An Autonomic Control Plane
draft-ietf-anima-autonomic-control-plane-04

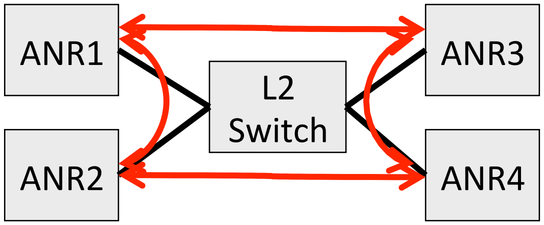
97th IETF, Nov 2016

Michael Behringer (editor), Toerless Eckert, S. Bjarnason

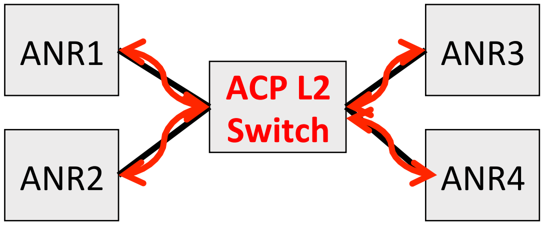
Changes since -03 (IETF96 version)

- Little progress (author issues) – will pick up pace after IETF97
- Reworked text about discovery protocol
    - -03 assumed/suggested mDNS for insecure ACP neighbor discovery because BRSKI design team prefers it, and common protocol for BRSKI/ACP sounded prudent
        - GRASP for A) (TLS) secured ACP channel negotiation, B) inside ACP
    - -04 adds back GRASP for insecure ACP neighbor discovery
        - Leveraging improved/detailed text/mode definitions of GRASP from its -08 draft.
            - Eg: DULL: mode of GRASP for insecure L2 discovery
        - Revisited technical justifications for preference of GRASP in main part
        - Removed explanations from appendix.

Key argument for GRASP/DULL for ACP: L2/LANs



"Full mesh ACP channels"



"physcial link ACP channels"

- Large LANs without ACP capable L2 switch:
  - Unpredictable scale requirement for ACP
  - (potentially many) more neighbors than interfaces – secure channel crypto associations, virtual interfaces..
    - Makes ACP support across arbitrary platforms harder
- ACP enabled L2 switch highly desirable
  - Makes ACP in routers require scale based on #interfaces
  - Makes L2 switches manageable via ACP

Key argument for GRASP/DULL for ACP: L2/LANs

- How can ACP capable L2 switch help to avoid full mesh:
- “Use CDP/LLDP” – packets filtered by switch on L2 ports
  - Packets also filtered by non-ACP L2 switch – eg: does not work
    - Would have to use LLDP over new / unique MAC address as done in Ethernet OAM, but its questionable if this would result in less “novel, unproven code basis” than DULL ?!
- “Use mDNS”
  - If L2 switch supports ACP, filter mDNS packets
  - Not aware of mDNS implementations doing this across L2 ports.
    - LANs canhave high number of mDNS packets. Many L2 switches will not SW-process mDNS at all today. Some will L3 send/receive mDNS. SW-replication to multiple L2 ports can be a possible killer for supporting this discovery.
    - Adding SW-L2 replication to mDNS code basis adds more “nove, unproven” code than GRASP/DULL ?!
- Use GRASP/Dull
  - Unique L2 address, not used by other functions on insecure LANs. No conflicts.
  - DULL: Minimalistic protocol/signaling (see DULL definition). KISS = less prone to attacks ?!

Key argument for GRASP/DULL for ACP: L2/LANs

- How about mDNS for BRSKY ?
- L2 arguments for GRASP/DULL for ACP not crucial for BRSKI discovery:
  - BRSKI proxy does not have to be ACP L2 switch. Can equally well be any other BRSKI router on the LAN.
  - AN/ACP capable L2 switch does not need to change mDNS code basis to support BRSKI discovery via mDNS as pledge(client) or proxy. BRSKI is just another DNS-SD service.
- Need rough consensus determination for BRSKY protocol
  - Currently only rough consensus pro-mDNS from bootstrap design team.

Key argument for GRASP/DULL for ACP: L2/LANs

- Security concern (bootstrap team)
  - Shared code basis GRASP inside secure domain and outside
- Counter arguments (toerless) :
  - Prohibit any use of (shared code) mDNS inside ACP ? because it is used outside ?
  - Prohibit any use of SNMP, NTP, TFTP, netconf, SSH, … inside ACP ?
  - In routers, single, cross-VRF instance of many protocols is common practice
    - Yes, generic security issue. If this becomes important enough, separate instances with cross-VRF memory protection can help.
  - DULL is about as simple as an L2 discovery protocol can become (much simpler than CDP, LLDP, mDNS).
    - If any code basis could be duplicated for best security isolation, it is DULL.

ACP Open issues

- Details about ACP separation from “Data Plane”
  - Ask from Brian Carpenter
  - Toerless: “VRF” long term concept in IETF, but not clear of good implementation facing IETF spec example/reference
  - Reviewing “socket” API capabilities and identifying possible gaps would be good prototyping work
    - VRFs in routers currently no?! Built on top of socket APIs
    - Ability to build GRASP/ACP on top of socket/standard-linux APIs would be a plus.
- Work out example GRASP negotiaton of secure channel option
  - To validate GRASP sufficient in its current form to do this.
  - Target: Brian/Toerless