AUTOMATE ABSOLUTELY EVERYTHING

From traditional to unorthodox, Automate Everything
(With Ansible)

Adam Miller
Senior Principal Software Engineer
Ansible Engineering
AN EPITOME OF AUTOMATION
(Powered by Ansible)
WHAT IS ANSIBLE?
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**
- Plays nicely with "traditional" configuration management systems
  - There's even a puppet module!
QUICK INTRODUCTION
WAIT, YOU DON’T KNOW WHAT ANSIBLE IS?

Module: yum
Arguments: name=bash state=installed

$ ansible localhost -m yum -a "name=bash state=installed"
localhost | SUCCESS => {
"changed": false,
"msg": "Nothing to do"
}
QUICK INTRODUCTION

WAIT, YOU DON'T KNOW WHAT ANSIBLE IS?

**Module:** yum

**Arguments:** name=bash state=installed

```yaml
playbook.yml
---
- hosts: localhost
  tasks:
    - yum:
        name: bash
        state: installed
```
AUTOMATE EVERYTHING
(With Ansible)
USING ANSIBLE FOR EVERYTHING
WHY WOULD I WANT TO DO THAT?

Ansible is a simple automation tool that can:

- Execute tasks against one or many hosts, systems, clouds, REST APIs, network devices, hardware devices/appliances, SANs, etc. (basically anything)
- Orchestrate an otherwise complex order of operations, even conditionally based on “facts” or variables (optionally 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?
What are you trying to do?

- Configuration Management?
- Provision VMs, Containers, IaaS Instances, K8s Resources?
- Test software?
- Automate workflows?
- Continuous Integration / Continuous Deployment?
- Configure hardware switches, routers, firewalls, 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
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/Configuration:** acl archive assemble blockinfile copy fetch file find ini_file iso_extract lineinfile patch read_csv replace stat synchronize tempfile template unarchive xattr xml
- **System State:** aix_devices aix_filesystem aix_inittab aix_lvg aix_lvol alternatives at authorized_key awall beadm capabilities cron cronvar crypttab dconf deconf facter filesystem firewalld gather_facts gconf tool2 getent group hostname interfaces_file iptables java_cert java_keystore kernel_blacklist known_hosts locale_gen lvg lvol make mksysb modprobe mount nosh ohai open_iscsi openwrt init osx_defaults pamd pam_limits parted pids ping puppet python_requirements_facts reboot runit seboolean sefcontext selinux_permissive selinux selogin seport service_facts service setup solaris_zone svc sysctl systemd sysvinit timezone ufw user vdo xfconf xfs_quota
- **Package Management:** apk apt apt_key apt_repo apt_repository apt_rpm bower bundler composer cpanm dnf dpkg_selections easy_install flatpak flatpak_remote gem homebrew homebrew_cask homebrew_tap installp layman macports maven_artifact npm openbsd_pkg opkg package package_facts pacman pear pip pip_package_info pkg5 pkg5_publisher pkgin pkgng pkgutil portage portinstall pulp_repo redhat_subscription rhn_channel rhn_register rhsm_release rhsm_repository rpm_key slackpkg snap sorcery svr4pkg swdepot swupd urpmi xbps yarn yum yum_repository zypper zypper_repository

More modules being added all the time...
ANSIBLE DOES THAT - Windows Edition
DON’T WORRY WINDOWS FOLKS, WE DIDN’T FORGET ABOUT YOU <3

- Windows Modules:
  - async_status
  - setup
  - slurp
  - win_acl
  - inherit
  - aud
  - policy
  - audit
  - rule
  - certificate
  - store
  - chocolate
  - config
  - facts
  - chocolate
  - feature
  - chocolate
  - source
  - command
  - copy
  - credential
  - defrag
  - disk
  - facts
  - disk
  - image
  - dns
  - client
  - record
  - domain
  - computer
  - domain
  - controller
  - domain
  - group
  - membership
  - domain
  - user
  - dotnet
  - ngen
  - dsc
  - win_environment
  - eventlog
  - entry
  - eventlog
  - feature
  - file
  - file
  - version
  - find
  - firewall
  - firewall
  - rule
  - format
  - get
  - url
  - group
  - membership
  - group
  - hostname
  - hosts
  - hotfix
  - http
  - proxy
  - iis
  - virtualdirectory
  - webapplication
  - webapp
  - webapping
  - iis
  - webbinding
  - iis
  - website
  - inet
  - proxy
  - line
  - in
  - file
  - mapped
  - drive
  - msg
  - win
  - msi
  - nssm
  - optional
  - feature
  - owner
  - package
  - page
  - file
  - partition
  - path
  - pester
  - ping
  - power
  - plan
  - product
  - facts
  - psexec
  - psmodule
  - psrepository
  - rabbitmq
  - plugin
  - rds
  - cap
  - rds
  - rap
  - rds
  - settings
  - reboot
  - regedit
  - region
  - reg
  - merge
  - reg
  - stat
  - robocopy
  - route
  - say
  - scheduled
  - task
  - scheduled
  - task
  - stat
  - security
  - policy
  - service
  - share
  - shell
  - shortcut
  - snmp
  - stat
  - temp
  - file
  - template
  - timezone
  - toast
  - unzip
  - updates
  - uri
  - user
  - profile
  - user
  - user
  - right
  - wait
  - for
  - process
  - wait
  - for
  - wakeonlan
  - webpicmd
  - whoami
  - xml

More Windows modules being added all the time too!
The following categories of Infrastructure Needs are covered extensively by Ansible modules:

- Cloud
- Clustering
- Commands
- Crypto
- Database
- Files
- Identity
- Inventory
- Messaging
- Monitoring
- Network

- Notification
- Packaging
- Remote Management
- SecOps
- Source Control
- Storage
- System
- Utilities
- Web Infrastructure
- Windows
Over 3300 modules and plugins as of Ansible v2.8.0
(not to mention all that sweet Ansible Galaxy Content)
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
- Cloudscale
- Digital Ocean
- DimensionData
- Docker Swarm
- Google Cloud
- Linode
- Microsoft Azure
- OpenStack
- OVH
- Packet

... and Datacenter Virtualization
- Profitbricks
- Rackspace Public Cloud
- Softlayer
- WebFaction
- Atomic Host
- libvirt resource management
- Joyent SmartOS Virt
- oVirt
- Red Hat Virtualization
- VMWare (VSphere/ESXi)
- Univention
PROVISIONING - CONTINUED
OMG, THIS LIST JUST KEEPS GOING…

Networking
- A10 Networks
- Apstra AOS
- Arista EOS and Cloudvision
- Aruba
- Avi Networks
- BigSwitch
- Cisco (ASA, ACI, IOS/IOS-XR, NX-OS, WLC)
- Cumulus Networks (Cumulus Linux)
- Dell EMC (OS6, OS9, and OS10)
- F5 BigIP
- Fortios Firewall
- Huawei Cloudengine
- JunOS
- Lenovo CNOS

Databases
- Netscaler
- Netvisor
- Open vSwitch
- Ordnance
- Palo Alto Networks PAN-OS
- Nokia SR OS
- VyOS
- InfluxDB
- Redis
- Riak
- MS-SQL
- MySQL
- Postgresql
- Vertica
Infrastructure, Web, Clustering and Cloud

- Apache HTTPD (module and mod_proxy management)
- Atomic Host
- Consul
- Django Management
- eJabberd
- htpasswd
- HP iLO
- JBoss
- Jenkins (Jobs, Plugin, and Jenkinsfile management)
- Jira
- Kubernetes
- Letsencrypt
- ManageIQ

Storage

- OpenShift
- Pacemaker
- Supervisord
- ZooKeeper
- AIX LVM
- Gluster Volume
- Infinidat
- LVM2
- NetApp
- Purestorage
- ZFS
DOING THINGS WITH ANSIBLE
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”
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)
Flow controlled automation by data from the environment allowing the automation tasks to make “intelligent” decisions.
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
CONTINUOUS INTEGRATION
THERE IS ONLY ZUUL

Zuul CI - https://zuul-ci.org/

- Project Gating
- Workflow pipelines
- Cross-Project and Cross-Repo dependency gating
- Multi-node deployment job testing
- Tests are written as Ansible Playbooks
- Run your tests with the same Playbooks you use in Production
- 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
  - >1800 disjoint git repositories to perform gating on
  - Spread across 7 public OpenStack clouds and 4 private OpenStack clouds
    - Hybrid cloud anyone?
Using Ansible playbooks to build you OCI compliant 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
- Get more out of your time investment writing roles and playbooks
- Create multi-container builds easily (Ansible Container)
  - (Think Docker Compose, but like ... better)
- Deploy to Container Orchestration Platforms (Ansible Container)
  - Currently Supports OpenShift and Kubernetes
Automating OpenShift and Kubernetes

- openshift and k8s modules
  - Automate command/control of OpenShift or Kubernetes native resources
- Automation Broker
  - Orchestrate/Manage Service Catalog Apps
  - http://automationbroker.io/
  - (AKA - Ansible Service Broker)
- Ansible Operators
  - Create Kubernetes Operators with Ansible
  - Operator Framework runs an Ansible playbook or role every time a certain type of object is detected / modified
TESTING YOUR PLAYBOOKS AND MODULES
TEST DRIVEN WHAT???

Molecule (https://molecule.readthedocs.io)

● Designed to aid in the development and testing of Ansible roles.
● Provides support for testing with multiple:
  ○ Instances
  ○ operating systems / Linux distributions
  ○ virtualization providers
  ○ test frameworks
  ○ testing scenarios
● Molecule uses Ansible playbooks to exercise the role and its associated tests.
● Molecule supports any provider that Ansible supports.
Ansible Galaxy

- Galaxy is your hub for finding, reusing and sharing the best Ansible content
- Share Roles you’ve created
- Collaborate upstream on improving re-usable functionality
- https://galaxy.ansible.com
USING ANSIBLE TO HARDEN SYSTEMS
ONE DOES NOT SIMPLY WALK INTO MORDOR

Security Audit and Compliance

- DISA Security Technical Implementation Guides (STIGs)
- Ansible Lockdown - (in partnership with MindPointGroup)
  - Available in Ansible Galaxy
  - DISA and CIS Guidelines

- OpenSCAP
  - SCAP Implementation
  - Ansible Integrations for remediation
ANSIBLE TOWER
PRETTY GRAPHS! (But no seriously, this is awesome)

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)
- Clustered Auto-Sharding Job Execution
- Self Documenting REST API
- Tower CLI and Ansible Engine Tower modules
- ... and much much more!
ANSIBLE AS AN UNIVERSAL LANGUAGE
ANSIBLE FOR EVERYTHING
THANK YOU