ANIMA DNS-SD compatible services auto configuration

draft-eckert-anima-grasp-dnsssd-03
draft-eckert-anima-services-dns-autoconfig-01

IETF114 July 2022

Toerless Eckert (Futurewei USA), tte@cs.fau.de
Mohamed Boucadair, mohamed.boucadair@orange.com
Christian Jacquenet, christian.jacquenet@orange.com
Michael H. Behringer, michael.h.behringer@gmail.com
Summary

• Refresher
• Documents quite stable
  • Content/functionality, not necessarily text quality

• Please read the documents
Autonomic Networks..

BRSKI is autonomic (we hope... 0-touch.. except power/ethernet-cable)

But what if anything goes wrong.. troubleshooting?

How about ACP? “booted automatically after BRSKI”
First/most-simple Use-case: ACP for connectivity from NOC/SDN RFC8366

From experience with pre-standard ACP implementation:
Need a few infrastructure services to run AUTOMATICALLY
To successfully bring up BRSKI and ACP (troubleshoot)
For secure use of ACP by NOC(operator) / SDN-controller
Core ANI infrastructure services

Syslog to appropriate servers (in some NOC) across ACP
- Most fundamental troubleshooting tool
- For BRSKI/ACP itself (BRSKI-proxy)
- Also syslog from BRSKI registrar to notify of successful enrolment
- ACP nodes “just” need to learn which syslog server(s) to log to

Clock synchronization (to some NTP servers) across ACP
- Even just for basic crypto/certificate validation
- Tracing of event propagation (syslog originator timestamps, msec resolution)
- ACP nodes “just” need to learn about NTP server(s) to sync to
Core ANI infrastructure services

Remotely access ACP nodes via ACP (manually, from SDN controller)
  • SSH via ACP (controller use netconf via SSH)
  • Authenticate SSH login (username/credential) via Radius (or alternative)
  • Need to learn Radius server to use via ACP

Alternative/additional: New ACP nodes connect Netconf server via ACP
  • Netconf Call-home model
  • Need to learn ACP call-home server via ACP

Convenience ? Necessary ? Auto-enable DNS across ACP
  • Manual CLI operator actions (to eliminate need to know IPv6 addresses)
  • More and more router functions also want DNS ?!
  • Need to learn DNS server to use via ACP
Operator sets up servers in NOC – MUST SUPPORT IPv6

- Syslog, NTP, DNS, Radius/(Diameter/...), Netconf
- And connects them to ACP (e.g.: ACP-connect LAN interface in NOC)
- Nothing yet happens!

- Operator enables service announcements for ACP
- Configured servers could just announce DNS-SD
- Or the service announcement is feature on ACP-connect router
- Now all ACP nodes auto-start their syslog, NTP, DNS, radius, Network “agents” – and the ACP network is fully “manageable/useable”
- IMHO most crucial: whenever ANI network grows (troubleshoot enrolment BRSKI/ACP and any other following initial, automated steps)
- All services optional to enable by operator – only what's announced will run!
How to announce services/server across ACP

We have and want GRASP across ACP

• But we do not want to reinvent the wheel

DNS-SD has defined most of what we need

• Services names with IANA registry
• Service instance names, selection parameters (priority, weight, additional TXT params)
• None of these service aspects are specific to DNS
  • They just have been defined encoded into DNS only so far (probably there was a Localtalk encoding in Apple in before ?)
Great! How does it work?

- draft-eckert-anima-grasp-dnssd
- In DNS data for a single service is split across 4++ messages ("Resource Records")
- AAAA, CNAME, PTR, SRV, TXT, ...
- Unicast DNS discovering service requires multiple round-trips (when no cache)
- mDNS somewhat better, but still request/reply round-trips involved
- In GRASP, all service instance parameters are just one GRASP objective message
- Can easily add standard/custom parameters as well.
- If Objective name is SRV.<name>, then <name> must be an IANA registered service name. Aka: reuse existing registry!
Operationalizing/Using GRASP/DNS-SD

Ideal ?!

• Introduce application API for these service announcement/discoveries
  • Maybe draft can include suggested/minimum abstract API ?
• API Library can then use GRASP and/or DNS-SD/Unicast/mDNS as it chooses
  • Could be automatic. If ACP discovered, use GRASP, else DNS-SD/mDNS

On ACP router

• Simple announcer for services (configured)
• GRASP/DNS-SD ↔ mDNS gateway function (ACP-connect interface router)

MUST use this objective format.. (suggested requirements)

• ...when using a SRV.<name> objective name
• ...when wanting the objective to go across a GRASP/mDNS gateway
Summary

• draft-eckert-anima-grasp-dnssd-03
  • Specification for DNS-SD service compatible GRASP objectives
  • Recommended for ANY applicable service-instance announcements
  • E.g.: any new ASAs announcing service instances
  • Use SRV.name (and register name in services registry) when service should be compatible with mDNS/unicast-DNS
  • Use any desired objective name when meant to be used only across GRASP (and register in standard GRASP objective name registry)

• draft-eckert-anima-services-dns-autoconfig
  • Definition of core ANI infrastructure services ASA
  • Syslog, NTP, SSHd, Netconf-call-home, DNS-resolver, radius-authentication-client
  • Each activated by discovery of server instances
  • Goal is to create most simple fully-autonomic ANI to support RFC8366 (SDN, legacy “CLI”) style management.
  • Can be a template for similar pragmatic/incremental “automation” ASA
Ask

• Would like to see adoption call for these two drafts