

# Clearinghouse and Monitoring: operational status updates

Chaos Golubitsky

July 22, 2013

[www.geni.net](http://www.geni.net)



<http://groups.geni.net/geni/wiki/GeniTrustAnchors>

- Goal: experimenters can use consistent set of AMs with accounts from any trusted authority:
  - GPO clearinghouse: ch.geni.net
  - Utah ProtoGENI: emulab.net
  - PlanetLab Central: planet-lab.org
  - GPO legacy SA: pgeni.gpolab.bbn.com
- As of GEC17, these are trusted by:
  - ExoGENI and InstaGENI rack compute and FOAM AMs
  - Core and mesoscale FOAM AMs
  - ProtoGENI federation
- Using nagios to “regression test” trust relations

# Making ch.geni.net operational

GEC16: how is GPO making portal/CH operational?

- Rename: from panther.gpolab.bbn.com to:
  - portal.geni.net: the web frontend (a tool trusted by CH)
  - ch.geni.net: CH services for users, projects, slices
- Improved user management, delete, audit (in progress)
- Management improvements:
  - Clean rebuild of production, staging, sandbox CHes
  - Better nagios monitoring of portal/CH service health
  - Cleaner code deployment and tracking
- Monitoring: ch.geni.net slice data to GMOC

# Operational monitoring status report

- Status of gmoc-db monitoring and APIs:
  - Relational data:
    - Submission and download via authenticated queries (gmoc.py)
    - Accounts for operators only right now – access to all GENI data
  - Measurement data:
    - Submission via authenticated GMOC XML format (gmoc.py)
    - Download as unauthenticated XML, has to be hand-parsed
    - No measurement submission with sensitive metadata
  - What's being submitted?
    - Resource information: ExoGENI, InstaGENI, FOAM
    - Sliver information: InstaGENI, FOAM
    - Some resource measurements: ExoGENI, FOAM
    - Slice data from authorities: ch.geni.net, pgeni.gpolab.bbn.com
    - Some remote aggregate health checks

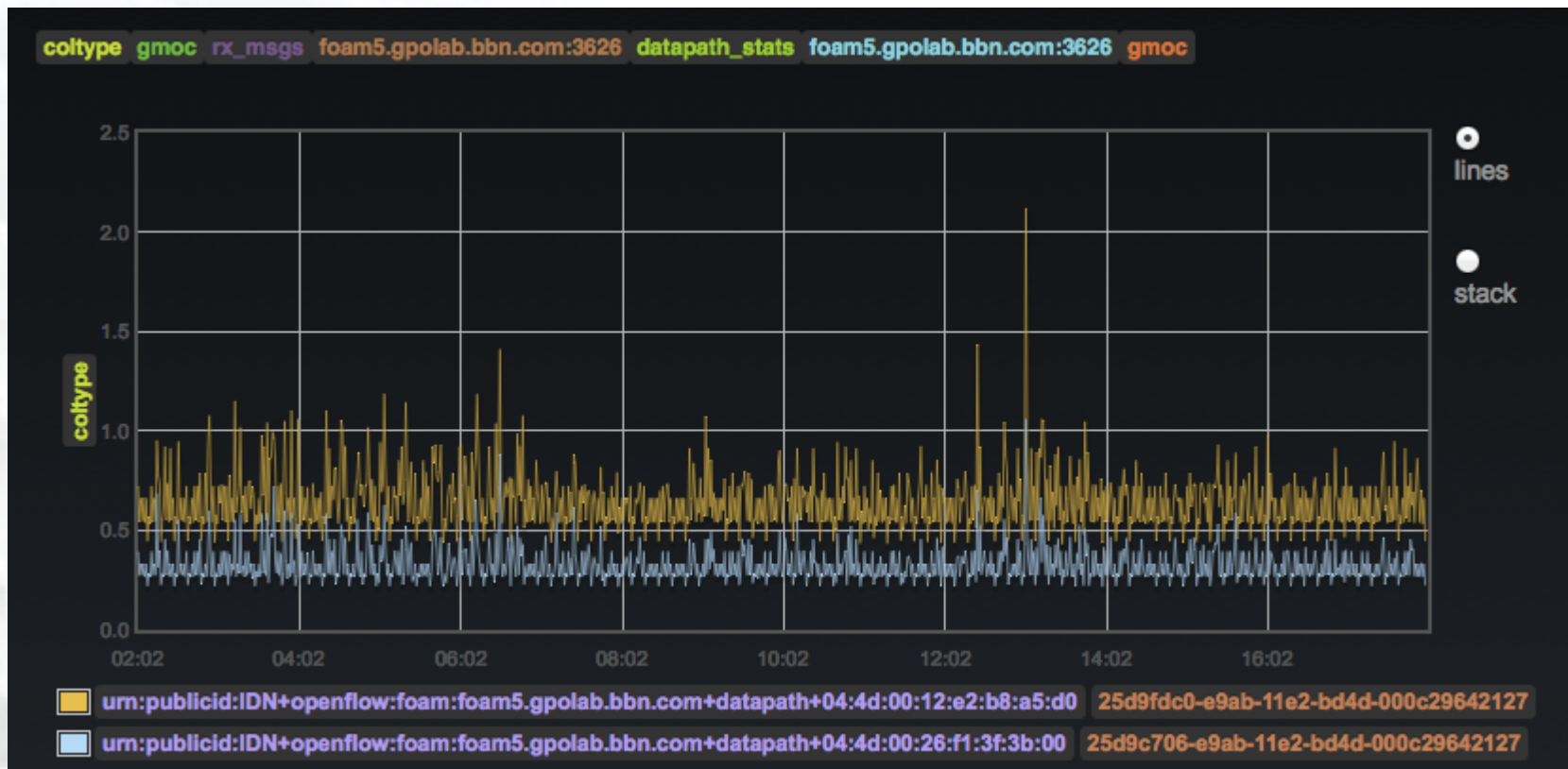
# Prototype measurement metadata (1)

- Problem: measurement data is not yet tied robustly to relations:
  - We know some VM server resource is running 8 VMs, but can't deterministically learn that the VM server is  
`urn:publicid:IDN+utah.geniracks.net+node+pc3`
- Prototype solution:
  - Submit measurements using the gmoc.py client
  - Specify GENI objects as “tags” in a portable way
  - Use JSON submission encoding to send tags with data
  - Store measurement/object mapping in GMOC database
  - Store data on disk using easy-to-manage UUIDs

# Prototype measurement metadata (2)

Example:

- FV rx\_msgs from 2 datapaths on one aggregate
- Labelled with datapath URNs from relational data





- Data submission:
  - Finish submission of defined data from all aggregates
  - Submit sliver data for GENI slices from all SAs
  - Relations/measurements for networks and stitching
  - Procedure for adding new measurement types
- Data retrieval:
  - Implement retrieval of tagged measurements
  - Keep improving privacy/ACL model
- Data use:
  - More nagios tests of GENI health using downloaded data