Shortest path bridging state machines

Clause 13 modifications for SPB/SPBB

Mick Seaman
mick_seaman@ieee.org

Agenda

- Why are we modifying Clause 13
- What are we changing, what stays the same
- Agreements – refresher and SPB additions
- Additional variables and procedures, state machine changes

Why modify Clause 13 ?

If IS-IS is being used can’t we just add the extra loop prevention stuff there ?

Why does IS-IS for bridging have to have anything to do with Spanning Tree BPDUs ?

Why we need to modify Clause 13 (1)

- Have to interoperate with existing bridges, and deploy in islands/network cores
- Network loop prevention requires synchronizing state at island/core boundaries with bridges that only understand RSTP/MSTP BPDUs
- That means:
  - using ISIS-SPB results in BPDUs
  - injecting BPDU information into ISIS-SPB

Why we need to modify Clause 13 (2)

- ISIS-SPB can make some decisions faster, e.g. determine CIST Port Roles and priority vectors
  - Stop existing mechanisms from overriding with temporarily incorrect information
- Ports inside SPT Regions synchronize forwarding state with boundary ports
  - Need to specify how, without reinvention

What to add/change

- Agreements for ISIS-SPB calculated CIST
- Agreements for SPTs (using Agreement Digest)
- ISIS-SPB calculates CIST in Region
- Match SPT state to CST at Region boundary
- Tweak Region boundary calculation

What to keep the same

Most things, including:

- No new state machines
- MSTI operation

Treat SPB and SPBB identically

- Differences in allocating frames to trees, not in calculating trees and forwarding transitions

CIST Agreements (1)

New connectivity created by Designated Ports becoming Forwarding. Allowed iff:

- Neighbour's port is Root Port or Alternate Port
- Agreement received that is worse than any own Root Port has outstanding

- Agreement from Root/Alternate Port promises:
  - Each of my Forwarding Designated Ports has received an Agreement no better than this
- Designated Port discards old Agreements

CIST Agreements (2)

For RSTP/MSTP CIST every message from a Designated Port is an implicit discard:

- Distance vector propagation ensures parents have discarded old unusable Agreements

Link state neighbours can compute in any order:

- Check Root Port's outstanding Agreements before making Designated Port Forwarding
- Block ports to ensure no connectivity to parents holding old Agreements

MSTI & SPT Agreements

Only one CST Root Port for Region (Master Port)

- Agreement sent by MSTI Designated Ports
- When all other ports (recursively) agree Master
- Master agreement differs from implicit discard, propagates up tree branches before down

Explicit discard provides same capability for SPTs

- Root Port possibly connected to old Master only when old Agreement outstanding

Agreements for ISIS-SPB

Use existing Proposal/Agreement variables: sync all ports after link state update, port is synced if Discarding or agreed, agree when Discarding or all other ports synced

- Designated Port:
  - agreed set when a received Agreement complements designatedPriority, and no Agreements are outstanding
  - When agree for CIST, discard unusable Agreements
  - When agree for all SPTs, transmit new TAP Digest
- Root/Alternate Port:
  - agreed set when all outstanding Agreements (if any) complement designatedPriority
  - When agree for CIST transmit Agreement, and discard received Agreements
  - When agree for all SPTs transmit new TAP Digest

New variables

No new timers (or extra uses of existing timers)

Per bridge variables: agreementDigest

Per port variables: agreedN, agreedND, agreeN, agreeND, agreePending, agreedDigest, agreeDigest

Per tree variables: agreementOutstanding, neighbourPriority (SPT only)

New or changed procedures

- txRstp() – can now transmit SPT BPDUs, updates agreeN, agreedND, agreePending before transmission.
- updtRolesTree() – ensure that ISIS-SPB makes the decisions on IST and SPT roles and priorities
- rcvdMsgs() – use rcvAgreements() if rcvd BPDU is internal to SPT Region
- rcvAgreements() – extract Agreement Digest and agreement number from received BPDUs.
- updtAgreement() – update the Agreement variables after link state computation or BPDU receipt

State machine changes

No new state machines. No changes to PTI, PRS[1], PPM, BDM, PIM[2], PRS[3], PST, TCM, L2GP[4]

No new states in other machines apart from PTX (1).

- PTX – Updates agreeDigest and prompts transmission when all SPTs agree for the port.
- PRT:DESIGNATED_AGREED – Rename (_AGREE), also execute when CIST SPB Discarding.
- PRT:ROOT_DISCARD – Use state added for dispute detection
- Initialization of new variables (not yet done).

Conclusion

The hard part was figuring it out, the changes are minor