Establishment of Point-to-Multi-Point path in GMPLS controlled Wide Area Ethernet

Ko Kikuta, Daisuke Ishii, Satoru Okamoto and Naoaki Yamanaka
KEIO University, JAPAN
kikuta@yamanaka.ics.keio.ac.jp

www.mpls2009.com

Outline

- Motivation for P2MP
- Fundamental P2MP Architecture
- Extensions for P2MP
- Implementation
  - Node Behavior
  - RESV message combining
  - Testbed network
  - Captured packets
- Further study
- Conclusion
- Acknowledgement
Motivation for P2MP

- Multicast Applications are well developed
  - Various applications, such as TV distribution and Video conference, need a technology to send packets/frames to different destinations simultaneously
- Connection Oriented (CO) Path provisioning became popular by MPLS and GMPLS
- RFC 4875, “Extensions to RSVP-TE for Point-to-Multipoint TE LSPs” has been published

P2MP for Wide Area Ethernet

- Wide-Area-Ethernet...
  - A network providing Layer-2 VPN services
  - CO path is established using a VLAN technology
  - High-speed and cost effective communication
  - GMPLS is applicable

- P2MP for Wide-Area-Ethernet
  - Multi-points connectivity is necessary for Layer-2 VPN
  - Ethernet originally supports multicasting. Easy to replicate the data traffic
  - Lack P2MP path provisioning mechanism
This presentation

- The implementation of P2MP path provisioning based on RFC4875 will be presented
  - We extended a RSVP-TE program and established P2MP path on an experimental network

Fundamental P2MP architecture

- A P2MP path is established among a Sender (INGRESS) and multiple Receivers (EGRESSes).

- P2MP LSP is comprised of a set of Sub-LSPs which is established from INGRESS to each EGRESS
Extension for P2MP (1)

- A new SESSION object is defined for P2MP

- SESSION object for P2P (LSP Tunnel IPv4):
  - Tunnel End Point Address
  - Tunnel ID
  - Extended Tunnel ID

- SESSION object for P2MP (P2MP LSP Tunnel IPv4):
  - P2MP ID
  - Tunnel ID
  - Extended Tunnel ID

- A new S2L (Source to Leaf) SUB LSP object carries each Destination Address.

Extension for P2MP (1) cont.

- SESSION object does not carry any Destination Address

- SESSION object:
  - P2MP ID
  - Tunnel ID
  - Extended Tunnel ID

- SENDER_TEMPLATE object:
  - Tunnel sender address
  - LSP ID

This couple of objects identifies a P2MP LSP

- One or more S2L SUB LSP objects for all destinations.
  At each BRANCH node, they will be distributed into multiple messages.
Extension for P2MP (2)

- SERO (SECONDARY ERO) encodes an explicit route for each Destination, which is compressed for shared hops.

**SESSION object**

- A -> B -> D -> G
- Node G
- Node E
- B -> C -> E
- Node F
- C -> F

**PATH message**

- ERO to G
- SERO to E
- SERO to F

**SESSION object**

- EGRESS
- INGRESS
- TRANSIT / BRANCH

Extension for P2MP (2) cont.

- PATH Message will be replicated at Branch node.

**SESSION object**

- Replicates the original message...
- Inherits the original SESSION object
- Takes away relevant S2L SUB LSP objects and SEROs from original message
A new SENDER TEMPLATE object is defined for P2MP

P2P (LSP Tunnel IPv4)  P2MP (P2MP LSP Tunnel IPv4)

SENDER TEMPLATE obj.
- Tunnel Sender Address
- LSP ID

SENDER TEMPLATE obj.
- Tunnel Sender Address
- Tunnel ID
- Sub-Group Originator ID
- Sub-Group ID

Sub-Group Originator ID and Sub-Group ID will be changed at BRANCH nodes when replicated messages are generated.

Sub-Group Originator ID must be changed to distinguish from the original message.
Implementation environment

- Software: “GMPLS Engine” (C++)
  
  - Developed by NTT-AT
  - We extended the source codes of RSVP-TE

- Implementation
  
  - RSVP-TE (extended)
  - OSPF-TE (no extension in this time)
  - LMP was not used

- Data-Plane: NETGEAR Layer-2 Switch

Implemented Node Behavior: INGRESS node

1. Receive a **new Path request** from the Sender

2. Get the explicit route for each Egress node. Each route should **share the hops** (ex. A - C)

3. Make up a PATH message with a new **SESSION**, **SENDER_TEMPLATE** and **S2L_SUB_LSP** object, also **SEROs** are added with the explicit routes

4. Send the PATH Message to downstream node
Implemented Node Behavior: TRANSIT, BRANCH, EGRESS

When receiving a PATH Message from Upstream

- **first hop** in each SERO is my address?  
  - YES → Branch Processing
  - NO → **Destination Address** is my Address?  
    - YES → Egress Processing (same as P2P)
    - NO → **Transit Processing** (same as P2P)

Message Replication:
- update the Sub-Group Originator ID
- take away S2L SUB LSPs and SEROs from Original message

Replicated message

Original message

included in the first S2L SUB LSP object

Note that the Branch node may be also Egress node, and Branching may occur recursively.

---

RESV message combining

- To reduce a number of RESV messages, BRANCH node should **combine** the messages

- **Merit**
  - The **number of RESV message processing** is reduced at all upstream nodes

- **Demerit**
  - The BRANCH node **has to wait** all the messages received from all downstream nodes. It increases **extra setup latency**.
  - Total message quantity is only reduced a little
Testbed network

C-Plane

connected to neighbors with GRE tunneling

D-Plane

Linux PC (Fedora Release 8)
Kernel : 2.6
CPU : Intel Core 2 Duo 2.10 GHz
Memory : 2GByte

NETGEAR
Gigabit Layer-2 Managed Switch
(GSM 7212)

Captured packets (and modified Wireshark)

☐ will be shown in the presentation..
Further study

- Extension for **MP2MP**
  - Because of the characteristic of the Ethernet switch, P2MP path can be used as MP2MP path (tree)...
    - Users will use this P2MP path as a MP2MP network

- To establish MP2MP tree
  - How to route the tree? (without loop)
  - How to present a directional bandwidth?
    - Asymmetric Bidirectional LSP (RFC5467) is insufficient.
  - Who control the tree?
    - Initiator-less-setup/teardown will be required.

Conclusion

- For Wide-Area-Ethernet, P2MP path provisioning mechanism is necessary

- Based on RFC 4875, prototype RSVP-TE is extended for supporting P2MP

- Implemented P2MP path provisioning works well on the experimental network
Acknowledgement

☐ This work is partially supported by the following;

☐ “Lambda Access” Project funded by the National Institute of Information and Communication Technology (NICT)

☐ Global COE Program
  “High-Level Global Cooperation for Leading Edge Platform on Access Spaces (C12)”

Do you want to recruit me?

Contact here:

kikuta@yamanaka.ics.keio.ac.jp