Aggregate Manager Topics
GEC 15
Houston, Texas

Aaron Helsinger
October 23, 2012
www.geni.net

- # Stitching – Tom Lehman, ISI
  - Progress Report
  - Implementation Plan
  - Discussion

- # AM API v3
  - Status
  - Proposed Small Changes
  - Discussion

- # Update() – Rob Ricci for Jon Duerig, ProtoGENI / InstaGENI

- # Discussion and Next Steps

- Why stitching?
  - GENI promises custom layer 2 topologies.
  - To do this requires 'stitching' together resources from multiple aggregates.
  - VLANs let us do this across regionals and backbones

- History
  - Stitching architecture adopted at GEC10
  - Demo at GEC11+

- Today: Progress Report & Next Steps

Goal: Experimenters can create custom layer 2 topologies at GEC16

- Adopted March, 2012

- Implemented at ProtoGENI-Utah and InstaGENI-Utah

- Implemented in Omni and GCF sample aggregate

- Implementation in SFA/PlanetLab in process

- Supports AM API v3

- Defaults to AM API v2, GENI v3 RSpecs

- Auto re-direct to proper URL for specified AM API version, if the AM advertises it

- Various usability options
  - Specify multiple aggregates on commandline (`-a`)
  - Specify output filename for results (`--outputfile`)
  - Allow passing bad arguments for testing (`--devmode`)

- The AM API has (most of) what we need.

- Time to leave it alone
  - Use it
  - Avoid large changes

- Get experience with version 3 before making more big changes
  - Small changes and additions are possible…

Proposed Small Changes

- Monitoring has turned up some extra information and standardization that would help
  - Change Set N
- AM API v3 implementation has suggested some tweaks to string character restrictions
  - Change Set O

- Changes are posted on the GENI wiki and were sent to the dev@geni.net mailing list
- If adopted, aggregates may begin implementing these as part of their API v3 implementations

- Supply more information about the AM implementation to help monitoring and tools

- **Proposal N1**: `geni_am_code_version`
  - Standardize advertising the version of the AM software (duplicate `code_tag`, etc)

- How can tools know what kind of aggregate this is or how to find custom return fields?

- **Solution**:
  - N2: `geni_am_type`: New GetVersion return. Required

  - N3: `am_type`, `geni_am_type` and AM specific prefixes should all use the same string

- **Example**: `geni_am_type`=`protogeni` and new field in `code` is called `protogeni_error_log`

Proposed Change O1: Sliver names

- Sliver names were defined in AM API v3.

- **Problem**: Hyphen is the only legal separator character

- **Solution**: Also allow underscore and period
  - Note: We attempt to keep sliver names constant from the public ID form to the URN form.

- **Example**: `urn:publicid:IDN+plc+sliver +plc.bbn_slice_myslice-1234567`

- **Problem**: Some fields in the AM API are defined only as 'string'.

- **Solution**: Define regular expressions for
  - RSpec `type` and `version`: alphanumeric plus hyphen, underscore, period, colon
  - RSpec `schema` and `namespace`: valid URL characters
  - Credential `geni_type`: alphanumeric plus hyphen, underscore, period, colon
  - `geni_operational_status`: alphanumeric plus underscore

Proposed Change O3: Allow unicode

- International support is important to the future of GENI

- Many AM API values appear limited to ASCII

- But we don't want to require large changes now

- **Solution**:

  – Encourage unicode support without requiring it

  – `geni_unicode_compliant`: Optional field (default false) to advertise unicode support at an AM

- Concerns?

- Questions?

- Agreed?

- ## Why Update()?
  - Modify slice without losing resources

- ## History of the proposal
  - Discussed prior to and at GEC12
  - Email proposal from Jon D prior to GEC13
  - Verbal agreement at GEC14 coding sprint, details TBD

- ## Can include in v3, required in v4?

- Concerns?

- Questions?

- Agreed?

- Stitching:
  - Implementation well under way
  - Stitching RSpec v2
  - Topology, Path Computation & Workflow services
  - **Custom VLAN-based L2 topologies working at GEC16**

- AM API:
  - AM API v3 being implemented
  - Minor changes proposed

- Update()
  - Formal proposal published and discussed
  - Aggregates may not all implement for a while
  - Must formally document & adopt for API v4

- Come to the <mark>Coding Sprint Thursday 1pm</mark> to work out details

- Implement VLAN-based stitching by GEC16

- Implement AM API v3

- Discuss and adopt other changes for AM API v4+