Enabling Docker for HPC
CUG 2018

Jonathan “Bill” Sparks, Cray Inc.

Motivation

- **Purpose**
  - Demonstrate that Docker can be used in the HPC context
  - Highlight areas of investigation and development
    - Authentication, storage, resource access
- **Benefit**
  - Standard container implementation Docker
  - Integration with existing orchestration software
    - Kubernetes, Swarm, Nomad, Mesos

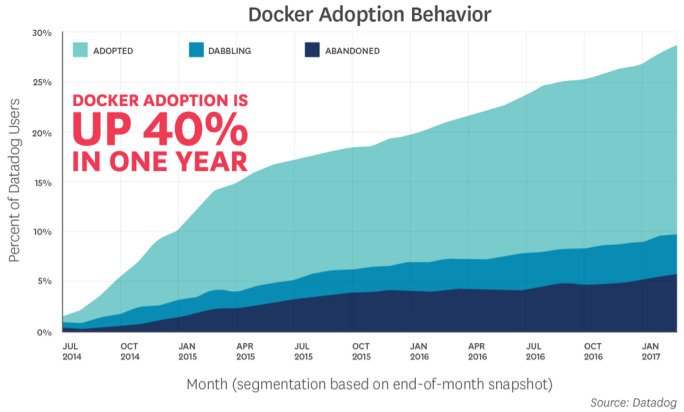
Background

- **Fragmented solution space**
  - Enterprise: Docker, CoreOS
  - HPC: Shifter, Charliecloud, Singularity
  - Orchestrators: Swarm, Kubernetes, Mesos, Nomad
- **Fragmented feature set**
  - Volumes, networking, authorization, capabilities
- **Fragmented user experience**
  - CLI options, APIs, usage patterns

Rationale

- **Use standard Docker in the HPC context**
- **Standardization of software implementation**
  - **Docker**, CoreOS rkt, Shifter, Singularity, Charliecloud
- **Integration with existing orchestration software**
  - Kubernetes, Swarm, Nomad, Mesos
- **Larger addressable market**
  - Compatibility with existing software

https://www.datadoghq.com/docker-adoption

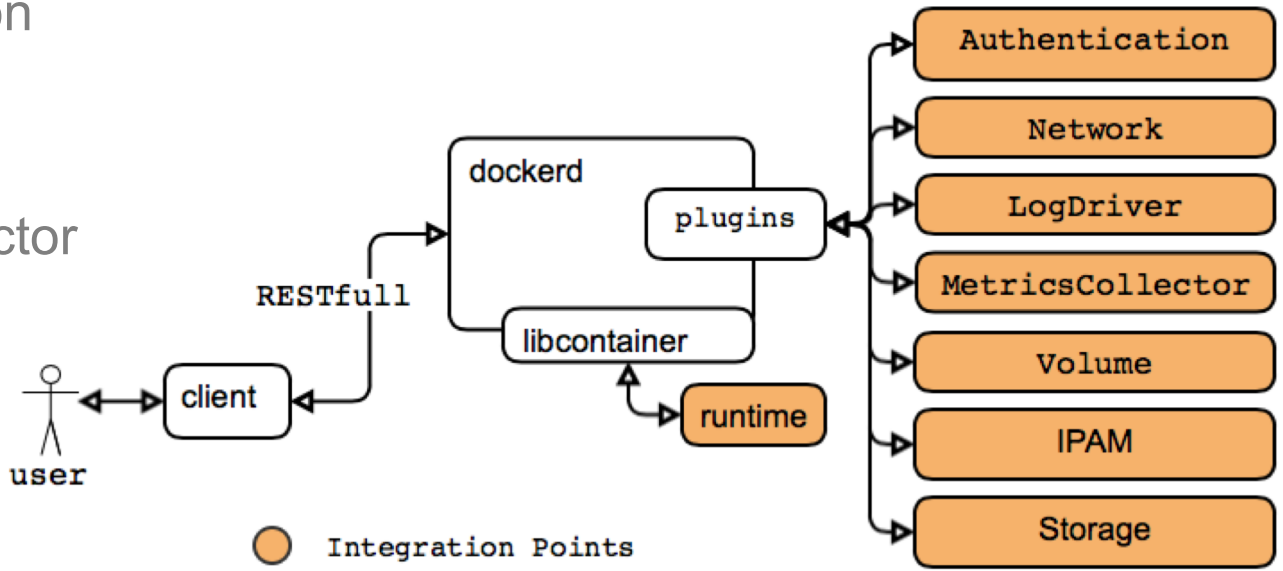


Investigation

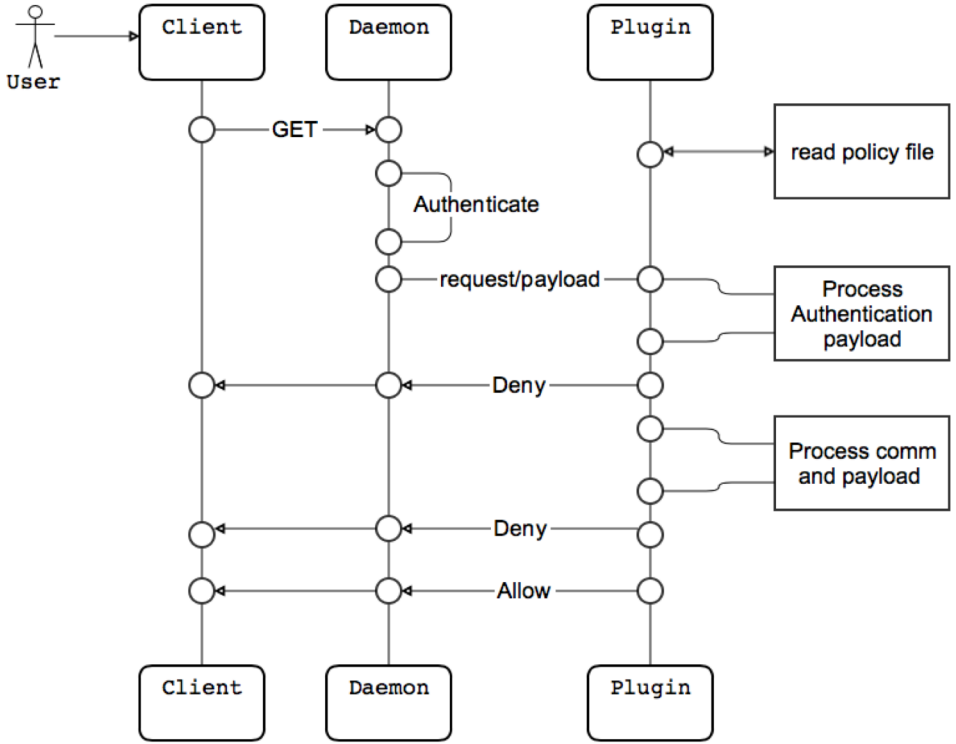
- **Docker plugin architecture**
  - Authentication
  - Storage
  - Isolation and resources
- **Investigate prior studies in this area**
  - Authentication
  - Storage
- **Collaboration between Cray, Docker, and customers**
  - Docker, Cray R&D, Cray customers

Docker Extension Architecture

- **Plugins/Drivers**
  - Authentication
  - Network
  - Log
  - MetricsCollector
  - Volume
  - IPAM
  - Storage
  - Runtime



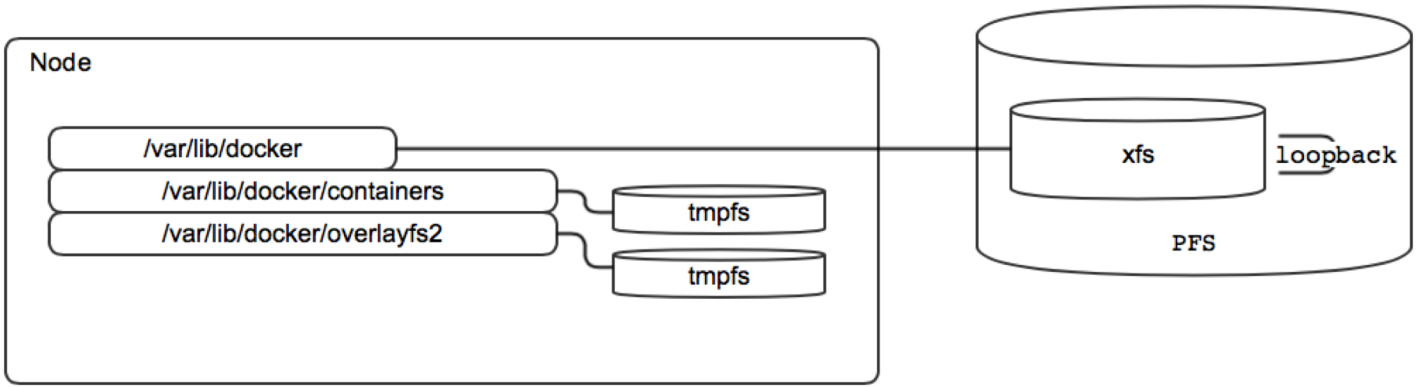
Authentication – AuthZ Plugin




```
{
  "name": "policy_1",
  "users": [
    "authclients"
  ],
  "allow_actions": [
    "container_create"
  ],
  "deny_payloads": [
    "Privileged",
    "CapAdd"
  ]
}
```

Storage – Graph Plugin

- **Support for diskless nodes**
- **Shared PFS storage**
  - Hybrid approach
  - Overlayfs2 was introduced – Docker 1.12 - 07/28/2016



Isolation & Resources

- **Namespace isolation**
  - Use virtual resources
  - Drop kernel namespace for device/RDMA access (passthrough)

```
docker run --net=host \
           --device=/dev/infiniband/uverbs0 \
           --device=/dev/infiniband/uverbs1 \
           --device=/dev/infiniband/rdma_cm -ti --rm centos radioss.job
```

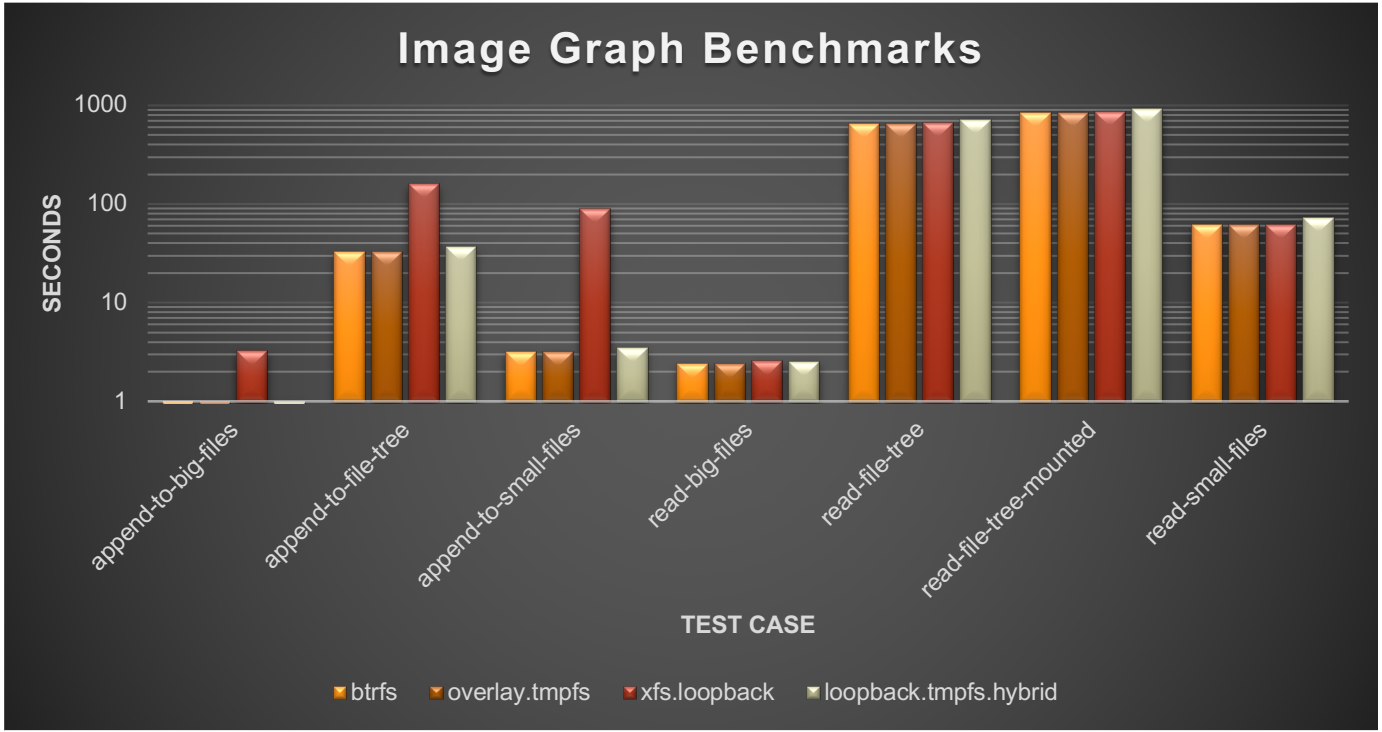
Authentication

- **Enhanced AuthZ CA certificate authentication**

```
# full disclosure (TLS set)
DOCKER_TLS_VERIFY=1
DOCKER_CERT_PATH=$HOME
$ docker  run -ti --rm --user=$(id -u) --cap-add CAP_SYS_ADMIN centos date
docker: Error response from daemon: authorization denied by plugin authz-broker:
        action 'container_create' and 'CapAdd' not allowed for user 'client' by policy 'policy_1'.
```

{"name":"**policy_1**","users":["**client**"],"actions":["docker_*","image_*","**container_***"], ”excl_payloads":["UsernsMode","**CapAdd**"]}

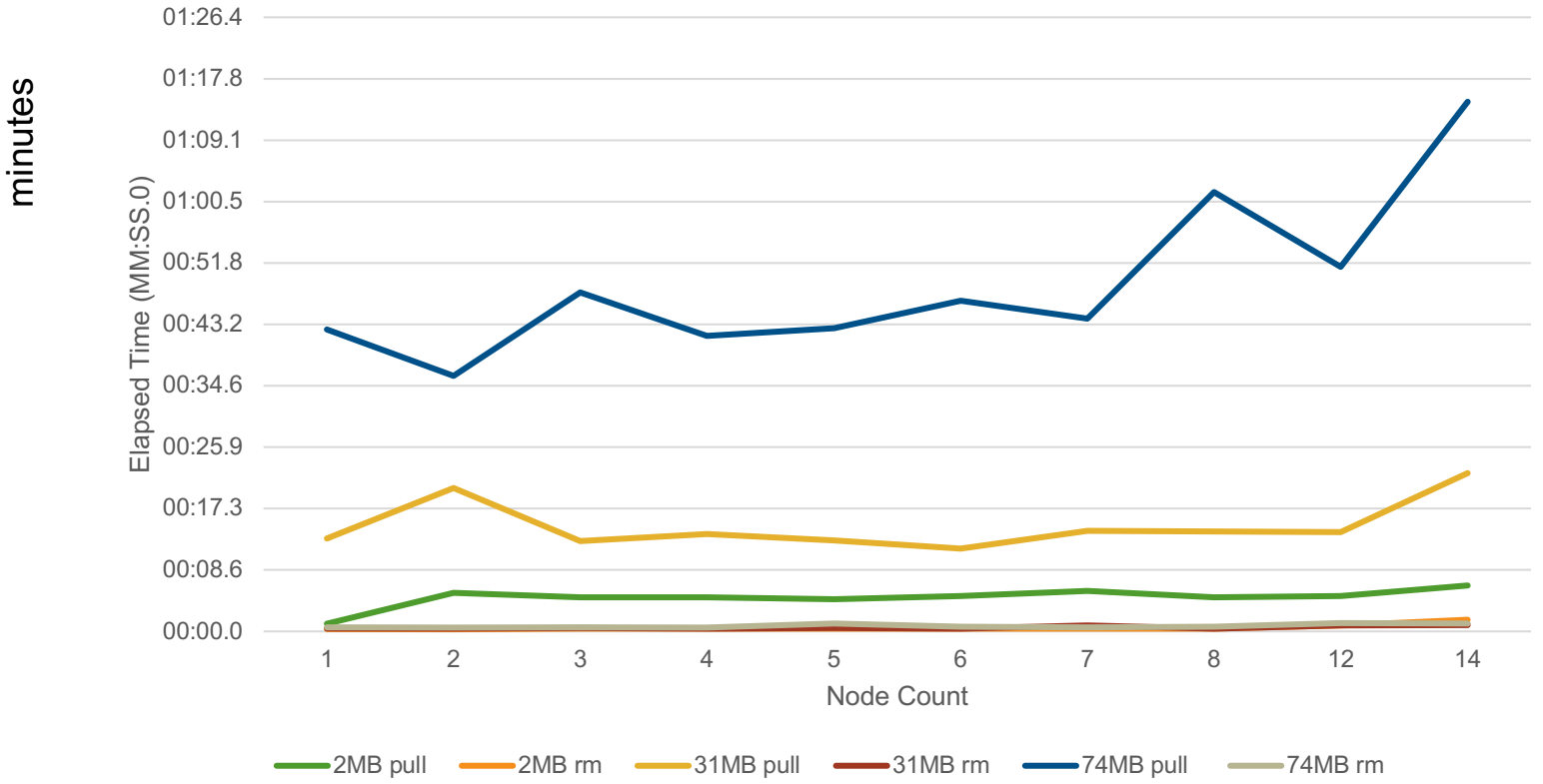
Storage Graph Benchmarks Results



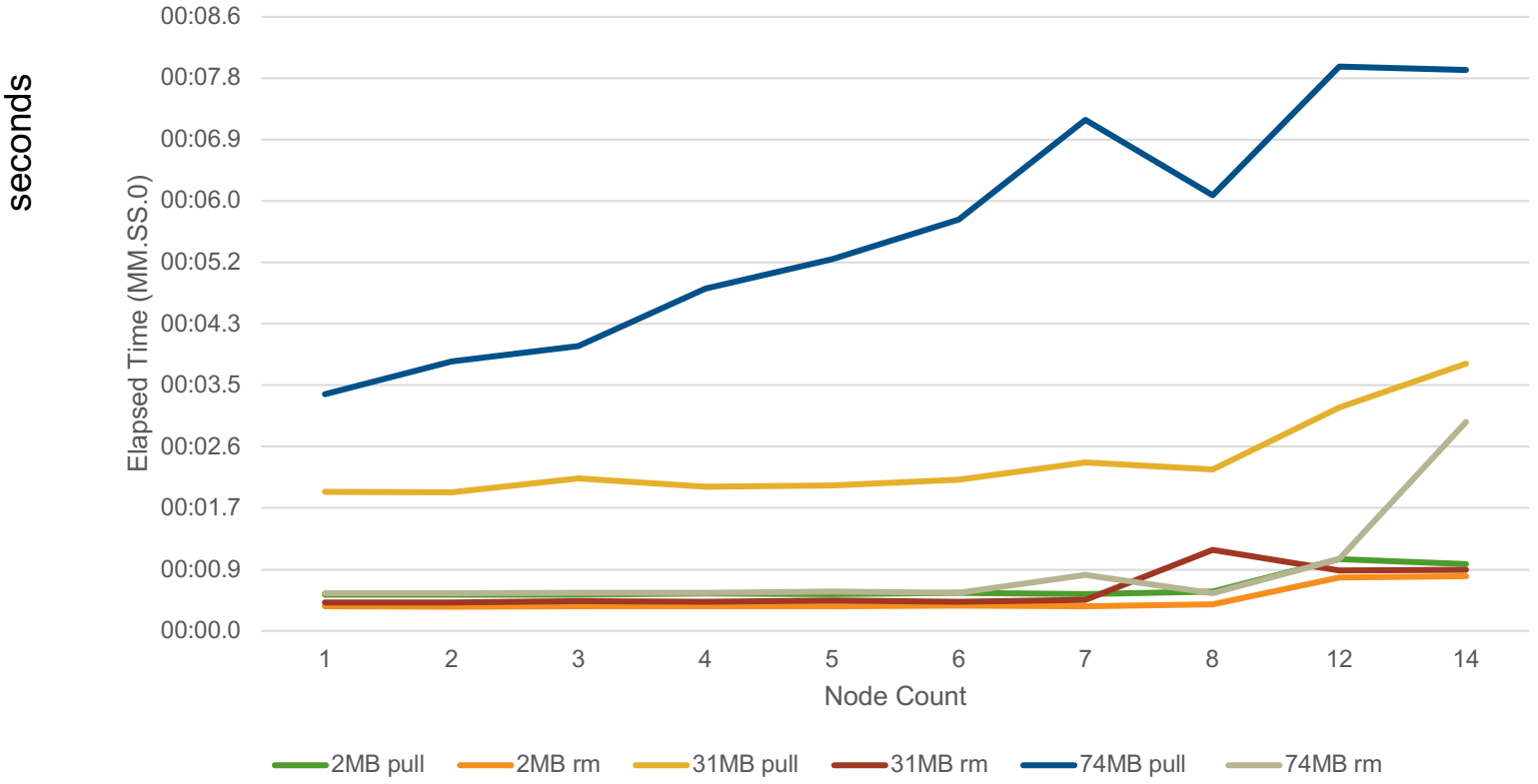
small files: 512 B
large files: 1,8,32,128 MB

https://github.com/chriskuehl/docker-storage-benchmark

Docker Image Results – Dockerhub



Docker Image Results – Local Registry



Conclusions and Future Work

- **Plugin infrastructure**
  - User enforcement plugin support needed
  - Storage plugin for shared HPC storage backend support
- **Investigation and collaboration**
  - Working with Docker, Inc to investigate architectural changes
    - User enforcement
    - Shared HPC storage
    - WLM batch integration
  - Identify HPC requirements and extensions
  - Scalability and host resource access

Legal Disclaimer

Information in this document is provided in connection with Cray Inc. products. No license, express or implied, to any intellectual property rights is granted by this document.

Cray Inc. may make changes to specifications and product descriptions at any time, without notice.

All products, dates and figures specified are preliminary based on current expectations, and are subject to change without notice.

Cray hardware and software products may contain design defects or errors known as errata, which may cause the product to deviate from published specifications. Current characterized errata are available on request.

Cray uses codenames internally to identify products that are in development and not yet publicly announced for release. Customers and other third parties are not authorized by Cray Inc. to use codenames in advertising, promotion or marketing and any use of Cray Inc. internal codenames is at the sole risk of the user.

Performance tests and ratings are measured using specific systems and/or components and reflect the approximate performance of Cray Inc. products as measured by those tests. Any difference in system hardware or software design or configuration may affect actual performance.

The following are trademarks of Cray Inc. and are registered in the United States and other countries: CRAY and design, SONEXION, URIKA and YARCDATA. The following are trademarks of Cray Inc.: CHAPEL, CLUSTER CONNECT, CLUSTERSTOR, CRAYDOC, CRAYPAT, CRAYPORT, DATAWARP, ECOPHLEX, LIBSCI, NODEKARE, REVEAL. The following system family marks, and associated model number marks, are trademarks of Cray Inc.: CS, CX, XC, XE, XK, XMT and XT. The registered trademark LINUX is used pursuant to a sublicense from LMI, the exclusive licensee of Linus Torvalds, owner of the mark on a worldwide basis. Other trademarks used on this website are the property of their respective owners.


Q&A
Jonathan “Bill” Sparks
jsparks@cray.com