Gap Analysis for Autonomic Networking

draft-irtf-nmrg-an-gap-analysis-00

Michael Behringer
Brian Carpenter
Sheng Jiang

NMRG @ IETF 90
July 2014

Contents

- Introduction
- Current Status of Autonomic Behaviors
- Current Non-Autonomic Behaviors
- Approach toward Autonomy

Introduction

- Goals and definitions are from draft-irtf-nmrg-autonomic-network-definitions.
  - self-configuration
  - self-optimization
  - self-healing
  - self-protection
  - eliminate tedious and error-prone tasks
- This draft aims to identify status of autonomic behaviors and outline what is missing.
- Fairly small updates since IETF 89.

Status: Address management

- Address assignment is automated by SLAAC or DHCP[v6] (central policy via DHCP state).
  - But still widespread static addressing for servers
- DHCPv6 Prefix Delegation [RFC3633]
  - But still open issues in this (and nothing for IPv4)
    - (see pfister-homenet-prefix-assignment for a homenet approach to this)

Status: DNS

- DNS coordinated with addressing via central IP Address Management tools
  - Dynamic DNS Update is available too
- DNS server address provided by
  - DHCP[v6], which must be configured accordingly
  - RA option, which must also be configured in router

    (see mglt-homenet-front-end-naming-delegation and mglt-homenet-naming-architecture-dhc-options for a homenet approach to autonomic (m)DNS)

Status: Routing

- Routing and forwarding table computation is autonomic
  - routers need some initial configuration data to start up the autonomic routing protocol.
    - (see HNCP draft for a homenet approach to this)
  - BGP-4 routers need static configuration of routing policy data.

Status: Configuration of Default Router

- IPv4: Automatic with DHCP
  - but DHCP server must be configured consistently with routing setup
- IPv6: Automatic with RA
  - more complex Route Information Options also available but not supported by all O/S
  - IPv6 routing information via DHCPv6 is controversial; so is extending the role of RA
  - open issues when more than one prefix is in use on a subnet

Status: Security & AAA

- Many configured attributes are candidates for autonomic approach
  - management of user authentication information remains manual by network administrators
  - but it is essential that a network's central policy should be applied strictly for all security configuration
- Many security mechanisms show some autonomic properties, e.g.
  - PPP, RADIUS and Diameter automatically configure & account
  - negotiating crypto algorithms

but central configuration of policy remains.

Non-autonomic behaviors (1)

- Network establishment:
  - analyze the requirements of the new network
  - design network architecture and topology
  - decide device locations and capacities
  - etc. etc.
  - part of these jobs may be able to become autonomic
  - initial network management policies/behaviors might be transplanted from other networks and automatically localized
  - but this goal is difficult

Non-autonomic behaviors (2)

- Network Maintenance & Management:
  - New requirements of network services may not be able to be met quickly by human management.
  - Today, configuration of new devices depends either on human intelligence or rigid templates. This is the source of most network configuration errors.
  - Configuration updates after installing (or removing) devices are a prime candidate for autonomic techniques.
  - Self-adapting network configuration would adjust the network into the best possible situation, which also prevents configuration errors from having lasting impact.

Non-autonomic behaviors (3)

- Troubleshooting and Recovery:
  - Risk of overload of central or human management during major failures.
  - Associating warnings from multiple devices, together with automated learning techniques, could allow autonomic network diagnosis and troubleshooting.
  - Autonomic network management behavior may help reduce the impact of errors.
  - Software failures and configuration errors could be corrected autonomically.
  - Another possible autonomic function is predicting device failures or overloads before they occur.

Approach to autonomy: what's missing? (1)

- More Coordination among Devices or Network Partitions
  - Exchange knowledge between components
  - Horizontal as well as vertical information exchange
  - Detect and correct inconsistencies where they arise
- Don't rely on a superior intelligence except for general policy intent.
  - Do not wait for instructions before correcting or improving configuration.

Approach to autonomy: what’s missing? (2)

- Forecasting and Dry Runs
  - In a conventional network, configuration changes have to be designed and their probable effects have to be estimated theoretically (or with a complete and realistic network simulator).
  - There is a real risk that applying the changes to the running network will cause a failure.
  - An autonomic network could fill this gap with a "dry run" mode, in which a configuration change could be tested out in the control plane without actually affecting the data plane.

Also Missing Today: Benefit from Knowledge

- Historic knowledge is very helpful for correct decisions, in particular to reduce network oscillation or to manage resources over time.
- Transplantable knowledge from other networks can be helpful to initially set up a new network or new devices.
- Knowledge of relationship between network events and configuration may help network to decide according to real-time feedback.
- All these aspects today depend on humans rather than software applying the knowledge.

Questions? Discussion?

Other comments before approval?