Ansible

the easiest automation, ever.

What is Ansible?

- **Ansible is a radically simple IT automation engine that automates cloud provisioning, configuration management, application deployment, intra-service orchestration, and many other IT needs.**

Manage Infrastructure

- **Easiest Automation Language**
- **Scalable Ad-Hoc Parallel Execution**
- **Multi-tier Orchestration**
- **Ansible models your IT infrastructure by describing how all of your systems inter-relate, rather than just managing one system at a time.**

Ansible USP's

- **Simple, Easy**
- **Minimal requirements**
- **Ready for various clouds & Docker**
- **Secure, only SSH**
- **Audit-able**
- **Role-based access in paid version**

Ansible vs. Puppet

- **No agents**
- **No master**
- **No database**
- **No Ruby programming**
- **No network & firewall changes**
- **No new public key infrastructure**

Concept: Inventory

- **All machines in infrastructure/cloud**
- **Arbitrary grouping**
- **Dynamic inventory**
- **Can be used for staging**

Concept: Playbook

```
- name: webapp server setup
  hosts: webservers
  sudo: yes

  pre_tasks:
    - include: webservers/pre_tasks.yml

  roles:
    - common
    - geerlingguy.java
    - hostclick.tomcat

  post_tasks:
    - include: webservers/post_tasks.yml
```

Concept: Actions

```
- name: 'fix JAVA_HOME'
  shell: echo export JAVA_HOME=/usr/lib/jvm/jre > /etc/profile.d/jre.sh

- name: 'fix permissions'
  file: dest=/etc/profile.d/jre.sh mode=0755
```

```
- name: 'fix "web" in firewall configuration'
  lineinfile: dest=/etc/sysconfig/iptables
              line='-A INPUT -p tcp -m tcp --dport 8080 -j ACCEPT'
              state=present
              insertbefore=^COMMIT

- name: 'restart iptables'
  service: name=iptables state=restarted
```

Jinja2

- **variable: value**
- **my_config = {{ variable }}**
- **{% if 'webservers' in group_names %}**
- **{% for user in users %}**

Concept: Module

- **cloud**
- **clustering**
- **commands**
- **database**
- **files**
- **inventory**
- **messaging**
- **monitoring**
- **network**
- **notification**
- **packaging**
- **source control**
- **system**
- **utilities**
- **web infrastructure**
- **Windows**

Concept: Role

- **Roles are ways of automatically loading certain vars_files, tasks, and handlers based on a known file structure.**

- **Grouping content by roles also allows easy sharing of roles with other users.**

- **http://galaxy.ansible.com**

Concept: Idempotency

- **Speed reduces when changing blindly**
- **System life cycle as transaction log**
- **No uncertainty: describe desired state**
- **Ensure no changes unless things change**
- **Tripwire real changes for audit**

Best Practice

- **Don’t solve everything with Ansible**
- **Package your deployables**
- **Provide simple wrappers**
- **Create smoketests**
- **Use tags to test and verify**
- **Long running tasks should run in screen**
- **Validate proper ordering**

DRY

- **Avoid duplication**
- **Re-use Galaxy roles**
- **Leverage Jinja2 templates**

Playbooks

- **Make it readable**
- **Keep it simple**
- **Always provide a task name**
- **Always define state**
- **Over-use comments and white-space**

Tag all the things

- **Tags help organization**
- **Tags can help in testing**
- **—skip-tags=tags,to,skip**
- **—tags=only,run,these,tags**

Demo

- **https://github.com/bbaassssiiee/lunchbox**