LA-UR-24-23077

Approved for public release; distribution is unlimited.

Title: OpenCHAMI for collaborators and the collaborator-curious

- Author(s): Cotton, Travis Bradley
- Intended for: Cray User Group, 2024-05-05/2024-05-09 (Perth, Australia)

Issued: 2024-04-04









Los Alamos National Laboratory, an affirmative action/equal opportunity employer, is operated by Triad National Security, LLC for the National Nuclear Security Administration of U.S. Department of Energy under contract 89233218CNA000001. By approving this article, the publisher recognizes that the U.S. Government retains nonexclusive, royalty-free license to publish or reproduce the published form of this contribution, or to allow others to do so, for U.S. Government purposes. Los Alamos National Laboratory requests that the publisher identify this article as work performed under the auspices of the U.S. Department of Energy. Los Alamos National Laboratory strongly supports academic freedom and a researcher's right to publish; as an institution, however, the Laboratory does not endorse the viewpoint of a publication or guarantee its technical correctness.

Birds of a Feather

OpenCHAMI at CUG 2024



What's an OpenCHAMI?



- Open Composable Heterogeneous Adaptable Management Infrastructure
- AKA OCHAMI or ochami
- From unknown hardware to SLURM
- HTTP Microservices in containers



Discussion Topics

- Core Services vs Extra Services
- Deployment Strategies
- Performance at Scale
- Development Practices



Core Services vs Extra Services

- Magellan, SMD, BSS
- Dhcp, krakend
- Authentication
 - o Opaal
 - JWTs
- Post boot configs
 - Cloud-init
 - o Ansible
- Other hooks?
- Other missing functionality





Services currently in use (at LANL)

- Magellan
- SMD
- BSS
- Dnsmasq-dhcp
 - Reads from SMD
 - Updates every 10 seconds
- Krakend-ce
 - API gateway
 - Still working through this
- Ory Hydra
 - OAuth server
 - OpenID connect provider
- Opaal
 - OAuth client



What should be added?



- Are the current list of core services enough?
- What should be considered core vs extra?
- What should be added and supported by OpenCHAMI?
- What would you like to see?



Deployment Strategies

- Docker Compose
- Kubernetes
- Openshift
- GKE
- On Switches?
- Others?





Deployments at LANL

• Docker compose

- Two main deployments
- SI clusters (student clusters) 10 nodes each
- Badger old crusty hardware- ~650 nodes
- Plan on trying a OpenShift deployment
 - Still in the works





How do we deploy?



- What would people like to see?
 - Preferred deployments?
 - Cloud vs local?
- Any deployments to avoid?
- Should we care where/how OpenCHAMI is deployed?



Performance at Scale



- 512 nodes per failure domain
- Scaling dimensions
 - Orchestrating multiple failure domains
 - Adding Gateways and/or Load Balancers
 - Caching Layers
 - Per-ToR deployments



How do we handle scale?



- How would like to see OpenCHAMI run at scale?
 - Multiple Failure domains
 - gateways/loadbalancers
 - Top-of-Rack?
 - Switches?
- Opt-in Complexity? start simple?
 - Should be able to handle very small and very large systems
- Concerns about running at scale?



Development Practices

- Adding Services
- Adding to Existing Services
- Removing/Rewriting Code
- DevOps and CI/CD
 - Hosting testing hardware
 - CI/CD Pipelines
 - Unit Testing
 - Software Supply Chain





Development at LANL

- 15 minute daily standup
- Code reviews
- CI pipeline testing
 - For individual OpenCHAMI services
 - Integration testing





Development



- How to add/modify/remove/whatever the OpenCHAMI code base?
 - Get involved! <u>https://github.com/OpenCHAMI</u>
 - Multi-site development meeting? Weekly? Monthly?
- CI/CD pipeline testing
 - Do you want to help?
 - Hosting testing hardware?



Contact - How to complain



- Slack: openchami.slack.com
- Github: https://github.com/OpenCHAMI

