DINRG & ANIMA
IETF102

T. Eckert, Huawei (tte@cs.fau.de)
v1.5
Summary

• Existing ANIMA work can serve as infra/dev platform for DINRG work
  • If DINRG solutions can leverage what ANIMA offers
  • And does not want/need to reinvent/improve it

• DINRG could help define guidelines or work for ANIMA
  • Like NMRG did for first charter round of ANIMA
  • ANIMA continues to look for definitions from NMRG,
    but DINRG likely a better source for multiple unresolved ANIMA items
Overview: From NMRG to ANIMA

- **NMRG** defined RFC7575/RFC7576 for **Autonomic Networks**:
  - **Goal**: evolve networks to be built with self-X (configuring, healing, managing, optimizing, protecting)
  - **Key building block**: **ASA** – Autonomic Service Agents. Distributed software modules embodying a distributed function/service on a node.
    - Managed by Intent (Q: what is Intent ?)
    - Leveraging a shared Autonomic Network Infra
  - This was the seed to charter ANIMA
    - Bottoms up, starting with ANI

---

**Figure 1: Reference Model for an Autonomic Node**

*from RFC7575 slightly enhanced*
Overview: ANIMA now

- Charter of ANIMA until now:
- **Build ANI**
  - Details next slide
- **Define two example validation documents**
  
  To show applicability of ANI
  
  **RFC8368** - use/benefits of ANI for classical centralized network management (“stable connectivity)

  **draft-ietf-anima-prefix-management** – automated prefix assignment for access interface via ANI (ACP/GRASP). First simple ASA. Prototype code:
  
  
  - documented at
  

Figure 1: Reference Model for an Autonomic Node from RFC7575 slightly enhanced
Autonomic Network according to ANIMA

Autonomic Network (AN)

Intent and Reporting
Network wide, abstract management

Autonomic Function N

ASA N

Autonomic Function 1

ASA 1
Autonomic Service Agent

Secure GRASP messages

GRASP
Generic Signaling

ACP - Autonomic Control Plane

BRSKI – Secure Bootstrap

ANI: Secure, reliable and automatic IPv6 NOC connectivity, Secure bootstrap, Zero touch service auto configuration

Domain wide (NOC and infrastructure) zero-touch certificates

Autonomic Network (AN):
Intent based network management

Controller

EMS

Domain Certificates

Reporting/Telemetry/Analytics

MSO OSS

Simple, legacy Management Tools

NOCS – Network Operations Center
OAM – Operation, Administration, Maintenance
ANIMA vs DINRG

• ANI should be a great tool for DINRG work
  • Eliminates the need to re-implement most fundamental common requirements for distributed software (e.g.: DINRG software / “ASA”)
    • **BRSKI: Bootstrap / Certificates**: Zero-touch bring-up of network (BRSKI)
      • Each node gets a certificate/trust anchor usable for any mutual authentication
    • **ACP: Addressing/secure-connectivity**: An IPv6 only management “VRF” with a lightweight routing protocol (RPL) is automatically build, and hop hop-by-hop encrypted and a simple (ACP).
      • Distributed software can securely and reliably talk to each other without requiring any SDN backend – BRSKI/ACP automate everything
    • **GRASP: discovery/protocol-session-layer-framework**: and A lightweight JSON/CBOR encoding protocol allows to easier design new protocol between distributed software components. Eliminates need for custom TLV protocols.
      • GRASP also provides automatic service discovery for distributed software components
Some ANIMA ideas/draft for simple network-wide configuration distribution, no model, languages, ...

**NMRG to the rescue ??**! Wants to define Intent better

**What distributed services ?**
Many idea draft for distributed services, one RFC in editor queue (distributed address management)

**DINRG to the rescue ??**! What distributed services are important to DINRG. Could they use ANIMA framework ?

**How to build distributed services**
APIs, design guidelines, .. Ides in ANIMA. Candidate next charter round work for ANIMA. **DINRG collab welcome**

**ANI: Result of ANIMA charter01**
provides a range of important functions
Improvements welcome
Decentralized alternative discussions ???
ANI does not manage user control/data!

• ANI is ONLY management plane!
  • Any DINRG work that is meant to manage/control/monitor the network
    • Is not in conflict with ANI
    • But can leverage ANI to make it easier to self-orchestrate

• Life without ANI:
  • See picture
    • Difficult to get from “unconfigured boxes” to “network where distributed software can run” – and depend on yourself to pull out of the mud.

• Example: distributed agents autoconfiguring addressing, IGP/iBGP.
  • Q: How do you ensure your distributed autoconfiguration agents can still talk to each other when their addresses or routes are not correctly autoconfigured?
  • Not only a day-0, but ongoing issue when there is ongoing autoconfiguration.
  • A: Agents can use ANI to talk to each other
    • its like a separate Mgmt network
ACP domains

- Distributed!
  - ACP Domains (e.g. @lake) consists of members that trust each other because of their certificates
  - ACP: Fully distributed autonomic building of secure IPv6 connectivity between members using these certificates between all members
  - GRASP: Fully distributed autonomic messaging including service-discovery

- How do pledges become a member?
  - Get a certificate, somehow

- And how do I do this...?
  - Next slide
Domain membership management

- Registrar
  - Drives/coordinates process
- Manufacturer (MASA) -> Voucher
  - Let Pled know Registrar may control it
- Admission Control
- Address allocation
  - Simple sequential allocation enough, but want to maintain database
- Certificate (signing)
  - Rely on certificate authority (CA)
    - Potentially a hierarchy.

- ACP/GRASP + BRSKI/EST = ANI
  - BRSKI/EST: Automated, secure instance of this:
    - Protocols/State-machineries
      - Pledge, Registrar CA, MASA

*Many non-decentralized components in this!!!*
Decentralized == “Federated”? vision

**Federated Ownership System**
- Transactions
  - Owner buys node from manufacturer
  - Owner resells node ...
- (Anonymous/public) Ownership-(claim) Ledger (pledge/owner)
- Federated across multiple manufacturers/resellers
- Reduce work for Mfgs
- Reduces need for owners to trust Mfgs
- Eases reselling
- ...

**Federated Domain member Ops**
- Transactions:
  - Owner pledges node to domain
  - Domain enrolls pledge including address ?!
  - Domain kicks member
- Ownership Ledger
- (Anonymous/public) Pledge/member Ledger (include address/…)
- One instance per domain ?
- Runs on Domain member nodes
- Domain Mgmt members may also be domain members

**Federated Domain Mgmt / Policy Ops**
- Transactions:
  - Propose Domain policies (change)
  - Consensus voting on policies ...
- Domain rules/policies
  - Member / Management: admit/eject
  - Member address allocation
  - Mgmt member policies ...
- One instance per domain ?
- Run on Domain management team nodes

- Technically interesting for ANIMA
  - But unclear about short term business acceptance (especially replacing sales receipts, CA)
Thank You
Some more technical details

• ANI uses RPL routing protocol because we did not want to invent a complex large-scale network self-configuration mechanism for addresses that can be aggregated (can DINRG do that please?).
  • RPL uses host-routes for network nodes, can scale to networks with >> 20,000 small-scale IoT nodes. Trick: Spanning tree routes (no shortest path), only routes away from root are remembered. Could support 100,000++ non-constrained (rfc7228) network nodes.

• GRASP not a complete protocol but a “common message encoding/exchange scheme”
  • For new protocols between distributed components
  • How would we have done encoding for IETF TLV protocols if we had today’s tools? (RIP, ISIS, OSPF, BABEL, NTP, DNS, PIM, IGMP, DHCP, ….
    100th more):
    • GRASP message structure uses JSON like encoding: CBOR is ~ binary JSON
      • Software sending/receiving GRASP packets therefor as easy to code as JSON app software (common in web apps)
      • Schema definition language for CBOR used to define new GRASP protocols messages: CDDL
      • GRASP itself defines few common headers – and discovery.

• GRASP not tied to ANI. Just use it for any new protocol you want to build.
  • You choose whether to run over TCP or TLS or UDP (or any other underlying transport)
Administrative thoughts (may be boring, WG-chair territory)

• When and what work to do in ANIMA
  • ANIMA is IETF-WG:
    • Focus on interop standardization. Work/specs must be precise enough to allow for interoperable implementations.
  • ANIMA is OPS-Area WG
    • Architectures, Frameworks, Use-cases less welcome than Specs and Yang models
      • IETF/OPS/AD area choice, not necessarily ANIMA WG-chair/participant preference
    • Goal is on enhancing operations.
    • Wide scope, but NOT reinventing wheels that exist.
      • ANI is defined through integration of existing technology components, incremental improvements of existing technologies, inventing only new when nothing existed (e.g.: GRASP protocol).
    • Standardization of mayor new complex or contentious items resulting from DINRG might potentially go to a different WG