

Ansible basics



Main features

- Agentless
- Secure
- Provisioning AND deployment
- Easy to start
- Data driven
- Idempotent

Static inventory

- Static – ini file

```
[web]
# multiple hosts
web[1:2].davidkarban.cz

[mariadb]
mariadb.davidkarban.cz ansible_ssh_host=185.8.164.70

[loadbalancer]
haproxy.davidkarban.cz

[monitoring]
icinga-master.karban.eu

[backups]
duplicitybackup.davidkarban.cz

[slack:children]
web
```

Dynamic inventory

- Get inventory from cloud services.
- Aws, Digital Ocean, Rackspace, Docker, ...
- Easy to create own for your infrastructure.
- Automatically create group by tags, security groups, ...

Facts

```
$ ansible -m setup localhost
localhost | success >> {
  "ansible_facts": {
    "ansible_all_ipv4_addresses": [
      "10.13.3.193",
      "192.168.122.1",
      "10.8.0.16"
    ],
    "ansible_all_ipv6_addresses": [
      "fe80::4eeb:42ff:fe7a:21b9",
      "fe80::8cb4:6eff:fed6:5743"
    ],
    "ansible_architecture": "x86_64",
    "ansible_bios_date": "01/09/2012",
    ...
  }
}
```

Playbook

```
# gather all facts
- include: common_scripts/gather_facts.yml
# useful includes
- include: common_scripts/common.yml
- include: common_scripts/ntp.yml

- hosts: webservers
  pre_tasks:
    - name: Ensure www user exists
      become: yes
      become_user: root
      user: name=www state=present group=www-data home=/dev/null shell=/bin/false
  roles:
    - apache
    - php
    - mysql
  tasks:
    - name: Additional action not in roles
      authorized_keys:
        user: www
        key: "{{ lookup('file', inventory_dir + '/public_keys/www.pub') }}"
        state: present
```

Modules

- “batteries included”
- 250+ modules
- Cloud services (Amazon, Rackspace, Google Compute Platform, ...)
- Packaging (apt, yum, pip, gem, ...)
- Source control (git, svn, ...)
- OS stuff (service, command, file, template, ...)

Variables

- group_vars
- host_vars
- inventory
- facts, local facts
- registered variables
- “magic” hostvars variable

Tasks

- Basic work unit, one action.

```
- name: Get current time
tags:
  - Debian
  - RedHat
run_once: true
always_run: true
changed_when: false
command: date +%Y-%m-%dT%H:%M:%S+02:00
register: current_time

- name: Create configuration file on monitoring servers
tags: nrpe
template: src=host.cfg.j2 dest=/etc/
{{ nrpe_nagios_provider }}/ansible-managed/{{ nrpe_fqdn + ".cfg"}}
owner=root group=root mode=0644
delegate_to: "{{ item }}"
with_items: nagios_servers
notify: restart nagios
```

Handlers

- Special task, runs only if there was change.
- Usable for service restarts, database upload, ...

```
tasks:
```

```
- name: Load apache2 configuration
```

```
  template: src=apache2.conf.j2 dest=/etc/apache2/apache2.conf
```

```
  notify: restart apache2
```

```
# handler will run only if apache2.conf is changed
```

```
handlers:
```

```
- name: restart apache2
```

```
  service: name=apache2 state=restarted
```

Templates

- Jinja2 templating engine

```
listen stats
  mode http
  option httplog
  stats enable
  stats auth admin:{{ item }}
  stats refresh 20s
  stats uri /
  bind {{ ansible_default_ipv4.address }}:8080
```

```
backend app
  balance roundrobin
```

```
{% for host in groups.web %}
  server {{ host }} {{ hostvars[host]['ansible_default_ipv4']['address'] }}:80 check
{% endfor %}
```

Includes

- Two level of includes
 - Playbook wide

```
# Basic settings
- include: common_scripts/common.yml
- include: common_scripts/ntp.yml

hosts: web
```

- Task level

```
tasks:
- include: setup_environment var1=true var2="/etc/hosts"
```

Roles

- Includes on steroids, encapsulate set of tasks, variables, templates, files together.

```
~/ansible/roles/haproxy$ tree
```

```
.
├── defaults
│   └── main.yml
├── files
│   └── haproxy
├── handlers
│   └── main.yml
├── meta
│   └── main.yml
├── tasks
│   └── main.yml
└── templates
    └── haproxy.cfg.j2
```

Tools

- Vault (ansible-vault cmdline app)
Encrypt and deencrypt variable files.
- Galaxy (<https://galaxy.ansible.com>)
Web based hub with free to use roles from community.
- Tower
Enterprise web interface with Role based access control, periodic jobs, logs, dashboards and more.

Questions?

David Karban

david@karban.eu - www.karban.eu

Training! - www.ansible.cz

