CCAMP Working Group
Internet-Draft
Intended status: Informational
Expires: March 29, 2014

D. Ceccarelli
Ericsson
O. Gonzalez de Dios
Telefonica I+D
F. Zhang
X. Zhang
Huawei Technologies
September 25, 2013

Use cases for operating networks in the overlay model context
draft-ceccadedios-ccamp-overlay-use-cases-02

Abstract

This document defines a set of use cases for operating networks in the overlay model context through the Generalized Multiprotocol Label Switching (GMPLS) overlay interfaces.

Status of this Memo

Copyright Notice

Table of Contents

1. Introduction

The GMPLS overlay model [RFC 4208] specifies a client-server relationship between networks where client and server layers are managed as separate domains because of trustiness, scalability and operational issue. By means of procedures from the GMPLS protocol suite it is possible to build a topology in the client (overlay) network from Traffic Engineering paths in the server network. In this context, the UNI (User to Network Interface) is the demarcation point between networks. It is a boundary where policies, administrative and confidentiality issues apply that limit the exchange of information.

This GMPLS overlay model supports a wide variety of network scenarios. The packet over optical scenario is probably the most popular example where the overlay model applies.

In order to exploit the full potential of client/server network interworking in the overlay model, it may be desirable to know in advance whether is it feasible or not to connect two client network nodes [INTERCON-TE]. This requires to have a certain amount of TE information of the server network in the client network. This need not be the full set of TE information available within each network, but does need to express the potential of providing TE connectivity. This subset of TE information is called TE reachability information.

The goal of this document is to define a set of solution independent use cases applicable to the overlay model. In particular it focuses on the network scenarios where the overlay model applies and analyzes the most interesting aspects of provisioning, recovery and path computation.

2. Terminology

The following terms are used within the document:

- Edge node [RFC4208]: node of the client domain belonging to the overlay network, i.e. nodes with at least one interface connected to the server domain.

- Core node [RFC4208]: node of the server domain.

- Access link: link between core node and edge node. It is the link where the UNI is usually implemented.

- Remote node: node in the client domain which has no direct access to the server domain but can reach it through an edge node

in its same administrative domain.

- Local trigger: LSP setup request issued to an edge node. It triggers the setup of a client layer FA through the server domain via a UNI interface.

- Remote trigger: LSP setup request issued to a remote node. It triggers the setup of a client layer LSP which, upon reaching an edge node, will use connectivity in the server domain dynamically provided via an UNI interface.

3. Background and Assumptions

All the use cases listed in the sections below can be applied to any combination of, unless otherwise specified:

* Local trigger or remote signaling

* Grey interface or colored interface

With local trigger we mean the case in which a trigger for the provisioing of a service over the overlay interface is issued to one of the edge nodes belonging to the overlay network, i.e. directly connected to the UNI.

```
               1.Trigger
                  |            2. Setup
                  V   -------------------->
   +--+   +--+   +--+     /-\       /-\     /-\      +--+   +--+   +--+
   |R1|---|R2|---|R3|****( A )-----( B )---( C )*****|R5|---|R6|---|R7|
   +--+   +--+   +--+     \-/       \-/\    \-/      +--+   +--+   +--+
     \            /        | \     / |  \    |         \           /
      \          /         |  \   /  |   \   |          \         /
       \        /          |   \ /   |    \  |           \       /
        \ +--+ /           |    \    |     \ |            \ +--+/
          |R4|             |   / \   |      \|              |R8|
          +--+            /-\ /    \/-\     /-\             +--+
     3.Advertisement     ( D )-----( E )---( F )      3.Advertisement
                          \-/       \-/     \-/
        *** = overlay interfaces
```

Figure 1: Local trigger

As it is possible to see in the figure above, a trigger is issued on R3 (edge node) for starting the setup request procedure over the

overlay interface (R3-A). Once the LSP in the server domain is setup and an adjacency in the packet layer between R3 and R5 is created, it can be advertised in the rest of the client domain and used by the signaling protocol (e.g. LDP) for setting up end-to-end (e.g. from R1 to R7) client layer LSPs.

On the other hand, the remote signaling consists on the utilization of a connection oriented signaling protocol in the client domain that allows issuing the end to end service setup trigger directly on the end nodes of the client domain. The signaling message, upon reaching the edge node (R3), will trigger the setup of the service in the server layer via the overlay interface.

```
 1.Trigger
 | 2. Signaling  3. Trigger
 V -------------->                                  |------------>|
                 |------>----------------->------->|
                 |<-----<-----------------<--------|
<----------------|---------------------------------|-------------|
+--+   +--+   +--+     /-\       /-\     /-\      +--+   +--+   +--+
|R1|---|R2|---|R3|****( A )-----( B )---( C )*****|R5|---|R6|---|R7|
+--+   +--+   +--+     \-/       \-/\    \-/      +--+   +--+   +--+
  \            /        | \     / |  \    |         \           /
   \          /         |  \   /  |   \   |          \         /
    \        /          |   \ /   |    \  |           \       /
     \ +--+ /           |    \    |     \ |            \ +--+/
       |R4|             |   / \   |      \|              |R8|
       +--+            /-\ /    \/-\     /-\             +--+
                      ( D )-----( E )---( F )
                       \-/       \-/     \-/
```

Figure 2: Remote Signaling

The utilization of the remote trigger allows for a strict control of the resources that will be used for the setup of the end to end service. In order to have a correct setup of the end to end service the trigger issued to R1 must include the overlay nodes to be used for the setup of the service in the server layer (R3 and R5). The network operator is supposed to know that the edge nodes to be used are R3 and R5.

When operating an IP over WDM network in the overlay context, a further distinction between grey and colored interfaces needs to be taken into account. In other words in the former case the transponder is hosted on the core border nodes, while in the latter in the edge node. The physical impairments to be considered are

different in the two cases (for further details please see Section 4.11) but the behaviour of the interface does not change and all use cases depicted below can be applied both to the grey and colored interfaces.

4. Use Cases

4.1. UC 1 - Provisioning

Requirement: The network operator must be allowed to setup an unprotected end to end service between two client layer nodes.

This use case simply consists on providing an operator with the capability of setting up a service in the client layer either by means or local trigger or remote signaling. The operator does not put any constraint over the path computation in the server layer.

4.2. UC 2 - Provisioning with optimization

Requirement: The network operator must be allowed to setup a service expressing which parameter must be optimized when computing the path.

This use case applies both to the local trigger and the remote signaling scenarios. In both cases the path computation element in the server layer (being it centralized or distributed) is demanded to provide a path between R3 and R5 which minimizes a given parameter (e.g. delay, jitter, TE metric).

```
               1.Trigger(param min)
                  |  2. Setup(param min)   3.Path computation(param min)
                  V  ------>
   +--+   +--+   +--+     /-\       /-\     /-\      +--+   +--+   +--+
   |R1|---|R2|---|R3|****( A )-----( B )---( C )*****|R5|---|R6|---|R7|
   +--+   +--+   +--+     \-/       \-/\    \-/      +--+   +--+   +--+
     \            /        | \     / |  \    |         \           /
      \          /         |  \   /  |   \   |          \         /
       \        /          |   \ /   |    \  |           \       /
        \ +--+ /           |    \    |     \ |            \ +--+/
          |R4|             |   / \   |      \|              |R8|
          +--+            /-\ /    \/-\     /-\             +--+
                         ( D )-----( E )---( F )
                          \-/       \-/     \-/
        *** = overlay interfaces
```

Figure 3: Provisioning with optimization

In the figure above the case of local trigger with specified parameter to be minimized is depicted, but same considerations apply to the remoe signaling (trigger on R1). In that case the parameter to be minized needs to be conveyed from R1 to R3 so that the setup request over the overlay interface can be issued taking into account the OF.

4.3. UC 3 - Provisioing with constraints

Requirement: The network operator must be allowed to setup a service imposing upper bounds for a set of parameters during the path computation.

This use cases is extremely similar to the Provisioing with Optimization one. This time, instead of/in addittion to giving the possibility of specifying which parameter needs to be optimized during the path computation, the network operator is also able to indicate and upper bound for a set of parameters which is not being minimized in the path computation.

```
            1.Trigger(constraint)
               |  2.Setup(const) 3.Path computation(const)
               V  ------>
+--+   +--+   +--+     /-\       /-\     /-\      +--+   +--+   +--+
|R1|---|R2|---|R3|****( A )-----( B )---( C )*****|R5|---|R6|---|R7|
+--+   +--+   +--+     \-/       \-/\    \-/      +--+   +--+   +--+
  \            /        | \     / |  \    |         \           /
   \          /         |  \   /  |   \   |          \         /
    \        /          |   \ /   |    \  |           \       /
     \ +--+ /           |    \    |     \ |            \ +--+/
       |R4|             |   / \   |      \|              |R8|
       +--+            /-\ /   \/-\     /-\              +--+
                      ( D )-----( E )---( F )
                       \-/       \-/     \-/
     *** = overlay interfaces
```

Figure 4: Provisioning with constraints

It is possible for example to ask for a path between R3 and R5 which, in addition to minimizing a given OF, does not introduce a delay higher than 10ms or where the jitter is not more than 3ms.

As per the optimization use case, when remote signaling is used (trigger on R1) a mean to convey the path computation constraints till the edge node (R3) is needed.

4.4. UC 4 - Provisioing with diversity

Requirement: The network operator must be allowed to setup a services in the server layer in diversity with respect to server layer resources or not sharing the same fate with other server layer services.

This scenario is extremely common in those cases where different services in the server domain are used to provision protected services in the client layer. The services in the server layer can be computed/provisioned sequentially or in parallel but in both cases the requirement is to have them totally disjoint, so that a single failure in the server layer does not impact two or more services in the client layer which are supposed to be in a protection relationship between each other (e.g. 1+1 protection).

```
+--+   +--+   +--+     /-\       /-\     /-\      +--+   +--+   +--+
|R1|---|R2|---|R3|****( A )*****( B )***( C )*****|R5|---|R6|---|R7|
+--+   +--+   +--+     \-/       \-/\    \-/      +--+   +--+   +--+
                #       | \     / |  \    |      #
                 #      |  \   /  |   \   |     #
                  #     |   \ /   |    \  |    #
                   #    |    \    |     \ |   #
                    #   |   / \   |      \|  #     ***=Service A
                     # /-\ /    \/-\     /-\#      ###=Service B
                      ( D )#####( E )###( F )
                       \-/       \-/     \-/
```

Figure 5: Provisioing with diversity

In a scenario like the one depicted above, it is possible to use Service A and Service B for the setup of a protected service in the client domain as a fault in the server domain would not impact both of them. In the case of parallel request, R3 asks the path computation in the server domain to provide two totally disjoint paths. On the other side, when sequential requests are issued, and identifiers for Service A (or a set of identifiers indicating its resources) is needed so that the request for the setup of Service B can be issued with the constraint of avoiding the resources related to such identifier.

Another case of provisioning with diversity is the one where the operator in the client domains wants the server domain PCE to exclude some resources from the path computation because of e.g. trustness reasons. In such a case, supposing that such resources are known to the operator, it must be possible to indicate them as path

computation constraint in the service setup request.

4.5. UC 5 - Concurrent provisioing

Requirement: The network operator must be allowed to setup a plurality of services not necessarily between the same pair of edge nodes.

Here is another case particularly interesting from a protection point of view. In the case above the same edge node was asking for different services in the server layer, but in order to have end to end diversity (i.e. from R1 to R8 in figure below), there is the need to be able to provide disjoint services between different pairs of edge nodes.

```
+--+   +--+   +--+     /-\       /-\     /-\      +--+   +--+   +--+
|R1|---|R2|---|R3|****( A )*****( B )***( C )*****|R6|---|R7|---|R8|
+--+   +--+   +--+     \-/       \-/\    \-/      +--+   +--+   +--+
    \                   | \     / |  \    |                     /
     \                  |  \   /  |   \   |                    /
      \                 |   \ /   |    \  |                   /
       \                |    \    |     \ |                  /
        \               |   / \   |      \|                 /
       +--+   +--+     /-\ /    \/-\     /-\      +--+   +---+
       |R4|---|R5|####( D )#####( E )###( F )#####|R9|---|R10|
       +--+   +--+     \-/       \-/     \-/      +--+   +---+

   ***=Service A
   ###=Service B
```

Figure 6: Concurrent provisioning

In this example Service A is provided between R3 and R6 and Service B between R5 and R9. Some sort of coordination is needed between R3 and R5 (directly between them or via R1) so that the requests to the server layer can be conveniently issued.

4.6. UC 6 - Reoptimization

Requirement: The network operator must be allowed to setup a plurality of services so that the overall cost of the network is minimized and not the cost of a single service.

TBD

4.7. UC 7 - Query

Requirement: The server network must be able to tell the network operator the actual parameters characterizing an existing service.

The capability of retrieving from the server domain some parameters qualifying a service can be estremely useful in different cases. One of them is the case o sequential provisioning with diversity requirements. In the case the operator wants to set-up a service in diversity from an existing one, hence it must be possible for the server domain to export some parameters univocally identifying the resources (e.g. SRLGs).

Editor note: collection of other TE metrics (e.g. latency, jitter) to be considered?

4.8. UC 8 - Availability check

Requirement: The network operator must be allowed to check if in the server layer there are enough resources to setup a service with given parameters.

TBD

4.9. UC 9 - P2MP services

Requirement: The network operator must be allowed to setup a P2MP service with given parameters.

TBD

4.10. UC 10 - Privacy

Requirement: The network operator must be allowed to provision different groups of users with independent addressing spaces.

This is a particularly useful functionality for those cases where the resources of the service provider are leased and shared among several other service providers or customers.

4.11. UC 11 - Colored overlay

Requirement: The network operator must be allowed to provision a service in the server layer through a colored overlay interface.

This use case applies to networks where the server domain is a WDM network. In those cases it is possible to either have a grey interface between client and server domains (i.e. transponder on the

border core node) or a colored interface between them (i.e. transponder on the edge node).

All the previous use cases assume the case of grey interface, but there are particular network scenarios in which it is possible to move the transponders from the core to the edge nodes and hence save on expensive pieces of hardware.

The issue with this solution is that the PCE in the server layer, being either centralized or distributed, has only visibility of what is inside the server domain and hence has not all the info needed to perform the validation of a path. The edge node must provide the PCE in the server domain with a set of info needed for a correct path computation and path validation from transponder to transponder (i.e. between edge nodes) all along the server domain.

The type of information needed for this scenario can be classified into three categories:

- Feasibility: Parameters like the output power of the transponder are needed in order to state e.g. the amount of km that can be reached without regeneration.

- Compatibility: The egress transponder must be compatible with the ingress one. Parameters that influence the level of compatibility can be for example the type of FEC (Forward Error Correction) used or the modulation format (which also impacts the feasibility together with the bit rate).

- Availability: Transponders can be tunable within a range of lambdas or even locked to a single lambda. This impacts the path computation as not every path in the network might have such lambda(s) supported or available at the time the path computation is performed.

In figure below it is possible to see that the PCE is aware of all the info between A and C (i.e. within the server domain scope) but what is missing is info related to the transponders on R1 and on R2 and of the access links. (i.e. R1-A and C-R2).

```
        -Feasibility
        -Compatibility |=====|
        -Availability  | PCE |
           /---------->|=====|
          /
         /
   +--+ /      /-\       /-\     /-\         +--+
   |R1|*******( A )-----( B )---( C )********|R2|
   +--+        \-/       \-/\    \-/         +--+
                | \     / |  \    |
                |  \   /  |   \   |
                |   \ /   |    \  |
                |    \    |     \ |
                |   / \   |      \|
               /-\ /    \/-\     /-\
              ( D )-----( E )---( F )
               \-/       \-/     \-/
        *** = colored overlay interfaces
```

Figure 7: PCE feeding for colored UNI

There is not yet a standard set of parameters that is needed for path computation in WDM networks but an example of some of them is provided in the following list:

- o Modulation format

- o FEC (type or gain)

- o Minimum transponder output power

- o Bitrate

- o Dispersion tolerance

- o OSNR (minimum required)

5. Security Considerations

TBD

6. IANA Considerations

TBD

7. Contributors

TBD

8. References

8.1. Normative References

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

8.2. Informative References

Authors' Addresses

Daniele Ceccarelli
Ericsson
Via E. Melen 77
Genova - Erzelli
Italy

Email: daniele.ceccarelli@ericsson.com

Oscar Gonzalez de Dios
Telefonica I+D
Don Ramon de la Cruz 82-84
Madrid 28045
Spain

Email: ogondio@tid.es

Fatai Zhang
Huawei Technologies
F3-5-B R&D Center, Huawei Base
Shenzhen 518129 P.R.China Bantian, Longgang District
Phone: +86-755-28972912

Email: zhangfatai@huawei.com

Xian Zhang
Huawei Technologies
F3-5-B R&D Center, Huawei Base
Shenzhen 518129 P.R.China Bantian, Longgang District
Phone: +86-755-28972913

Email: zhang.xian@huawei.com