802.1aq: link-state handshake for loop prevention

János Farkas
Outline

- Introduction
- Link-state handshake mechanism
- Implementation possibilities in SPPBB
- Examples
- Summary
Introduction

- There are no loops in stable topologies
- Loops may appear during topology transients
- Inconsistent view on network topology at different nodes may cause transient loops in case of a link-state control protocol
Link-state handshake mechanism

- Let’s make it sure that bridges having different view on network topology do not exchange frames
- The link between adjacent neighbors has to be blocked after a topology change until they agree that both of them received the latest advertisement(s) on the change(s)
- The agreement between neighbors can be implemented in a handshake mechanism
- Agreements at different part of the network are independent of each other
Handshake operation

- Link-State Advertisement on a topology change received?
  - Yes
    - Block ports
    - Send out “Synchronization Request” messages on the ports
    - “Sync Ack” received?
      - No
        - “Sync Req” on same Link-State Advertisement received?
          - Yes
            - Unblock the port on which “Sync Ack” received
          - No
            - Are there still blocked ports?
              - Yes
                - Unblock the Port if it is blocked
              - No
                - “Sync Req” received?
                  - No
                    - Same Link-State Advertisement received?
                      - Yes
                        - Send “Sync Ack”
                      - No
                        - “Sync Req” on same Link-State Advertisement sent?
                          - Yes
                            - Send “Sync Ack”
                          - No
                            - Unblock the Port if it is blocked
Simple example

- Solves the problem identified in aq-farkas-loop-prevention-1107 and analyzed further in aq-fedyk-loop-prevention-0108
Implementation possibilities in SPPBB

**MSTP BPDU**
- LSP ID and Sequence number has to be embedded
- Proposal = Synchronization Request
- Agreement = Synchronization Acknowledgement

**IS-IS PDU**
- LSP = Synchronization Request
- Partial Sequence Number PDU (PSNP) = Synchronization Acknowledgement
- PSNP Interval determines the convergence time: it has to be in the order of milliseconds
- New flag per port is needed to control link blocking
Generic example: B and F are notified first

- Note that the order of nodes becoming aware of the change is considered as a random order
Generic example: B and F request synchronization

Informed about topology change(s)

Has outdated view on the topology

Blocking

Blocked Link

Synchronization Request

Synchronization Acknowledgement
Generic example: A and H realized the topology change

Informed about topology change(s)
Has outdated view on the topology
Blocking
Blocked Link
Synchronization Request
Synchronization Acknowledgement
Generic example:
C, D, E, I and J are notified too
Generic example: G is also aware of the change
Generic example: All nodes are updated
Summary

- Transient loops may appear due to inconsistent topology view in case of a link-state control protocol

- Synchronization can be implemented by a handshake mechanism

- Neighbors have to agree on latest changes before they send frames to each other

- Thus loops are prevented