Autoconfiguration of NOC services in ACP networks via GRASP

IETF 102 Montreal 2018

draft-eckert-anima-noc-autoconfig-00

Toerless Eckert tte+ietf@cs.fau.de (Huawei USA)

v1.0
Problem and resolution

• ANI Pledge gets enrolled with BRSKI zero-touch
• Now what?

• We wanted to create more zero-touch “autonomic” nodes
• This draft proposes the smallest possible incremental step
  • For ANI nodes used in conjunction with existing, centrally managed network
  • Minimum standards work to make RFC8368 work better
    • RFC8366 = ANIMA stable connectivity draft: Use ACP for management from NOC
    • RFC8366 is informational, deals with all possible migration scenarios, etc. pp.
    • Does not specify any of the really good, simple automations possible
    • Those automations need standards work

• This draft is extending ANI standard framework to support RFC8366 neworks
So, what do we need?

• Node needs accurate time. Find NTP server, sync time from it.

• Node needs to automatically syslog, so NOC knows what is going on
  • Including “I am new Pledge, enrolled but unconfigured”

• Remote access to node via ACP needs to be possible SECURELY
  • SSH, Netconf (Netconf typically on top of SSH – or TLS)
  • Automatically discover Radius/Diameter server, enable SSH/Netconf servers, allow access to node SH/Netconf VIA ACP ONLY

• Good minimum? Open to expand/change the minimum set of automated service
  • Also want to use this document as showcase for vendors to understand how to easily automate any more services (standards based or not)
Great! How does it work?

• DNS-SD service discovery across ACP via GRASP
  • draft-eckert-anima-grasp-dnssd
  • GRASP allows to flood objective, this draft proposed a standard to announce/discover DNS-SD services

• draft-eckert-anima-noc-autoconfig
  • Defines required services that ACP nodes (supporting this document) must support, and what to do then:
    • Syslog -> log
    • ntp -> sync time
    • Radius/Diameter -> enable SSH/Netconf

```plaintext
[M_FLOOD, 12340815,
 h'fd89b714f3db000020000064000001', 210000,
 ["SRV.syslog", 4, 255,
  { rfcXXXX: {
    &(sender-loop-count:1) => 255,
    &(srv-element:2) => {
      &(msg-type:1) => &(describe: 0),
      &(service:2) => "syslog",
      &(instance:3) => "east-coast-primary",
      &(priority:5) => 0,
      &(weight:6) => 65535,
      &(kvpairs:7) => { "replicate" => 2 },
      &(range:8) => 2,
    }
  }},
],

[O_IPv6_LOCATOR,
 h'fd89b714f3db000020000064000001', TLS12, 514]
]```
How does it compare to “normal” DNS-SD?

• Multi-hop flooding of DNS-SD has no good solution?!
  • Proxy solutions for more constrained network

• Ability to flood allows to measure hop-distance, select nearest server
  • Part of spec draft-eckert-anima-grasp-dnssd

• DNS-SD for GRASP allows kvpairs like unicast/mDNS encoding
  • Added one kvpair “replication” to indicate #servers to connect to simultaneously
  • For redundancy in services: Eg: send syslog messages to 3 best found servers
  • In other services it does not make sense:
    • Connect to always only one Radius server, switch over quickly when the server is unresponsive.
Summary:

• Make ACP/ANI network automatically use NOC services
• NOC services announce themselves via DNS-SD
  • ACP/ANI nodes discovery serves, autoconfigure themselves accordingly

• GRASP/ACP as flooding transport, DNS-SD message encoding.

• NTP ssh/Netconf, syslog most key core services ?
  • If not, happy to modify list of key services

• Should be easy to individually add more and more services specs
Thank You!