I2RS, Mistake or Solution or …?

IETF #106
Singapore, November 2019

Dean Bogdanovic
Volta Networks
Quick Intro

In a way this is continuation from IETF #105
Network Automation Evolution
Presentation
Do we want to standardize split decision making between centrally located management and user applications and traditional routing protocols executed on network hardware?

Should all/most software run on the network hardware?

What should run on network hardware?
Are These Features Useful?

• Inject or retrieve information, policies and operational parameters to or from the routing system.
  – Read and write access to the Routing Information Base (RIB)
• Optimize, and choose network exit points. Base the decision on factors other than those provided by the routing protocols.
• Support a rapid, distributed reaction to network-based attacks. Reroute traffic for the destination under attack while maintaining normal operation for other routes.
• Modify service-layer routing to improve on existing hub-and-spoke traffic.
• Extract topology information from the network.
Do Traditional Vendors Support Something Similar?

• Yes, they do (or did in the past)
  – e.g. Junos SDK, Juniper JET, Cisco OnePK, …
• Proprietary interfaces, not the friendliest to third parties
• Lackluster support for it
What Constitutes A Network Element?

Network OS

Network hardware
Typical Network Device Components

- User Interface (CLI/GUI/API)
- Stats / Status (SNMP/Syslog)
- L2/L3 Protocol Daemons
- Boot Loader
- Other Hardware Management
- Common L2/L3 Data Plane APIs
- Kernel
- ASIC driver
- CPU
- DRAM
- Packet Forwarding ASIC
Bolting On More And More Things

Applications
Shared Infrastructure
I2RS Architecture

Application

I2RS Controller 1

I2RS Controller 2

I2RS Agent

Rtg Protocol A

Rtg Protocol B

RIB

FIB
I2RS Architecture

Application

I2RS Controller 1

I2RS Controller 2

Rtg Protocol A

Rtg Protocol B

RIB

FIB

Too Complicated Architecture
Initial Services Included In I2RS

- EGP
- Interfaces
- Dynamic data and statistics
- IGP
- PIM
- MPLS
- RIB Manager
Network Elements Layers

Applications

Shared Infrastructure and Data

Forwarding And Hardware Abstractions
Network Elements Layers

Applications

- Routing Protocols
- Network Services
- Network Management
- HW Platform and Chassis Management

Shared Infrastructure and Data

- Network state information data
- Data sharing infrastructure
- Conf and Ops infrastructure

Forwarding And Hardware Abstractions

- Forwarding abstraction
- SDKs
- Packet pipelines
Network Elements Layers

Forwarding And Hardware Abstractions
- Forwarding abstraction
- SDKs
- Packet pipelines

Shared Infrastructure and Data
- Network state information data
- Data sharing infrastructure
- Conf and Ops infrastructure

Applications
- Data models
- APIs
- Data models

Network Services
- Network state information data
- Data sharing infrastructure
- Conf and Ops infrastructure

Routing Protocols
- Network state information data
- Data sharing infrastructure
- Conf and Ops infrastructure

Network Management
- Network state information data
- Data sharing infrastructure
- Conf and Ops infrastructure

HW Platform and Chassis Management
- Network state information data
- Data sharing infrastructure
- Conf and Ops infrastructure

Data models
- Routing Protocols
- Network Services
- Network Management
- HW Platform and Chassis Management
There Is Something Similar Already Out There

RIB
Oper State

Configuration

Operational Commands

gRIBI
Routing Information Base Interface

gNMI
Network Management Interface

gNOI
Network Operations Interface

Figure source: https://datatracker.ietf.org/meeting/101/materials/slides-101-rtwg-sessa-grpc-services-on-network-devices-00
What Thought Experiments Can We Do?

• RIP I2RS
  – There are some good ideas, but the learnings from the group should be used in a new work

• Do we see a need to organize our work around new system architecture across multiple layers?