang (Ed.), W., Dong, L., and R. Gopal, “ForCES Protocol Specification,” November 2008.) section 5. Section 4.2.2 (Satisfying TML Requirements) describes how the TML requirements are met.

3.2.1. TML and PL Interfaces

TOC

There are two interfaces to the PL and TML, both of which are out of scope for ForCES. The first one is the interface between the PL and TML and the other is the CE Manager (CEM)/FE Manager (FEM)[RFC3746] (Yang, L., Dantu, R., Anderson, T., and R. Gopal, “Forwarding and Control Element Separation (ForCES) Framework,” April 2004.) interface to both the PL and TML. Both interfaces are shown in Figure 2 (The TML-PL interface).

```
                        +----------------------------+
                        |  +----------------------+  |
                        |  |                      |  |
       +---------+      |  |       PL Layer       |  |
       |         |      |  +----------------------+  |
       |FEM/CEM  |<---->|             ^              |
       |         |      |             |              |
       +---------+      |             |TML API       |
                        |             |              |
                        |             V              |
                        |  +----------------------+  |
                        |  |                      |  |
                        |  |       TML Layer      |  |
                        |  |                      |  |
                        |  +----------------------+  |
                        +----------------------------+
```

Figure 2: The TML-PL interface

Figure 2 (The TML-PL interface) also shows an interface referred to as CEM/FEM[RFC3746] (Yang, L., Dantu, R., Anderson, T., and R. Gopal, “Forwarding and Control Element Separation (ForCES) Framework,” April 2004.) which is responsible for bootstrapping and parameterization of the TML. In its most basic form the CEM/FEM interface takes the form of a simple static config file which is read on startup in the pre-association phase.
Appendix B (Service Interface) discusses in more details the service interfaces.

3.2.2. TML Parameterization

TOC

It is expected that it should be possible to use a configuration reference point, such as the FEM or the CEM, to configure the TML. Some of the configured parameters may include:

- PL ID
- Connection Type and associated data. For example if a TML uses IP/SCTP then parameters such as SCTP ports and IP addresses need to be configured.
- Number of transport connections
- Connection Capability, such as bandwidth, etc.
- Allowed/Supported Connection QoS policy (or Congestion Control Policy)

4. SCTP TML overview

TOC

SCTP [RFC4960] (Stewart, R., “Stream Control Transmission Protocol,” September 2007.) is an end-to-end transport protocol that is equivalent to TCP, UDP, or DCCP in many aspects. With a few exceptions, SCTP can do most of what UDP, TCP, or DCCP can achieve. SCTP as well can do most of what a combination of the other transport protocols can achieve (e.g. TCP and DCCP or TCP and UDP).
Like TCP, it provides ordered, reliable, connection-oriented, flow-controlled, congestion controlled data exchange. Unlike TCP, it does not provide byte streaming and instead provides message boundaries.
Like UDP, it can provide unreliable, unordered data exchange. Unlike UDP, it does not provide multicast support
Like DCCP, it can provide unreliable, ordered, congestion controlled, connection-oriented data exchange.
SCTP also provides other services that none of the 3 transport protocols mentioned above provide. These include:

- Multi-homing
  An SCTP connection can make use of multiple destination IP addresses to communicate with its peer.

- Runtime IP address binding
  With the SCTP Dynamic Address Reconfiguration ([RFC5061] (Stewart, R., Xie, Q., Tuexen, M., Maruyama, S., and M. Kozuka, “Stream Control Transmission Protocol (SCTP) Dynamic Address

Reconfiguration," September 2007.)) feature, a new IP address can be bound at runtime. This allows for migration of endpoints without restarting the association (valuable for high availability).

*A range of reliability shades with congestion control
SCTP offers a range of services from full reliability to none, and from full ordering to none. With SCTP, on a per message basis, the application can specify a message's time-to-live. When the expressed time expires, the message can be "skipped".

*Built-in heartbeats
SCTP has built-in heartbeat mechanism that validate the reachability of peer addresses.

*Multi-streaming
A known problem with TCP is head of line (HOL) blocking. If you have independent messages, TCP enforces ordering of such messages. Loss at the head of the messages implies delays of delivery of subsequent packets. SCTP allows for defining up to 64K independent streams over the same socket connection, which are ordered independently.

*Message boundaries with reliability
SCTP allows for easier message parsing (just like UDP but with reliability built in) because it establishes boundaries on a PL message basis. On a TCP stream, one would have to use techniques such peeking into the message to figure the boundaries.

*Improved SYN DOS protection
Unlike TCP, which does a 3 way connection setup handshake, SCTP does a 4 way handshake. This improves against SYN-flood attacks because listening sockets do not set up state until a connection is validated.

*Simpler transport events
An application (such as the TML) can subscribe to be notified of both local and remote transport events. Events that can be subscribed-to include indication of association changes, addressing changes, remote errors, expiry of timed messages, etc. These events are off by default and require explicit subscription.

*Simplified replicasting
Although SCTP does not allow for multicasting it allows for a single message from an application to be sent to multiple peers. This reduces the messaging that typically crosses different memory domains within a host (example in a kernel to user space domain of an operating system).

4.1. Rationale for using SCTP for TML

TOC

SCTP has all the features required to provide a robust TML. As a transport that is all-encompassing, it negates the need for having multiple transport protocols in order to satisfy the TML requirements ([FE-PROTO] (Doria (Ed.), A., Haas (Ed.), R., Hadi Salim (Ed.), J., Khosravi (Ed.), H., M. Wang (Ed.), W., Dong, L., and R. Gopal, “ForCES Protocol Specification,” November 2008.) section 5). As a result it allows for simpler coding and therefore reduces a lot of the interoperability concerns.
SCTP is also very mature and widely used making it a good choice for ubiquitous deployment.

4.2. Meeting TML requirements

TOC

```
PL
+----------------------+
|                      |
+-----------+----------+
            |   TML API
 TML        |
+-----------+----------+
|           |          |
|    +------+------+   |
|    |  TML core   |   |
|    +-+----+----+-+   |
|      |    |    |     |
|    SCTP socket API   |
|      |    |    |     |
|      |    |    |     |
|    +-+----+----+-+   |
|    |    SCTP     |   |
|    +------+------+   |
|           |          |
|           |          |
|    +------+------+   |
|    |      IP     |   |
|    +-------------+   |
+----------------------+
```

Figure 3: The TML-SCTP interface

Figure 3 (The TML-SCTP interface) details the interfacing between the PL and SCTP TML and the internals of the SCTP TML. The core of the TML interacts on its north-bound interface to the PL (utilizing the TML API). On the south-bound interface, the TML core interfaces to the SCTP layer utilizing the standard socket interface[SCTP-API] (Stewart, R., Poon, K., Tuexen, M., Yasevich, V., and P. Lei, “Sockets API Extensions for Stream Control Transmission Protocol (SCTP),” Feb. 2009.) There are three SCTP socket connections opened between any two PL endpoints (whether FE or CE).

4.2.1. SCTP TML Channels

TOC

```
                    +--------------------+
                    |                    |
                    |     TML   core     |
                    |                    |
                    +-+-------+--------+-+
                      |       |        |
                      |   Med prio,    |
                      |  Semi-reliable |
                      |    channel     |
                      |       |      Low prio,
                      |       |      Unreliable
                      |       |      channel
                      |       |        |
                      ^       ^        ^
                      |       |        |
                      Y       Y        Y
            High prio,|       |        |
             reliable |       |        |
              channel |       |        |
                      Y       Y        Y
                   +-+--------+--------+-+
                   |                     |
                   |        SCTP         |
                   |                     |
                   +---------------------+
```

Figure 4: The TML-SCTP channels

Figure 4 (The TML-SCTP channels) details further the interfacing between the TML core and SCTP layers. There are 3 channels used to separate and prioritize the different types of ForCES traffic. Each channel constitutes a socket interface. It should be noted that all SCTP channels are congestion aware (and for that reason that detail is left out of the description of the 3 channels). SCTP port 6700, 6701, 6702 are used for the higher, medium and lower priority channels respectively.

4.2.1.1. Justifying Choice of 3 Sockets

TOC

SCTP allows up to 64K streams to be sent over a single socket interface. The authors initially envisioned using a single socket for all three channels (mapping a channel to an SCTP stream). This

simplifies programming of the TML as well as conserves use of SCTP ports.
Further analysis revealed head of line blocking issues with this initial approach. Lower priority packets not needing reliable delivery could block higher priority packets (needing reliable delivery) under congestion situation for an indeterminate period of time (depending on how many outstanding lower priority packets are pending). For this reason, we elected to go with mapping each of the three channels to a different SCTP socket (instead of a different stream within a single socket).

4.2.1.2. Higher Priority, Reliable channel

TOC

The higher priority (HP) channel uses a standard SCTP reliable socket on port 6700. It is used for CE solicited messages and their responses:

1. ForCES configuration messages flowing from CE to FE and responses from the FE to CE.

2. ForCES query messages flowing from CE to FE and responses from the FE to the CE.

It is recommended that PL priorities 4-7 be used for this channel and that the following PL messages use the HP channel for transport:

*Association Setup

*Association Setup Response

*Association Teardown

*Config

*Config Response

*Query

*Query Response

4.2.1.3. Medium Priority, Semi-Reliable channel

TOC

The medium priority (MP) channel uses SCTP-PR on port 6701. Time limits on how long a message is valid are set on each outgoing message. This channel is used for events from the FE to the CE that are obsoleted

over time. Events that are accumulative in nature and are recoverable by the CE (by issuing a query to the FE) can tolerate lost events and therefore should use this channel. For example, a generated event which carries the value of a counter that is monotonically incrementing fits to use this channel.
It is recommended that PL priorities 2-3 be used for this channel and that the following PL messages use the MP channel for transport:

*Event Notification

4.2.1.4. Lower Priority, Unreliable channel

TOC

The lower priority (LP) channel uses SCTP port 6702. This channel also uses SCTP-PR with lower timeout values than the MP channel. The reason an unreliable channel is used for redirect messages is to allow the control protocol at both the CE and its peer-endpoint to take charge of how the end-to-end semantics of the said control protocol's operations. For example:

1. Some control protocols are reliable in nature, therefore making this channel reliable introduces an extra layer of reliability which could be harmful. So any end-to-end retransmits will happen from remote.

2. Some control protocols may desire to have obsolescence of messages over retransmissions; making this channel reliable contradicts that desire.

Given ForCES PL level heartbeats are traffic sensitive, sending them over the LP channel also makes sense. If the other end is not processing other channels it will eventually get heartbeats; and if it is busy processing other channels heartbeats will be obsoleted locally over time (and it does not matter if they did not make it).
It is recommended that PL priorities 0-1 be used for this channel and that that the following PL messages use the LP channel for transport:

*Packet Redirect

*Heartbeats

TOC

4.2.1.5. Scheduling of The 3 Channels

Strict priority work-conserving scheduling is used to process both on sending and receiving (of the PL messages) by the TML Core as shown in Figure 5 (SCTP TML Strict Priority Scheduling).

This means that the HP messages are always processed first until there are no more left. The LP channel is processed only if a channel that is higher priority than itself has no more messages left to process. This means that under congestion situation, a higher priority channel with sufficient messages that occupy the available bandwidth would starve lower priority channel(s).

The design intent of the SCTP TML is to tie prioritization as described in Section 4.2.1.1 (Justifying Choice of 3 Sockets) and transport congestion control to provide implicit node congestion control. This is further detailed in Appendix A.2 (Channel work scheduling).

```
    SCTP channel            +----------+
    Work available          |   DONE   +---<--<--+
        |                   +---+------+         |
        Y                                        ^
        |         +-->--+         +-->---+       |
+-->-->-+         |     |         |      |       |
|       |         |     |         |      |       ^
|       ^         ^     Y         ^      Y       |
^      / \        |     |         |      |       |
|     /   \       |     ^         |      ^       ^
|    / Is  \      |    / \        |     / \      |
|   / there \     |   /Is \       |    /Is \     |
^  / HP work \    ^  /there\      ^   /there\    ^
|  \    ?    /    | /MP work\     |  /LP work\   |
|   \       /     | \    ?  /     |  \   ?   /   |
|    \     /      |  \     /      |   \     /    ^
|     \   /       ^   \   /       ^    \   /     |
|      \ /        |    \ /        |     \ /      |
^       Y-->-->-->+     Y-->-->-->+      Y->->->-+
|       |    NO         |    NO          |  NO
|       |               |                |
|       Y               Y                Y
|       | YES           | YES            |
^       |               |                |
|       Y               Y                Y
|  +----+------+    +---|-------+   +----|------+
|  |- process  |    |- process  |   |- process  |
|  |  HP work  |    |  MP work  |   | LP work   |
|  +------+----+    +-----+-----+   +-----+-----+
|         |               |               |
^         Y               Y               Y
|         |               |               |
|         Y               Y               Y
+--<--<---+--<--<----<----+-----<---<-----+
```

Figure 5: SCTP TML Strict Priority Scheduling

TOC

4.2.1.6. SCTP TML Parameterization

The following is a list of parameters needed for booting the TML. It is expected these parameters will be extracted via the FEM/CEM interface for each PL ID.

1. The IP address or a resolvable DNS/hostname of the CE/FE.

2. Whether to use IPsec or not. If IPsec is used, how to parameterize the different required ciphers, keys etc as described in Section 7.1 (IPsec Usage)

3. The HP SCTP port, as discussed in Section 4.2.1.2 (Higher Priority, Reliable channel). The default HP port value is 6700 (Section 6 (IANA Considerations)).

4. The MP SCTP port, as discussed in Section 4.2.1.3 (Medium Priority, Semi-Reliable channel). The default MP port value is 6701 (Section 6 (IANA Considerations)).

5. The LP SCTP port, as discussed in Section 4.2.1.4 (Lower Priority, Unreliable channel). The default LP port value is 6702 (Section 6 (IANA Considerations)).

4.2.2. Satisfying TML Requirements

TOC

[FE-PROTO] (Doria (Ed.), A., Haas (Ed.), R., Hadi Salim (Ed.), J., Khosravi (Ed.), H., M. Wang (Ed.), W., Dong, L., and R. Gopal, “ForCES Protocol Specification,” November 2008.) section 5 lists requirements that a TML needs to meet. This section describes how the SCTP TML satisfies those requirements.

4.2.2.1. Satisfying Reliability Requirement

TOC

As mentioned earlier, a shade of reliability ranges is possible in SCTP. Therefore this requirement is met.

4.2.2.2. Satisfying Congestion Control Requirement

TOC

Congestion control is built into SCTP. Therefore, this requirement is met.

4.2.2.3. Satisfying Timeliness and Prioritization Requirement

TOC

By using 3 sockets in conjunction with the partial-reliability feature, both timeliness and prioritization can be achieved.

4.2.2.4. Satisfying Addressing Requirement

TOC

There are no extra headers required for SCTP to fulfil this requirement. SCTP can be told to replicast packets to multiple destinations. The TML implementation will need to translate PL level addresses, to a variety of unicast IP addresses in order to emulate multicast and broadcast PL addresses.

4.2.2.5. Satisfying HA Requirement

TOC

Transport link resiliency is one of SCTP's strongest point. Failure detection and recovery is built in, as mentioned earlier.

- The SCTP multi-homing feature is used to provide path diversity. Should one of the peer IP addresses become unreachable, the other(s) are used without needing lower layer convergence (routing, for example) or even the TML becoming aware.

- SCTP heartbeats and data transmission thresholds are used on a per peer IP address to detect reachability faults. The faults could be a result of an unreachable address or peer, which may be caused by a variety of reasons, like interface, network, or endpoint failures. The cause of the fault is noted.

- With the ADDIP feature, one can migrate IP addresses to other nodes at runtime. This is not unlike the VRRP[RFC3768] (Hinden, R., “Virtual Router Redundancy Protocol (VRRP),” April 2004.) protocol use. This feature is used in addition to multi-homing in a planned migration of activity from one FE/CE to another. In such a case, part of the provisioning recipe at the CE for replacing an FE involves migrating activity of one FE to another.

TOC

4.2.2.6. Satisfying Node Overload Prevention Requirement

The architecture of this TML defines three separate channels, one per socket, to be used within any FE-CE setup. The scheduling design for processing the TML channels (Section 4.2.1.5 (Scheduling of The 3 Channels)) is strict priority. A fundamental desire of the strict prioritization is to ensure that more important work always gets node resources such as CPU and bandwidth over lesser important work.
When a ForCES node CPU is overwhelmed because the incoming packet rate is higher than it can keep up with, the channel queues grow and transport congestion subsequently follows. By virtue of using SCTP, the congestion is propagated back to the source of the incoming packets and eventually alleviated.
The HP channel work gets prioritized at the expense of the MP which gets prioritized over LP channels. The preferential scheduling only kicks in when there is node overload regardless of whether there is transport congestion. As a result of the preferential work treatment, the ForCES node achieves a robust steady processing capacity. Refer to Appendix A.2 (Channel work scheduling) for details on scheduling.
For an example of how the overload prevention works: consider a scenario where an overwhelming amount redirected packets (from outside the NE) coming into the NE may overload the FE while it has outstanding config work from the CE. In such a case, the FE, while it is busy processing config requests from the CE ignores processing the redirect packets on the LP channel. If enough redirect packets accumulate, they are dropped either because the LP channel threshold is exceeded or because they are obsoleted. If on the other hand, the FE has successfully processed the higher priority channels and their related work, then it can proceed and process the LP channel. So as demonstrated in this case, the TML ties transport and node overload implicitly together.

TOC

4.2.2.7. Satisfying Encapsulation Requirement

There is no extra encapsulation added by the SCTP TML.
In the future, should the need arise, a new SCTP extension/chunk can be defined to meet newer ForCES requirements [RFC4960] (Stewart, R., “Stream Control Transmission Protocol,” September 2007.).

TOC

5. SCTP TML Channel Work

There are two levels of TML channel work within an NE when a ForCES node (CE or FE) is connected to multiple other ForCES nodes:

1. NE-level I/O work where a ForCES node (CE or FE) needs to choose which of the peer nodes to process.

2. Node-level I/O work where a ForCES node, handles the three SCTP TML channels separately for each single ForCES endpoint.

NE-level scheduling definition is left up to the implementation and is considered out of scope for this document. Appendix A.4 (SCTP TML NE level channel scheduling) discuss briefly some constraints that an implementor needs to worry about.
This document provides suggestions on SCTP channel work implementation in Appendix A (SCTP TML Channel Work Implementation).
The FE SHOULD do channel connections to the CE in the order of incrementing priorities i.e. LP socket first, followed by MP and ending with HP socket connection. The CE, however, MUST NOT assume that there is ordering of socket connections from any FE.

6. IANA Considerations

TOC

This document makes request of IANA to reserve SCTP ports 6700, 6701, and 6702.

7. Security Considerations

TOC

The SCTP TML provides the following security services to the PL level:

- A mechanism to authenticate ForCES CEs and FEs at transport level in order to prevent the participation of unauthorized CEs and unauthorized FEs in the control and data path processing of a ForCES NE.

- A mechanism to ensure message authentication of PL data and headers transferred from the CE to FE (and vice-versa) in order to prevent the injection of incorrect data into PL messages.

- A mechanism to ensure the confidentiality of PL data and headers transferred from the CE to FE (and vice-versa), in order to prevent disclosure of PL level information transported via the TML.

Security choices provided by the TML are made by the operator and take effect during the pre-association phase of the ForCES protocol. An operator may choose to use all, some or none of the security services provided by the TML in a CE-FE connection.
When operating under a secured environment, or for other operational concerns (in some cases performance issues) the operator may turn off all the security functions between CE and FE.
IP Security Protocol (IPsec) [RFC4301] (Kent, S. and K. Seo, “Security Architecture for the Internet Protocol,” December 2005.) is used to provide needed security mechanisms.
IPsec is an IP level security scheme transparent to the higher-layer applications and therefore can provide security for any transport layer protocol. This gives IPsec the advantage that it can be used to secure everything between the CE and FE without expecting the TML implementation to be aware of the details.
The IPsec architecture is designed to provide message integrity and message confidentiality outlined in the TML security requirements ([FE-PROTO] (Doria (Ed.), A., Haas (Ed.), R., Hadi Salim (Ed.), J., Khosravi (Ed.), H., M. Wang (Ed.), W., Dong, L., and R. Gopal, “ForCES Protocol Specification,” November 2008.)). Mutual authentication and key exchange protocol are provided by Internet Key Exchange (IKE) [RFC4109] (Hoffman, P., “Algorithms for Internet Key Exchange version 1 (IKEv1),” May 2005.).

7.1. IPsec Usage

TOC

A ForCES FE or CE MUST support the following:

*Internet Key Exchange (IKE)[RFC4109] (Hoffman, P., “Algorithms for Internet Key Exchange version 1 (IKEv1),” May 2005.) with certificates for endpoint authentication.

*Transport Mode Encapsulating Security Payload (ESP)[RFC4303] (Kent, S., “IP Encapsulating Security Payload (ESP),” December 2005.).

*HMAC-SHA1-96 [RFC2404] (Madson, C. and R. Glenn, “The Use of HMAC-SHA-1-96 within ESP and AH,” November 1998.) for message integrity protection

*AES-CBC with 128-bit keys [RFC3602] (Frankel, S., Glenn, R., and S. Kelly, “The AES-CBC Cipher Algorithm and Its Use with IPsec,” September 2003.) for message confidentiality.

*Replay protection[RFC4301] (Kent, S. and K. Seo, “Security Architecture for the Internet Protocol,” December 2005.).

It is expected to be possible for the CE or FE to be operationally configured to negotiate other cipher suites and even use manual keying.

7.1.1. SAD and SPD setup

TOC

To minimize the operational configuration it is recommended that only the IANA issued SCTP protocol number(132) be used as a selector in the Security Policy Database (SPD) for ForCES. In such a case only a single SPD and SAD entry is needed.
It should be straightforward to extend such a policy to alternatively use the 3 SCTP TML port numbers as SPD selectors. But as noted above this choice will require increased number of SPD entries.
In scenarios where multiple IP addresses are used within a single association, and there is desire to configure different policies on a per IP address, then it is recommended to follow [RFC3554] (Bellovin, S., Ioannidis, J., Keromytis, A., and R. Stewart, “On the Use of Stream Control Transmission Protocol (SCTP) with IPsec,” July 2003.)

8. Acknowledgements

TOC

The authors would like to thank Joel Halpern, Michael Tuxen, Randy Stewart and Evangelos Haleplidis for engaging us in discussions that have made this draft better.

9. References

TOC

9.1. Normative References

TOC

[RFC2404]	Madson, C. and R. Glenn, “The Use of HMAC-SHA-1-96 within ESP and AH,” RFC 2404, November 1998 (TXT, HTML, XML).
[RFC3554]	Bellovin, S., Ioannidis, J., Keromytis, A., and R. Stewart, “On the Use of Stream Control Transmission Protocol (SCTP) with IPsec,” RFC 3554, July 2003 (TXT).
[RFC3602]	Frankel, S., Glenn, R., and S. Kelly, “The AES-CBC Cipher Algorithm and Its Use with IPsec,” RFC 3602, September 2003 (TXT).
[RFC4109]	Hoffman, P., “Algorithms for Internet Key Exchange version 1 (IKEv1),” RFC 4109, May 2005 (TXT).

[RFC4301]	Kent, S. and K. Seo, “Security Architecture for the Internet Protocol,” RFC 4301, December 2005 (TXT).
[RFC4303]	Kent, S., “IP Encapsulating Security Payload (ESP),” RFC 4303, December 2005 (TXT).
[RFC4960]	Stewart, R., “Stream Control Transmission Protocol,” RFC 4960, September 2007 (TXT).
[RFC5061]	Stewart, R., Xie, Q., Tuexen, M., Maruyama, S., and M. Kozuka, “Stream Control Transmission Protocol (SCTP) Dynamic Address Reconfiguration,” RFC 5061, September 2007 (TXT).

9.2. Informative References

TOC

[FE-MODEL]	Halpern, J. and J. Hadi Salim, “ForCES Forwarding Element Model,” October 2008.
[FE-PROTO]	Doria (Ed.), A., Haas (Ed.), R., Hadi Salim (Ed.), J., Khosravi (Ed.), H., M. Wang (Ed.), W., Dong, L., and R. Gopal, “ForCES Protocol Specification,” November 2008.
[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).
[RFC3768]	Hinden, R., “Virtual Router Redundancy Protocol (VRRP),” RFC 3768, April 2004 (TXT).
[SCTP-API]	Stewart, R., Poon, K., Tuexen, M., Yasevich, V., and P. Lei, “Sockets API Extensions for Stream Control Transmission Protocol (SCTP),” Feb. 2009.

Appendix A. SCTP TML Channel Work Implementation

TOC

As mentioned in Section 5 (SCTP TML Channel Work), there are two levels of TML channel work within an NE when a ForCES node (CE or FE) is connected to multiple other ForCES nodes:

1. NE-level I/O work where a ForCES node (CE or FE) needs to choose which of the peer nodes to process.

2. Node-level I/O work where a ForCES node, handles the three SCTP TML channels separately for each single ForCES endpoint.

NE-level scheduling definition is left up to the implementation and is considered out of scope for this document. Appendix A.4 (SCTP TML NE

level channel scheduling) discuss briefly some constraints that an implementor needs to worry about.
This document and in particular Appendix A.1 (SCTP TML Channel Initialization), Appendix A.2 (Channel work scheduling) and Appendix A.3 (SCTP TML Channel Termination) discuss details of node-level I/O work.

A.1. SCTP TML Channel Initialization

TOC

As discussed in Section 5 (SCTP TML Channel Work), it is recommended that the FE SHOULD do socket connections to the CE in the order of incrementing priorities i.e. LP socket first, followed by MP and ending with HP socket connection. The CE, however, MUST NOT assume that there is ordering of socket connections from any FE. Appendix B.1 (TML Boot-strapping) has more details on the expected initialization of SCTP channel work.

A.2. Channel work scheduling

TOC

This section provides high level details of the scheduling view of the SCTP TML core (Section 4.2.1 (SCTP TML Channels)). A practical scheduler implementation takes care of many little details (such as timers, work quanta, etc) not described in this document. The implementor is left to take care of those details.
The CE(s) and FE(s) are coupled together in the principles of the scheduling scheme described here to tie together node overload with transport congestion. The design intent is to provide the highest possible robust work throughput for the NE under any network or processing congestion.

A.2.1. FE Channel work scheduling

TOC

The FE scheduling, in priority order, needs to I/O process:

1. The HP channel I/O in the following priority order:
    1. Transmitting back to the CE any outstanding result of executed work via the HP channel transmit path.
    2. Taking new incoming work from the CE which creates ForCES work to be executed by the FE.

2. ForCES events which result in transmission of unsolicited ForCES packets to the CE via the MP channel.

3. Incoming Redirect work in the form of control packets that come from the CE via LP channel. After redirect processing, these packets get sent out on external (to the NE) interface.

4. Incoming Redirect work in the form of control packets that come from other NEs via external (to the NE) interfaces. After some processing, such packets are sent to the CE.

It is worth emphasizing at this point again that the SCTP TML processes the channel work in strict priority. For example, as long as there are messages to send to the CE on the HP channel, they will be processed first until there are no more left before processing the next priority work (which is to read new messages on the HP channel incoming from the CE).

A.2.2. CE Channel work scheduling

TOC

The CE scheduling, in priority order, needs to deal with:

1. The HP channel I/O in the following priority order:

    1. Process incoming responses to requests of work it made to the FE(s).

    2. Transmitting any outstanding HP work it needs for the FE(s) to complete.

2. Incoming ForCES events from the FE(s) via the MP channel.

3. Outgoing Redirect work in the form of control packets that get sent from the CE via LP channel destined to external (to the NE) interface on FE(s).

4. Incoming Redirect work in the form of control packets that come from other NEs via external (to the NE) interfaces on the FE(s).

It is worth to repeat for emphasis again that the SCTP TML processes the channel work in strict priority. For example, if there are messages incoming from an FE on the HP channel, they will be processed first until there are no more left before processing the next priority work which is to transmit any outstanding HP channel messages going to the FE.

A.3. SCTP TML Channel Termination

TOC

Appendix B.2 (TML Shutdown) describes a controlled disassociation of the FE from the NE.
It is also possible for connectivity to be lost between the FE and CE on one or more sockets. In cases where SCTP multi-homing features are used for path availability, the disconnection of a socket will only occur if all paths are unreachable; otherwise, SCTP will ensure reachability. In the situation of a total connectivity loss of even one SCTP socket, it is recommended that the FE and CE SHOULD assume a state equivalent to ForCES Association Teardown being issued and follow the sequence described in Appendix B.2 (TML Shutdown).
A CE could also disconnect sockets to an FE to indicate an "emergency teardown". The "emergency teardown" may be necessary in cases when a CE needs to disconnect an FE but knows that an FE is busy processing a lot of outstanding commands (some of which the FE hasn't got around to processing yet). By virtue of the CE closing the connections, the FE will immediately be asynchronously notified and will not have to process any outstanding commands from the CE.

A.4. SCTP TML NE level channel scheduling

TOC

In handling NE-level I/O work, an implementation needs to worry about being both fair and robust across peer ForCES nodes.
Fairness is desired so that each peer node makes progress across the NE. For the sake of illustration consider two FEs connected to a CE; whereas one FE has a few HP messages that need to be processed by the CE, another may have infinite HP messages. The scheduling scheme may decide to use a quota scheduling system to ensure that the second FE does not hog the CE cycles.
Robustness is desired so that the NE does not succumb to a DoS attack from hostile entities and always achieves a maximum stable workload processing level. For the sake of illustration consider again two FEs connected to a CE. Consider FE1 as having a large number of HP and MP messages and FE2 having a large number of MP and LP messages. The scheduling scheme needs to ensure that while FE1 always gets its messages processed, at some point we allow FE2 messages to be processed. A promotion and preemption based scheduling could be used by the CE to resolve this issue.

TOC

Appendix B. Service Interface

This section provides high level service interface between FEM/CEM and TML, the PL and TML, and between local and remote TMLs. The intent of this interface discussion is to provide general guidelines. The implementer is expected to worry about details and even follow a different approach if needed.
The theory of operation for the PL-TML service is as follows:

1. The PL starts up and bootstraps the TML. The end result of a successful TML bootstrap is that the CE TML and the FE TML connect to each other at the transport level.

2. Sending and reception of the PL level messages commences after a successful TML bootstrap. The PL uses send and receive PL-TML interfaces to communicate to its peers. The TML is agnostic to the nature of the messages being sent or received. The first message exchanges that happen are to establish ForCES association. Subsequent messages maybe either unsolicited events from the FE PL, control message redirects from/to the CE to/from FE, and configuration from the CE to the FE and their responses flowing from the FE to the CE.

3. The PL does a shutdown of the TML after terminating ForCES association.

B.1. TML Boot-strapping

TOC

Figure 6 (SCTP TML Bootstrapping) illustrates a flow for the TML bootstrapped by the PL.
When the PL starts up (possibly after some internal initialization), it boots up the TML. The TML first interacts with the FEM/CEM and acquires the necessary TML parameterization (Section 4.2.1.6 (SCTP TML Parameterization)). Next the TML uses the information it retrieved from the FEM/CEM interface to initialize itself.
The TML on the FE proceeds to connect the 3 channels to the CE. The socket interface is used for each of the channels. The TML continues to re-try the connections to the CE until all 3 channels are connected. It is advisable that the number of connection retry attempts and the time between each retry is also configurable via the FEM. On failure to connect one or more channels, and after the configured number of retry thresholds is exceeded, the TML will return an appropriate failure indicator to the PL. On success (as shown in Figure 6 (SCTP TML Bootstrapping)), a success indication is presented to the TML.

```
     FE PL      FE TML           FEM  CEM        CE TML              CE PL
       |            |             |    |            |                    |
       |            |             |    |            |      Bootup        |
       |            |             |    |            |<-------------------|
       |  Bootup    |             |    |            |                    |
       |----------->|             |    |get CEM info|                    |
       |            |get FEM info |    |<-----------|                    |
       |            |------------>|    ~            ~                    |
       |            ~             ~    |----------->|                    |
       |            |<------------|                 |                    |
       |            |                               |-initialize TML     |
       |            |                               |-create the 3 chans.|
       |            |                               | to listen to FEs   |
       |            |                               |                    |
       |            |-initialize TML                |Bootup success      |
       |            |-create the 3 chans. locally   |------------------->|
       |            |-connect 3 chans. remotely     |                    |
       |            |------------------------------>|                    |
       |            ~                               ~ - FE TML connected ~
       |            ~                               ~ - FE TML info init ~
       |            | channels connected            |                    |
       |            |<------------------------------|                    |
       | Bootup     |                               |                    |
       | succeeded  |                               |                    |
       |<-----------|                               |                    |
       |            |                               |                    |
```

Figure 6: SCTP TML Bootstrapping

On the CE things are slightly different. After initializing from the CEM, the TML on the CE side proceeds to initialize the 3 channels to listen to remote connections from the FEs. The success or failure indication is passed on to the CE PL level (in the same manner as was done in the FE).
Post boot-up, the CE TML waits for connections from the FEs. Upon a successful connection by an FE, the CE TML level keeps track of the transport level details of the FE. Note, at this stage only transport level connection has been established; ForCES level association follows using send/receive PL-TML interfaces (refer to Appendix B.3 (TML Sending and Receiving) and Figure 8 (Send and Recv Flow)).

TOC

B.2. TML Shutdown

Figure 7 (FE Shutting down) shows an example of an FE shutting down the TML. It is assumed at this point that the ForCES Association Teardown has been issued by the CE.
When the FE PL issues a shutdown to its TML for a specific PL ID, the TML releases all the channel connections to the CE. This is achieved by closing the sockets used to communicate to the CE.

```
     FE PL      FE TML                      CE TML              CE PL
       |            |                         |                    |
       |  Shutdown  |                         |                    |
       |----------->|                         |                    |
       |            |-disconnect 3 chans.     |                    |
       |            |------------------------>|                    |
       |            |                         |                    |
       |            |                         |-FE TML info cleanup|
       |            |                         |-optionally tell PL |
       |            |                         |------------------->|
       |            |- clean up any state of  |                    |
       |            | channels disconnected   |                    |
       |            |                         |                    |
       |            |<------------------------|                    |
       | Shutdown   |                         |                    |
       | succeeded  |                         |                    |
       |<-----------|                         |                    |
       |            |                         |                    |
```

Figure 7: FE Shutting down

On the CE side, a TML level disconnection would result in possible cleanup of the FE state. Optionally, depending on the implementation, there may be need to inform the PL about the TML disconnection.

B.3. TML Sending and Receiving

TOC

The TML is agnostic to the nature of the PL message it delivers to the remote TML (which subsequently delivers the message to its PL).
Figure 8 (Send and Recv Flow) shows an example of a message exchange originated at the FE and sent to the CE (such as a ForCES association

message) which illustrates all the necessary service interfaces for sending and receiving.
When the FE PL sends a message to the TML, the TML is expected to pick one of HP/MP/LP channels and send out the ForCES message.

```
     FE PL       FE TML           CE TML                CE PL
        |            |              |                      |
        |PL send     |              |                      |
        |----------->|              |                      |
        |            |              |                      |
        |            |-Format msg.  |                      |
        |            |-pick channel |                      |
        |            |-TML  Send    |                      |
        |            |------------->|                      |
        |            |              |-TML Receive on chan. |
        |            |              |-decapsulate          |
        |            |              |- mux to PL/PL recv   |
        |            |              |--------------------->|
        |            |              |                      ~
        |            |              |                      ~ PL Process
        |            |              |                      ~
        |            |              |  PL send             |
        |            |              |<---------------------|
        |            |              |-Format msg. for send |
        |            |              |-pick chan to send on |
        |            |              |-TML send             |
        |            |<-------------|                      |
        |            |-TML Receive  |                      |
        |            |-decapsulate  |                      |
        |            |-mux to PL    |                      |
        | PL Recv    |              |                      |
        |<---------- |              |                      |
        |            |              |                      |
```

Figure 8: Send and Recv Flow

When the CE TML receives the ForCES message on the channel it was sent on, it demultiplexes the message to the CE PL.
The CE PL, after some processing (in this example dealing with the FE's association), sends to the TML the response. And as in the case of FE PL, the CE TML picks the channel to send on before sending.
The processing of the ForCES message upon arriving at the FE TML and delivery to the FE PL is similar to the CE side equivalent as shown above in Appendix B.3 (TML Sending and Receiving).

Authors' Addresses

TOC

Jamal Hadi Salim
Mojatatu Networks
Ottawa, Ontario
Canada
Email: hadi@mojatatu.com

Kentaro Ogawa
NTT Corporation
3-9-11 Midori-cho
Musashino-shi, Tokyo 180-8585
Japan
Email: ogawa.kentaro@lab.ntt.co.jp