MongoNAS

HPC Academy 2019

Jordan Dorham | Jasmin Agustin | James Taliaferro

August 15, 2019

LLNL-PRES-788759

This work was performed under the auspices of the U.S. Department of Energy by Lawrence Livermore National Laboratory under contract DE-AC52-07NA27344. Lawrence Livermore National Security, LLC

Lawrence Livermore National Laboratory

The MongoNAS Team

Jordan Dorham
Cal State Long Beach

Jasmin Agustin
Cal State Long Beach

James Taliaferro
UC Santa Cruz

Overview

- The Old Way
- The Problem
- The Solution!
- Under The Hood
- Demo
- The Future

The Old Way

- SSH to Net App filer and use bash to produce user storage information
- Perl Script to make data human readable
- Data pulled from text files into web-app interfaces

The Problem

- Lots of parts leads to a lot of possible failure points
- Data is not in a single regularly updated location
- Data has minimal structure and lacks depth
- Difficult to query
- Text files are not easily transferred to the web

```
[dorhami@oslic2:FileUsageInfo]$ ls
cz_gapps-quota-report      yipyips.collab_repo.old      yipyips.cz_g12      yipyips.cz_g16.old  yipyips.cz_g21      yipyips.cz_g90.old  yipyips.cz_gdata
old                        yipyips.collab_repo_ng       yipyips.cz_g12.old  yipyips.cz_g17      yipyips.cz_g21.old  yipyips.cz_g91      yipyips.cz_gdata.old
yipyips.collab_gapps       yipyips.collab_repo_ng.old   yipyips.cz_g13      yipyips.cz_g17.old  yipyips.cz_g22      yipyips.cz_g91.old  yipyips.cz_give
yipyips.collab_gapps.old   yipyips.cz_g0                yipyips.cz_g13.old  yipyips.cz_g18      yipyips.cz_g22.old  yipyips.cz_g92      yipyips.cz_give.old
yipyips.collab_gdata       yipyips.cz_g0.old            yipyips.cz_g14      yipyips.cz_g18.old  yipyips.cz_g23      yipyips.cz_g92.old  yipyips.cz_global
yipyips.collab_gdata.old   yipyips.cz_g10               yipyips.cz_g14.old  yipyips.cz_g19      yipyips.cz_g23.old  yipyips.cz_g99      yipyips.cz_global.old
yipyips.collab_global      yipyips.cz_g10.old           yipyips.cz_g15      yipyips.cz_g19.old  yipyips.cz_g24      yipyips.cz_g99.old
yipyips.collab_global.old  yipyips.cz_g11               yipyips.cz_g15.old  yipyips.cz_g20      yipyips.cz_g24.old  yipyips.cz_gapps
yipyips.collab_repo        yipyips.cz_g11.old           yipyips.cz_g16      yipyips.cz_g20.old  yipyips.cz_g90      yipyips.cz_gapps.old
```

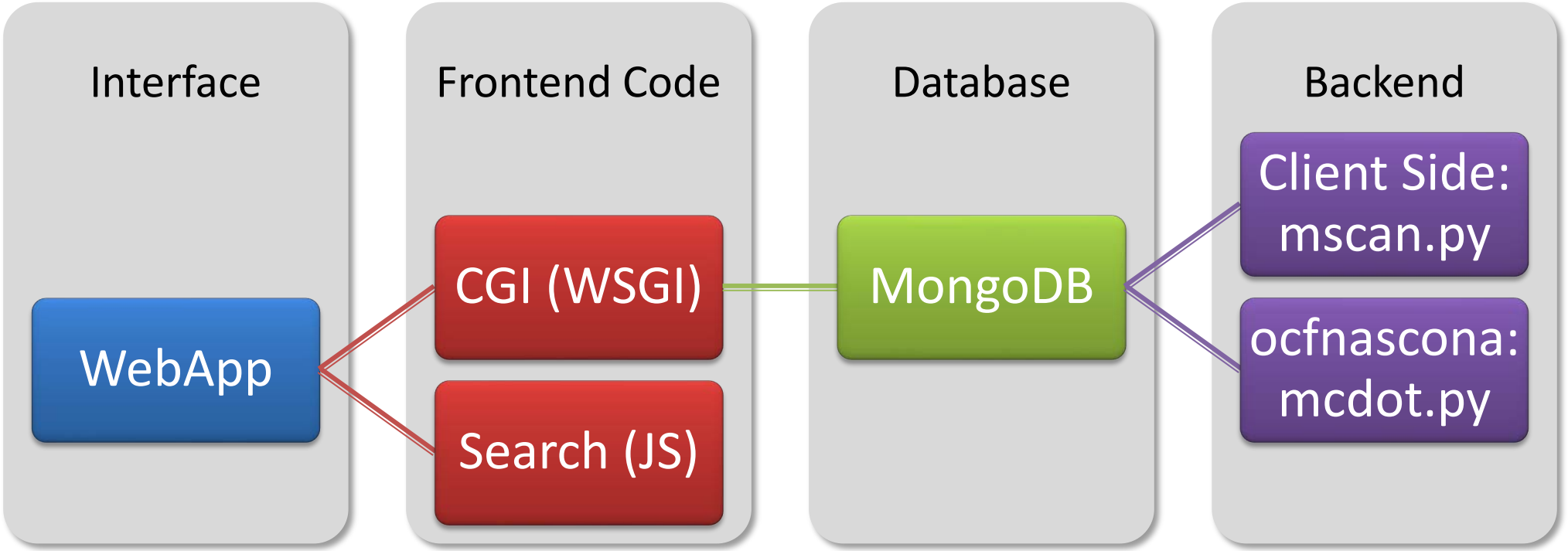
User	Size	Files
miller86	26.99GB	366364
tweis	23.86GB	248581
schneide	23.22GB	47261
landa	22.92GB	289576
stowell	22.11GB	242834
draeger	21.09GB	267623
akupres	20.10GB	82797
cah	19.09GB	21472
brown86	17.06GB	27012
puso	15.76GB	10885
caldwep	15.14GB	307633
benedict	15.13GB	14182
fried	15GB	19740
oh4	14.77GB	14427
mmorale	14.52GB	37261
chen41	14.07GB	84648
gokhale2	13.98GB	157479
streitz	13.71GB	117448
bennion1	13.67GB	212666
neely4	13.61GB	67398
acunning	13.43GB	17797
chase3	12.65GB	8469
lucas26	12.49GB	92097
jsc	12.45GB	14676
daniel	12.39GB	52457
luton2	12.10GB	51988
felice	12.02GB	18796
glascoe1	11.99GB	46276
u970344	11.55GB	6183
ilamni	11.54GB	2701
whitley3	11.37GB	196302
jjr	9.39GB	339
chambers	8.77GB	26015
jbogden	7.74GB	106057

```
/collab/usr/gapps/python:lee1001:2.50GB:2160:96GB
/collab/usr/gapps/python:lee218:84.04GB:2203785:96GB
/collab/usr/gapps/python:qtree:Qtree:Name:96GB
/collab/usr/gapps/python:taylor:4KB:1:96GB
/collab/usr/gapps/roguewave:38451:0B:1:10GB
/collab/usr/gapps/roguewave:ALL:0B:n/a:10GB
/collab/usr/gapps/roguewave:duthie1:0B:2:10GB
/collab/usr/gapps/samrai:ALL:3.71GB:n/a:10GB
/collab/usr/gapps/samrai:rwa:2.80GB:88683:10GB
/collab/usr/gapps/samrai:ukbeck:934.5MB:57758:10GB
/collab/usr/gapps/shroud:ALL:153.0MB:n/a:10GB
/collab/usr/gapps/shroud:root:0B:1:10GB
/collab/usr/gapps/shroud:taylor:153.0MB:8139:10GB
/collab/usr/gapps/stapre:ALL:0B:n/a:10GB
/collab/usr/gapps/stapre:kelley24:0B:1:10GB
/collab/usr/gapps/toss_3_x86_64:ALL:0B:n/a:1MB
/collab/usr/gapps/toss_3_x86_64:djd:0B:1:1MB
/collab/usr/gapps/toss_3_x86_64:root:0B:2:1MB
/collab/usr/gapps/toss_3_x86_64_ib:ALL:0B:n/a:1MB
/collab/usr/gapps/toss_3_x86_64_ib:djd:0B:1:1MB
/collab/usr/gapps/toss_3_x86_64_ib:root:0B:2:1MB
/collab/usr/gapps/tracker:ALL:0B:n/a:10GB
/collab/usr/gapps/tracker:chase3:0B:1:10GB
/collab/usr/gapps/tree:ALL:disk-used:n/a:disk-limit
/collab/usr/gapps/uk:ALL:0B:n/a:10GB
/collab/usr/gapps/uk:shale:0B:3:10GB
/collab/usr/gapps/uq:51209:126.9MB:2535:10GB
/collab/usr/gapps/uq:ALL:9.48GB:n/a:10GB
/collab/usr/gapps/uq:afillmor:27.75MB:194:10GB
/collab/usr/gapps/uq:dahlgren:1.25MB:1:10GB
/collab/usr/gapps/uq:ddom:17.09MB:206:10GB
/collab/usr/gapps/uq:minner2:4KB:1:10GB
/collab/usr/gapps/uq:vnvadm:9.31GB:13304:10GB
/collab/usr/gapps/vampire:ALL:703.0MB:n/a:10GB
/collab/usr/gapps/vampire:strozzi2:703.0MB:15645:10GB
/collab/usr/gapps/visrad:ALL:1.08GB:n/a:10GB
/collab/usr/gapps/visrad:jay:1.08GB:8971:10GB
/collab/usr/gapps/visrad:root:0B:1:10GB
/collab/usr/gapps/wci:ALL:0B:n/a:10GB
/collab/usr/gapps/wci:ines:0B:1:10GB
/collab/usr/gapps/wf:ALL:2.73GB:n/a:10GB
/collab/usr/gapps/wf:workflow:2.73GB:137238:10GB
/collab/usr/gapps/yorick:ALL:3.67GB:n/a:10GB
:
```

The Solution!

- MongoDB
  - Single Data Source
  - Access Control and Authorization
  - Easily accessible from multiple endpoints
- Python modules for inserting data into the database
- Web Interface
  - Quickly access and search the data

Under The Hood



The Future

- More advanced search and sort
- Usability enhancements
  - Rich tables
  - Human-readable data
- Integration into other tools
  - quotamod
  - Lorenz
- Rollout to all filers
  - Animal workspaces
  - RZ
- Collaborate with SAG and the Hotline

Special Thanks to our project mentors

Mike Gilbert

Stephanie Choate

Object Storage Investigation

(High Five's & Bash Scripts)

2019 Academy Project

Garrett Slone & Hoa Ngo

Mentors: Thomas Bennett, Rigo Moreno Delgado, Elsa Gonsiorowski

June 2019

LLNL-PRES-793037
This work was performed under the auspices of the U.S. Department of Energy by Lawrence Livermore National Laboratory under contract DE-AC52-07NA27344. Lawrence Livermore National Security, LLC

Lawrence Livermore National Laboratory

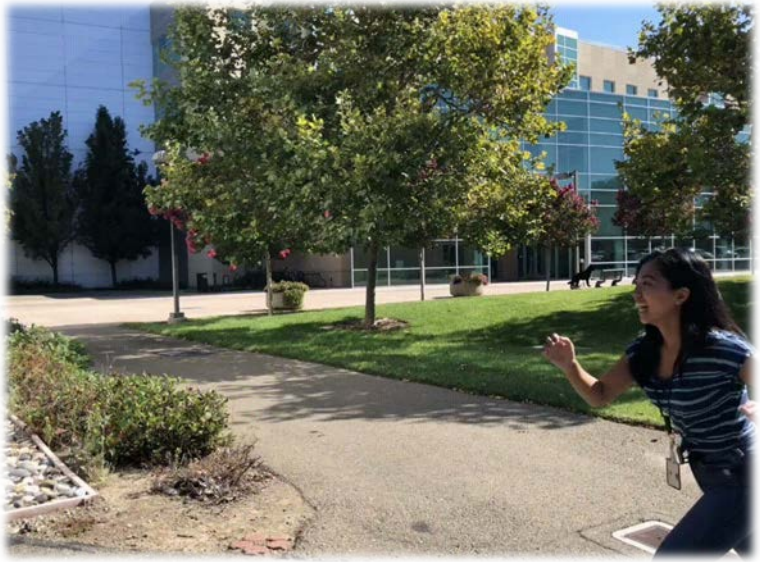
Overview

- Meet the Team
- What is all this buzz about Object Storage?
- Different Storage Architecture
- MinIO - Successes / Challenges
  - mc (MinIO Client)
- Ceph - Successes / Challenges
  - S3 API - python cmdline, s3cmd
- Wrap-Up

Object Storage Team

Garrett Slone

LAS POSITAS COLLEGE



./morninghighfives.sh

Hoa Ngo

CAL STATE EAST BAY

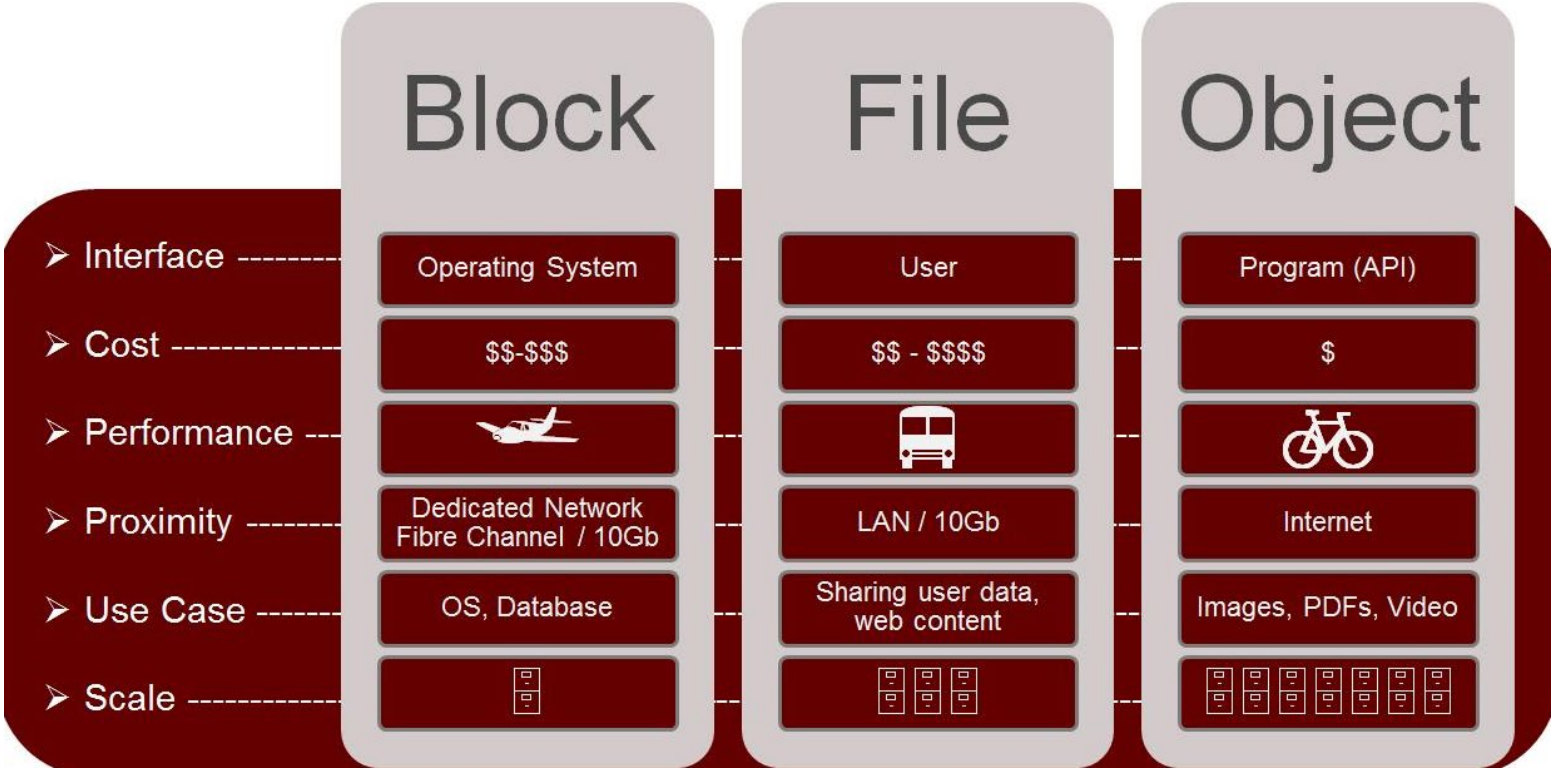
What is Object Storage?

- Alternative storage architecture

- 3 main components of objects:
  - Data
  - Metadata
  - Globally Unique Identifier

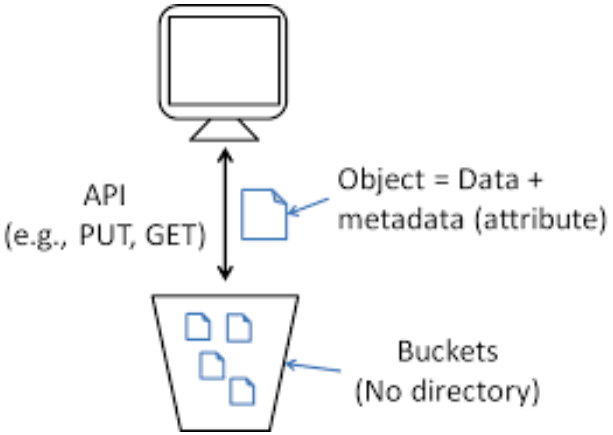
- Comparable to the system of valet parking

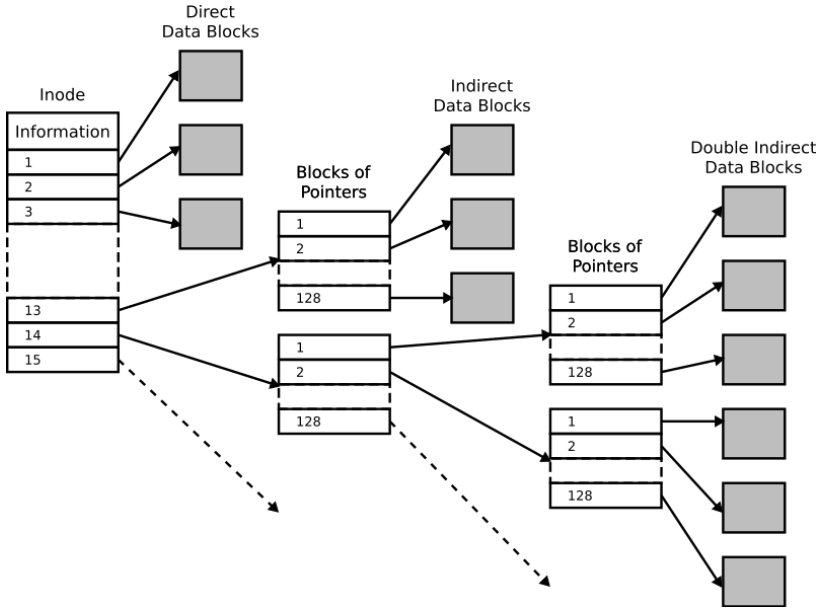
Differences (File, Block, Object)



Differences (cont.)

- Metadata





Minio

- Silicon-valley based tech startup (2014)
- Object storage server
- compatible with S3 interface
- MC = minIO client
  - minIO interface

Successes / Challenges

- Easy to install
- Comes w/ its own client interface:

  MC
- works well through GUI

- Server stopped if not ran as a service
- Security
- Full functionality cannot be access through CLI

```
[root@xenoni ~]# mc admin user disable team slone5
Disabled user `slone5` successfully.
[root@xenoni ~]# mc admin user set-policy team slone5 writeonly
Set a policy `writeonly` for user `slone5` successfully.
[root@xenoni ~]# mc admin user list team
enabled    ngo17                writeonly
disabled   slone5               writeonly
[root@xenoni ~]# su - slone5
Last login: Fri Jul 12 10:36:27 PDT 2019 on pts/1
[slone5@xenoni ~]$ mc ls slone5/slone5bucket
[2019-07-12 10:10:36 PDT]      0B it
[2019-07-11 15:46:33 PDT]      0B test.txt
[2019-07-09 16:12:40 PDT]    117B viminst.sh
```

Ceph – Red Hat Enterprise Object Storage

- *fun fact: first line of code ended up being part of Ceph written by Sage Weil @ a summer internship HERE at LLNL*
- Components:
  - Admin (ceph-admin)
  - Cluster Monitor (ceph-mon)
  - Object Storage OSDs (ceph-osds)
  - Rados Gateway (ceph-gateway)

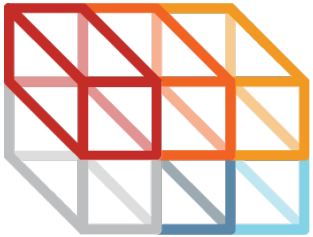
Successes / Challenges

- Fully installed
- Interfaced with S3 API
  - can also be interfaced with Swift API (have not yet been tested)
  - aws s3cmd
  - python

- Security
- Confusing installation
- Usage

Future Work

- OpenIO
- Triton
- Storage Grid



Joyent
TRITON™

StorageGRID®

openio
object storage
grid for apps

SPECIAL THANKS

David Fox
Thomas Bennett
Rigo Moreno Delgado
Elsa Gonsiorowski
Jason Shortino
Jean Shuler
Bryan Dixon

Sources

1) Introduction To Amazon Simple Storage Service (s3) - Cloud Storage on Aws - Create, Discover and Share Awesome Gifs on Gfycat
Anonymous - https://gfycat.com/illegalinsistentarmednylonshrimp
2) Object Storage Vs. Block Storage
Abhishek Ghosh-Abhishek Ghosh - https://thecustomizewindows.com/2017/09/object-storage-vs-block-storage/
3) https://upload.wikimedia.org/wikipedia/commons/4/4b/Object_Storage_Icon.png
4) Inode Pointer Structure
https://en.wikipedia.org/wiki/Inode_pointer_structure

Lawrence Livermore National Laboratory

Merlin Workflow Tools RabbitMQ and Redis

Sarah Mings
Zeke Morton
Eliana Neurohr

Mentors: Dave Fox, Jason Shortino

August 15, 2019

LLNL-PRES-XXXXXX
This work was performed under the auspices of the U.S. Department of Energy by Lawrence Livermore National Laboratory under contract DE-AC52-07NA27344. Lawrence Livermore National Security, LLC

Merlin Team

Sarah Mings
California State University, East Bay

Zeke Morton
University of California, Davis

Eliana Neurohr
Colorado College

Overview

- What is Merlin?
- Objectives
- What is RabbitMq, Celery, Redis?
- Puppet Manifest
- Docker Containers
- SSL Certificates
- Challenges
- What's Next?

What is Merlin?

- Open source workflow management tool for scientists to submit simulations to the HPCs
  - https://github.com/LLNL/merlin
- Our tools
  - Message brokers: RabbitMQ and Redis
  - Task queue: Celery
  - Configuration management tool: Puppet
  - Docker

Merlin

Objectives

- Install and test RabbitMQ, Redis, and Celery
- Puppetize the install of RabbitMQ and Redis
- Dockerize RabbitMQ and Redis
- Add security to RabbitMQ and Redis
  - Passwords and SSL certificates

What is RabbitMQ?

- Message broker that makes distributed systems development easy
- A message broker is to take incoming messages from applications and deliver to other applications

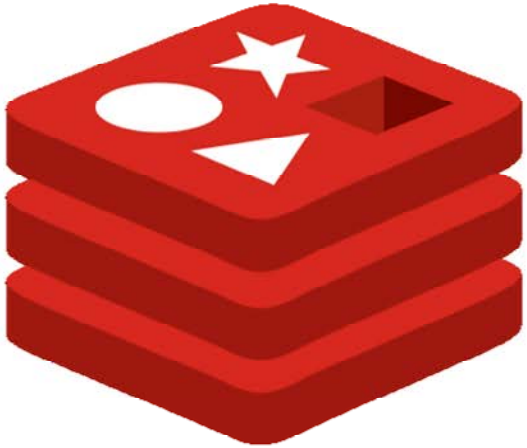
Testing RabbitMQ

```
[morton30@radon3 rabbit]$ python send.py
 [x] Sent 'Hello World!'
[morton30@radon3 rabbit]$ python rec.py
 [*] Waiting for messages. To exit press CTRL+C
 [x] Received 'Hello World!'
```

- Used the Pika Package in a virtual environment and a pip install

What is Redis?

- It's is an in-memory, key-value database, commonly referred to as a data structure server.
- Unlike simplistic key-value data stores that offer limited data structures, Redis has a vast variety of data structures to meet your application needs.



redis

https://en.wikipedia.org/wiki/Redis#/media/File:Redis_Logo.svg

What is Celery?

- It's a task queue with batteries included.
- Task queues let applications perform work, called tasks, asynchronously outside of a user request. If an app needs to execute work in the background, it adds tasks to task queues. The tasks are exec ces.

https://en.wikipedia.org/wiki/Celery_(software)#/media/File:Celery_logo.png

Install Celery & Test Celery

- $ pip install Celery
- Make task.py

```
from celery import Celery

BROKER_URL = 'amqp://Rabbit:passw0rd@localhost//Rabbit'

BACKEND_URL = 'redis://@localhost'

app = Celery('tasks', broker=BROKER_URL,
backend=BACKEND_URL)


@app.task

def add(x, y):

return x + y
```

puppet

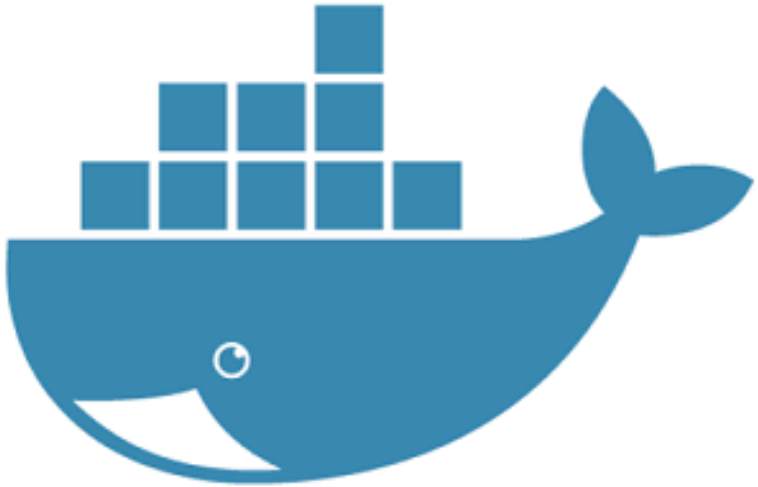
- Configuration management tool
- Best for downloading packages, placing files, and starting and enabling services
- `$ puppet resource <type> <item> >> manifest.pp`
- `$ puppet apply manifest.pp`

Puppet Manifest

```
package { 'Celery':
  ensure   => 'installed',
  provider => 'pip',
}
exec { 'certs':
  command => "sh ssl.sh",
  path => '/sbin:/bin:/usr/sbin:/usr/bin',
}
service { ['redis', 'rabbitmq-server']:
  ensure => running,
  enable => true,
}
file { '/etc/rabbitmq/rabbitmq.config':
  ensure   => 'file',
  group    => 0,
  mode     => '0777',
  owner    => 0,
  seltype  => 'usr_t',
  seluser  => 'unconfined_u',
  source   => '/tmp/rabbitmq.config',
}
```

Docker Containers

- Docker container is a standard unit of software that packages up code and all its dependencies, so the application runs quickly and reliably from one computing environment to another.



https://codeburst.io/basics-of-docker-c1416b02d03c

SSL Certificates Generation

- `tls-gen` is an open source tool originally used for RabbitMQ
- `tls-gen` generates a self-signed Certificate Authority (CA) certificate and two or more pairs of keys: client and server, all with a single command.
- Used basic profile that used a Elliptic Curve Cryptography(ECC) 256bit type
- https://github.com/michaelklishin/tls-gen

SSL Certificates RabbitMQ with Docker

- Used self sign certificates in environment variables
- Edit the docker-compose.yml

```
version: '3'

services:
 my-rabbit:
  hostname: my-rabbit
  image: rabbitmq:3
  ports:
    - 5671:5671
  environment:
    - SSL="true"
    - RABBITMQ_SSL_CERTFILE=/tmp/ssl/server_certificate.pem
    - RABBITMQ_SSL_KEYFILE=/tmp/ssl/server_key.pem
    - RABBITMQ_SSL_CACERTFILE=/tmp/ssl/ca_certificate.pem
    - RABBITMQ_DEFAULT_USER=Rabbit
    - RABBITMQ_DEFAULT_PASS=passw0rd
    - RABBITMQ_DEFAULT_VHOST=/Rabbit
  volumes:
    - /tmp/ssl:/tmp/ssl
```

Password for Redis

- Set up password in Redis configuration file
- Only can set up ONE password!
- Merlin team found work around by encrypting all data

```
version: '3'

services:
 some-redis:
  image: redis
  command: redis-server --requirepass foobared
  ports:
    - '6379:6379'
```

SSL Certificates RabbitMQ

- Used self sign certificates from RabbitMQ documentation
- Edit /etc/rabbitmq/rabbitmq.config

```
{ssl_listeners, [5671]},
```

```
{ssl_options, [{cacertfile,           "/tmp/ssl/ca_certificate.pem"},
               {certfile,             "/tmp/ssl/server_certificate.pem"},
               {keyfile,              "/tmp/ssl/server_key.pem"},
               {verify,               verify_none},
              {fail_if_no_peer_cert, false}]}
```

Challenges

- RabbitMQ Manual Install
- Managing all the software dependencies
- Puppet Manifest
- Add security
- SE Linux

What's Next

- Possible security enhancements for Redis
- Integration and testing it with Merlin
- Testing with other Linux distributions

Special Thanks

Dave

Jason

Thomas

Rigo

Bryan

ean

Thanks!