ANSIBLE ALL THE THINGS

From traditional to unorthodox, Ansible for Everything

Adam Miller
Principal Software Engineer
AnsibleFest SF 2017

redhat.

AGENDA

AGENDA

WHAT WE'RE GOING TO TALK ABOUT TODAY

- Why on earth would I want to do all the things with Ansible?
- Automation Tool
- Configuration Management
- Provisioning and Systems Management
- Deployment
- Application Lifecycle Management
- Orchestration
- Command Line Tooling
- Event Based Execution
- Workflow Automation
- CI/CD
- Ansible Container
- Ansible Tower

WHAT IS ANSIBLE?

redhat.

QUICK INTRODUCTION

WAIT, YOU DON'T KNOW WHAT ANSIBLE IS?

Ansible is an automation tool

- Ansible is a simple agentless idempotent **task automation tool**
  - By default, tasks are executed in-order but we can change that if we want.
- **Tasks** are performed via **modules**
- **Tasks** are grouped together via **plays**
  - Also via **roles**, which are reusable sets of plays we can pass variables to
  - A **play** operates on a set of hosts
- **Playbooks** can contain one or many **plays**
  - Can be used with "traditional" configuration management systems
    - There's even a puppet module!

ANSIBLE EVERYTHING

USING ANSIBLE FOR EVERYTHING

WHY WOULD I WANT TO DO THAT?

Ansible is a simple automation tool that can:

- Execute tasks on one or many hosts
- Orchestrate an otherwise complex order of operations, even conditionally based on system facts or variables provided at runtime.
- Custom modules can be written in any programming language with JSON support

Question of the day:

What are you trying to accomplish that could be automated?

redhat.

USING ANSIBLE FOR EVERYTHING

ANSIBLE ALL THE THINGS!!!!

What are you trying to do?

- Configuration Management?
- Provision Virtual Machines or IaaS instances?
- Test software?
- Automate workflows?
- Continuous Integration / Continuous Deployment?
- Configure hardware switches, routers, and load balancers?
- Replace terrible shell scripts that have survived too long already?
- Other?

ANSIBLE CAN DO ALL OF THAT! (AND MUCH MORE)

ANSIBLE DOES THAT

CONFIGURATION MANAGEMENT

KEEPING THE TRAIN ON THE TRACKS

What is configuration management?

> Systems engineering process for establishing and maintaining consistency of a product's performance, functional, and physical attributes with its requirements, design, and operational information throughout its life.

Generally boils down to:

- Managing file content
- Configuration Templating
- System and Service state
- Package Management
- Lifecycle Management

ANSIBLE DOES THAT

OMG, NO WAY?!?!?!

- Service state: `service` module
- Files and configuration modules: `acl archive assemble blockinfile copy fetch file find ini_file iso_extract lineinfile patch replace stat synchronize tempfile template unarchive xattr`
- System state modules: `aix_inittab alternatives at authorized_key beadm capabilities cron cronvar crypttab debconf facter filesystem firewalld gconftool2 getent gluster_volume group hostname iptables java_cert kernel_blacklist known_hosts locale_gen lvg lvol make modprobe mount ohai open_iscsi openwrt_init osx_defaults pam_limits pamd parted ping puppet runit seboolean sefcontext selinux selinux_permissive seport service setup solaris_zone svc sysctl systemd timezone ufw user`
- Package Management modules: `bower bundler composer cpanm easy_install gem maven_artifact npm pear pip apk apt apt_key apt_repository apt_rpm dnf dpkg_selections homebrew homebrew_cask homebrew_tap layman macports openbsd_pkg opkg package pacman pkg5 pkg5_publisher pkgin pkgng pkgutil portage portinstall pulp_repo redhat_subscription rhn_channel rhn_register rpm_key slackpkg sorcery svr4pkg swdepot swupd urpmi xbps yum yum_repository zypper zypper_repository`

More modules being added all the time...

redhat.

ADVANCED CONFIGURATION MANAGEMENT

THAT LITTLE EXTRA

The following categories of Infrastructure Needs are covered extensively by Ansible modules:

- Clustering
- Commands
- Crypto
- Database
- Files
- Identity
- Inventory
- Messaging
- Monitoring
- Network
- Notification
- Packaging
- Remote Management
- Source Control
- Storage
- System
- Utilities
- Web Infrastructure

redhat.

PROVISIONING

MAKING SOMETHING FROM NOTHING

What do you want to accomplish?

- Create IaaS compute instances, object stores, or ephemeral resources?
- Provision virtual machines?
- Create storage allocations?
- Set firewall rules?
- Configure highly available load balancers?
- Create VLANs?
- Deploy container orchestration resources?
- Create databases?
- Other?

ANSIBLE CAN DO THAT

WHAT? AGAIN? NO WAY!!

Provisioning support for many IaaS providers:

- Amazon Web Services
- Apache CloudStack
- Centurylink Cloud
- Digital Ocean
- DimensionData
- Google Cloud
- Linode
- Microsoft Azure
- OpenStack
- Rackspace Public Cloud
- Softlayer Webfaction

Datacenter and Virtualization:

- oVirt / RHV
- libvirt resource management
- Joyent SmartOS Virt
- VMWare (VSphere/ESXi)

Storage:

- AIX LVM
- Gluster Volume
- Infinidat
- LVM2
- NetApp
- ZFS

redhat.

PROVISIONING - CONTINUED

OMG, THIS LIST JUST KEEPS GOING...

Networking

- A10 Networks
- Apstra AOS
- Arista EOS
- Avi Networks
- BigSwitch
- Cisco (ASA, IOS/IOS-XR, and NX-OS)
- Cumulus Networks (Cumulus Linux)
- Dell EMC (OS6, OS9, and OS10)
- F5 BigIP
- Fortios Firewall
- JunOS
- Lenovo CNOS
- Netvisor
- Open vSwitch
- Palo Alto Networks PAN-OS
- Nokia SR OS
- VyOS

Databases

- InfluxDB
- Redis
- Riak
- MS-SQL
- MySQL
- Postgresql
- Vertica

redhat.

PROVISIONING - CONTINUED

SERIOUSLY? MORE STUFF?

Web Infrastructure and Clustering

- Apache HTTPD (module and mod_proxy management)
- Consul
- Django Management
- eJabberd
- htpasswd
- JBoss
- Jenkins (Jobs, Plugin, and Jenkinsfile management)
- Jira
- Kubernetes
- Letsencrypt
- Pacemaker
- Supervisord
- ZooKeeper

redhat.

DOING THINGS WITH ANSIBLE

DEPLOYMENT

I JUST GIT PUSH TO THE CLOUD, RIGHT?

Software Deployment is the act of making software available on systems; most often, this is a sequence of steps that must be performed in-order. (**In-order task execution anyone**?)

Example:

- Sync some data
- Database schema migration
- Remove systems from load balancer
- Push new code
- Put systems back in load balancer
    - Rinse/Repeat on previously not upgraded set
- Verify services are functional
- Status update

Remember what a Playbook does?

APPLICATION LIFECYCLE MANAGEMENT

DO IT LIVE!

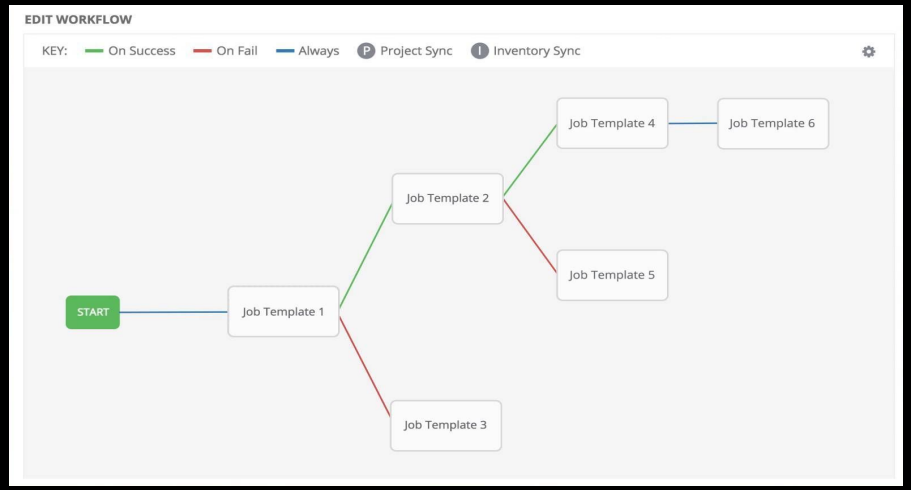
Managing application lifecycle across one or many hosts

- Ansible can orchestrate both simple and complex lifecycle management
- Lifecycle “order of operations” defined in Playbooks
  - Whatever your requirements are
- Plays can execute on different sets of hosts
  - Multiple plays per playbook
- Plays can use varying execution strategies for various requirements
  - Cluster node management
  - Database schema updates
  - etc
- Sky is the limit
  - (something something … cloud)

ORCHESTRATION AND WORKFLOW

AUTOMATION WITH FEELING

Flow controlled automation by data from the environment allowing the automation tasks to make “intelligent” decisions.



COMMAND LINE TOOLING

BUT WHAT ABOUT MY PERL ONE-LINERS?

Make Ansible your new command line tooling API, stop re-inventing the wheel

- Ansible provides a very capable Python API for modules
- Modules can be written in any programming language that understands JSON
- Provides a consistent “UX” for all tasks
- Gives you and your ops team an “on ramp” to scaling your tasks across the infrastructure

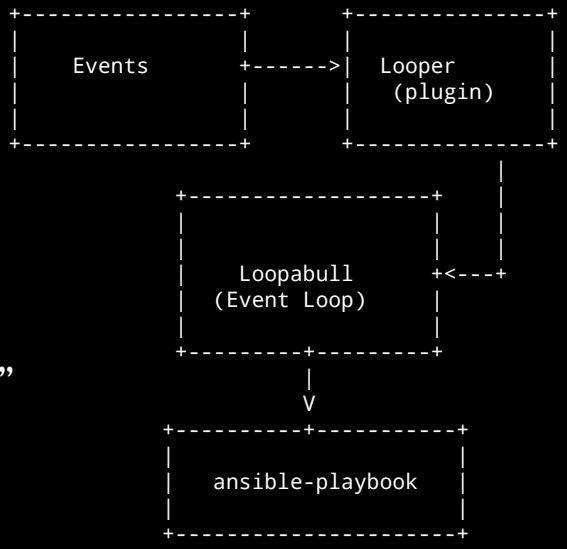
```
$ ansible localhost -m my_task -a “arg1=foo arg2=bar”
```

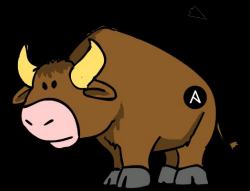
EVENT BASED EXECUTION

COWSAY WHAT?

Ansible can easily integrate with existing infrastructure to perform actions based on events.

- Example: `loopabull`
  - Events in the infrastructure spawn messages on the bus
  - `loopabull` listens on the bus, waiting for a “routing key” that it cares about (message topic)
  - Message payload is injected into Ansible playbooks as variables, allowing for decisions to be made based on message contents





redhat.

CONTINUOUS INTEGRATION

THERE IS ONLY ZUUL ... (BUT ALSO OTHER STUFF)

Brief story of OpenStack Zuul and Jenkins Job Builder

- OpenStack CI System (Zuul) - http://status.openstack.org/zuul/
  - 2,000+ jobs-per-hour
    - single-use OpenStack VMs -> create and destroy 2K+ VMs per hour
  - ~1750 disjoint git repositories to perform gating on
  - Spread across 7 public OpenStack clouds and 4 private OpenStack clouds
    - Hybrid cloud anyone?
- OpenStack wanted to not fiddle with XML for Jenkins Jobs
- Jenkins Job Builder (YAML) was created
- Jenkins Performance issues ran into...
- No more Jenkins, automatically convert JJB YAML into Ansible Playbooks
- Future: Migrate entirely away from JJB, make it all Ansible!

MORE CONTINUOUS INTEGRATION

THE OTHER STUFF

Fedora Taskotron - https://taskotron.fedoraproject.org/

- CI for the entire Fedora Linux Distribution
- “Tasks” definitions originally in YAML
- Tasks for every RPM, ISO, VM Image, Container, etc in the distro
- Automated reporting to the Fedora Updates System (Bodhi)
- Migration from Taskotron YAML to Ansible Playbooks

ANSIBLE CONTAINER

END THE DOCKERFILE MADNESS

Using Ansible playbooks to build you container images

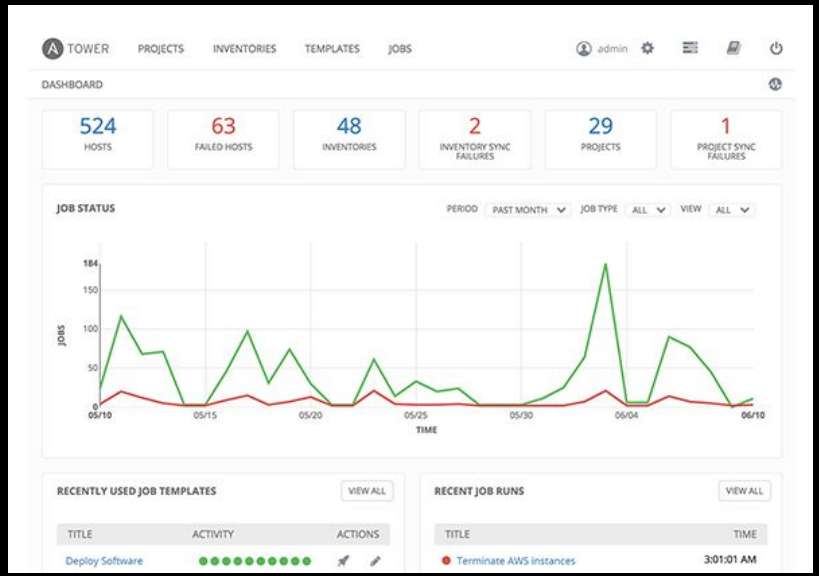
- Stop chaining together shell commands in Dockerfiles
- Create containers the same way you deploy to servers
- **roles** == services, build your containers using **roles**
  - Making single-purpose (microservice) containers easy
- Create multi-container builds easily
  - (Think Docker Compose, but like … better)
- Deploy to Container Orchestration Platforms
  - Currently Supports OpenShift and Kubernetes

ANSIBLE TOWER

PRETTY GRAPHS!

The definitive Ansible Centralized Management Portal

- Role Based Access Control
- Centralized Logging, History Visualizations
- Multi-Playbook Workflow Orchestration
- Playbook and System Auditing (System Tracking)
- Self-Service Automation
  - Sanitized form-based playbook runs
- Integrated Notifications (ChatOps, etc)
- REST API
- ... and much much more!



redhat.

THANK YOU

ADAM MILLER

 maxamillion

 maxamillion

 @TheMaxamillion