Exploring a new way to manage systems with ostree and atomic

Michael Scherer, misc@redhat.com

Who am I ?

Sysadmin @

In a previous episode..

Packager/sysadmin/build system maintainer @mageia

Python coder @ some firewall startup

Counter stri^W Sysadmin

@ unamed european institution

Old ways

Assemble set of packages

Binary packages

Sources

Both ways

Base + ports

Appliances..

.. or phones/tablets

A new hop^W way

Project Atomic

Codename for a set of distribution

Centos/Fedora

Others are welcome

Atomic upgrade

Containers

Lightweight

Base of openshift v3

Let's get technical

Components, bottom To top

Ostree

Created for
testing GNOME

Firmware-like

Bootable read only filesystem

All in /usr +

some link to /var

Reboot on new version

Reboot on old version

Make rollback possible

Git for filesystem

Docker & systemd

Ship applications with runtime

API to start, monitor, control services

Cgroups usage to limit containers

Geard / Kubernetes

Orchestrate containers

Manage network

Distribute tasks

Ensure redundancy

Various others things

SELinux

Isolation

Protect base system from containers

Protect containers from containers

Cherry on the cake

Cockpit as a web interface

Why ?

The stars aligned right

Ostree from Gnome QA

Jails

=> Openvz

=> Linux namespace

=> LXC

=> Docker

SELinux / svirt

Systemd

Pressure on IT departments

Automate more

Drive human costs down

Colocate services more

Drive hardware spending down

Deliver faster

Drive profits high

Agility

Dev Ops

Dev systems !=

Prod servers

Example

Gluster.org web infra

Need to be redone

Static website

Need a httpd server

Need a builder

Need a wiki

Mediawiki

MySQL || PostgreSQL

Varnish

Awstats

Traditional way

All on 1 server

VM way

1 VM per service

Containers way

Everything in a different containers

Microservices

Why ?

A few problems

Static website build

Use middleman

Ruby

No comment

Web designer use F20

So better use this as well

Unpackaged software

Mediawiki

Need specific isolation

Test upgrade
and rollback

Scaling issues

Mediawiki write on the FS

Use S3 backed storage

Swift container

Need a new version of something ?

Varnish 4.0 ?

(not a good example)

A few constraints

Treat container as cattle

Throw away instances

Intra container communication

Data container pattern

Immutable infrastructure pattern

Must take scalability in account

(May be complex to do)

Kubernetes / etcd

MySQL replication

Swift replication

Deployment workflow

Install a few atomic host

All are clients, one is master

Depend on services running

Custom containers

Need to have a private registry

Docker registry

Alternatives :

Pulp + crane

Can be run in a container

Upload of definition to kubernetes

```
$ kubectl create -f gluster.json
```

Exact syntax as exercise to the audience

Management of containers build

Layered containers

Select the base

Add runtime

Add custom code

Using STI
(Source to Image)

Workflow of containers update

Testing and QA

Lot of questions

are still open

Configuration management ?

Backup ?

Monitoring ?

Log centralisation ?

Fine grained management ?

Network management ?

http://projectatomic.io/

?