

Control Framework Working Group Meeting
GEC6
November 18, 2009

1. Agenda

- Introduce new WG Chairs, position CFWG
- Framing Talks (30 minutes total)
 - Rob Ricci (Utah/ProtoGENI): Current approaches to resource representation
 - Guido Appenzeller (Stanford/OpenFlow): Spiral One integration experiences
 - Jeff Chase (Duke/ORCA): Next-generation need for scheduling, stitching, resource representation
 - Steve Schwab (SPARTA/Security): Next-generation need for identity and authentication services
- Panel discussion (60 minutes)
 - Guido Appenzeller (Stanford/OpenFlow),
 - Andy Bavier (Princeton/PlanetLab),
 - Jeff Chase (Duke/ORCA),
 - Aaron Falk (GPO),
 - Rob Ricci (Utah/ProtoGENI),
 - Steve Schwab (SPARTA/Security),
 - Ivan Seskar (Rutgers/WINLAB/ORBIT)
- Wrap-up and next steps (15 minutes)

2. New working group chairs

- Jeff Chase, Duke / ORCA
- Rob Ricci, University of Utah / ProtoGENI

3. Goal for Spiral 2: Define the aggregate API

- Requirements, function calls, resource representation
- Converge APIs of two of the control frameworks
 - PlanetLab and ProtoGENI both started with SFA
 - Reconcile naming, credentials, host and network RSpecs
 - Output: API with matching implementations
- Define missing elements needed for more general (next-gen) API
 - Framework for policies, scheduling, more general resource representation
 - Output: API

4. Current approaches to resource representation (Rob Ricci, Utah/ProtoGENI)

Three approaches, which differ greatly:

1. Ontology (ORBIT, ORCA):
NDL, semantic web as starting points.
Goal is to understand resources and their relationships
2. Extensible data structure (ProtoGENI)
RSpec is a data structure, tells you what you get not what you can build on it.
Semantics assumed (out of band), e.g. PCs, Ethernet
3. "To each his own" (PlanetLab)
Let each aggregate define its own RSpec

If you want to use that aggregate, you need to understand its rspec. CF is RSpec-neutral. Are these approaches compatible? Can we converge on a single model? Should we? How do these models fit into a larger resource discovery story?

5. Thoughts on control frameworks (Guido Appenzeller, Stanford/OpenFlow)

Thought #1: different resource types require different control frameworks, different UIs
Virtual machines are very different from eGENI switches. Ergo, group building aggregate should build control framework.

Thought #2: should centralize and federate at the top level, between control frameworks

Aggregates of same type share a CF. CFs of different aggregate types federate.

Thought #3: use technologies that are prevalent elsewhere

E.g. use REST, SOAP, JSON, SSL+secret ; don't use PKI, XML-RPC.

6. Next-generation need for scheduling, stitching, resource representation (Jeff Chase, Duke/ORCA)

Federation is likely to be tough. Weak federation is easy to implement, hard to use. How much can we gloss over that's going to be hard later?

Clearinghouse model brings up a number of hard questions. What does CH know? Does CH wield power? Do AMs trust CH? How much do AMs know about identity, federation? What commitments do AMs make to CF?

We COULD put policy anywhere, need to have some global policy control ("no you can't have all resources")

More questions about time, fairness, metering usage.

Questions about stitching.

Who can reason about overall topology?

How will we represent complex substrates?

Multiple layers, topology, location, state; declarative vs. imperative representation

7. Next-generation need for identity and authentication services (Steve Schwab SPARTA/Security)

Slices are now principals, too. Slices need to authenticate to other entities; using the user's credentials leads to bad design (needed by RAVEN; intra-slice operations). What does this mean for future GENI security? What hasn't surfaced yet that will bite us later?

Slices as principals means need to think about how to delegate rights to ephemeral principals. We'll need to support multiple sorts of identification (e.g. shib, OpenID, Kerberos, ...). We'll need different kinds of authorization (e.g. ACLs, attributes, assertions, delegation, inference, ...)

8. Panel Discussion

Q. What is the CF responsible for? Are we really talking about the same thing here?

Rob: CF is set of common abstractions, APIs, and structures adopted by aggregates and services. It's not about access control, it's about resource allocation

Steve: identity, trust has been pulled in, some is embedded in rspecs

Ivan: CF needs to know about substrate to do its job

Q. this was supposed to be the "narrow waist"

Ivan: sure, bare minimum, but we need some info

Guido: narrow waist or least common denominator? What is common between openflow switch and planetlab node? Virtualize in different ways, different attributes, ... Need to at least allocate, reserve, and manage resources.

Jeff: there is a lot of difference between aggregates. Would prefer to think about who the players, stakeholders are, and then decide what to build.

Andy: In pretty close agreement with Rob; at the end of the day, facility to let users create experiments.

Guido: would like a completely general control framework; I don't think it's possible. I haven't been able to do it.

Rob: in general I agree, but both pl node and openflow switch have idea of create sliver, delete sliver, ... The way that they are managed is different, but that's the AMs job

Guido: I agree that when resources are similar enough, handle them together; but there is not commonality.

Q: isn't there some commonality?

Ivan: there are certainly commonalities, but they are few

Jeff: we really need a rich and powerful language to be able to talk about resources

Guido: it may be possible to use a language for that, but what do highly successful resource reservation systems do? Expedia has different ones for planes, hotels, cars, and federates them

Q: as an experimenter, what is common across CFs?

Rob: each CF is a new learning experience

Jeff: part of what we're doing in spiral 2 is make this better

Steve: worth building your own autoconf tools

Q: We've fallen back to user==experimenter. We conflate human resource discovery with machine-based resource discovery. We're ignoring interfederation technologies.

Steve: A lot of work has been done to federate identity. The harder decision to make is whether we need all the complexity from the security "tribe"

Rob: I don't see any evidence that end users need to be GENI principals

Aaron: GPO model for users has been experimenters, but that might include students

Jeff: something like shib makes supporting transient users (students) much easier

Rob: it hasn't been a problem

Jeff: I'd rather have a system where we agree with GENI that all of our students have access, not registering each user separately. We should reduce the burden when we can

Rob: it hasn't been an issue

Guido: as long as we can delegate administration, we don't care

Q: Expedia gives reservation across the three substrates there is commonality (location); i.e. stitching. Having a common language that spans substrates helps stitch things together

Ivan: Do we even know what experimenters want to stitch?

Guido: how you stitch together aggregates depends on the aggregate

Andy: "to each his own" might better be called "bottom up"—solve what we can, postpone rest

Q. Why isn't this just network management?

Jeff: it's an orchestration problem, not just a resource mgmt problem

Steve: network management timescales are different from testbeds

Guido: we can leverage

Q. (Rick McGeer) why don't you agree on what you agree on first, then go from there?

Guido: we agree on SOAP/REST, XML RSpecs

Steve: low-level implementation details aren't that important

Q (John Wroclosky) First time we've had a panel with nobody from the original planning group; we're going over old ground. Stitching is not a narrow-waist problem; should be handled by a tool. Ergo it's not part of the CF. Second, tension between coding things up that we understand and making something new for the ages. Third, I think the CFWG should focus on things that last between spirals.

Jeff: I appreciate your minimalist approach. I see minimalism as extensibility

Rob: one of the points of the CF is to do what the user can't do.

9. Wrap-up and next steps

Need to identify most important points we brought up, and dribble them out on the mailing list to get conversations started.

We should enumerate things we agree on

John W.'s minimalism argument

Maybe we should model all resources as hotel rooms?

We need to agree on a common vocabulary

There were provocative and hard questions in the slides