

ANSIBLE

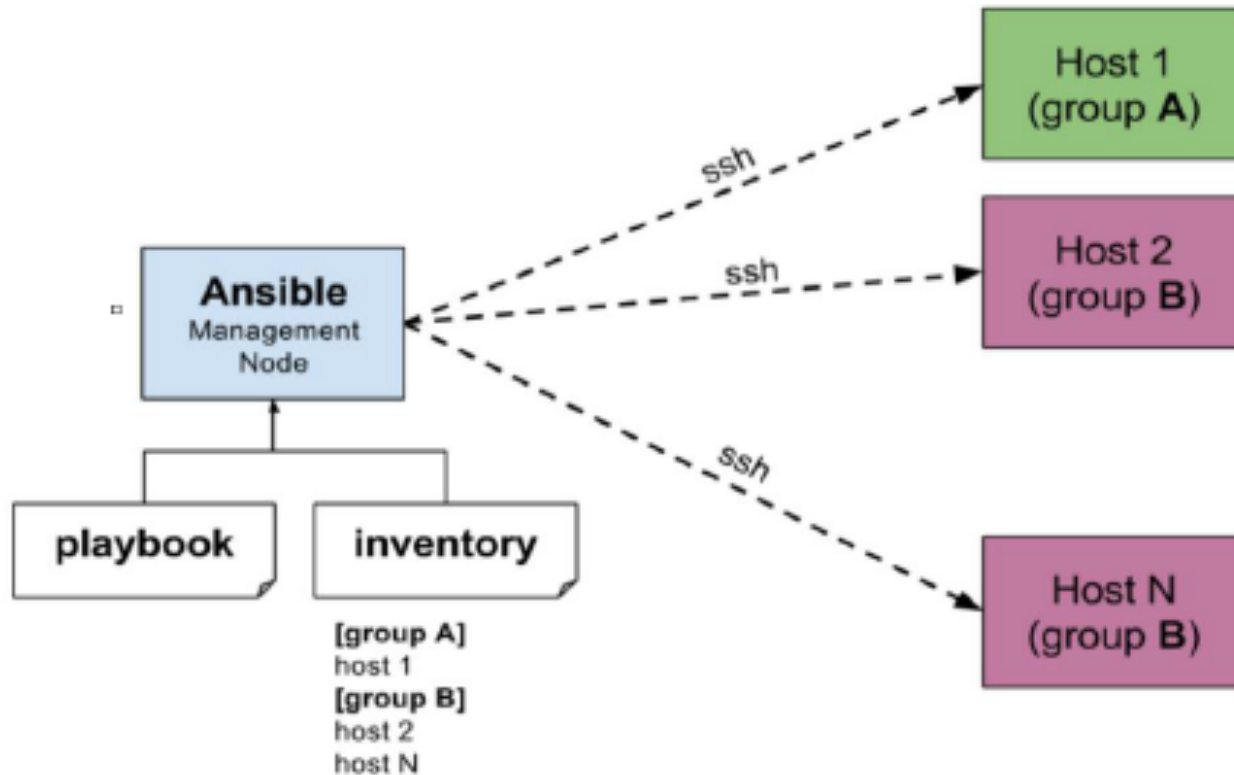
A tool for Configuration Management,
Deployment, Orchestration and
Provisioning

Gabriel Mateescu

What is Ansible

- Declare desired state of system using a YAML-based DSL
- Supports different styles
 - ad-hoc
 - playbooks
 - Roles
- Includes a rich set of modules
 - custom modules can be defined by developers
- Uses SSH and Python
 - YAML scripts converted to Python code that runs on target hosts
- Push model, agentless
- Idempotent
- Replaces using complex shell scripts and supports DRY

Ansible architecture



Ansible concepts

- Inventory
- Modules
- Ad Hoc Commands
- Playbooks
 - Tasks
 - Variables
 - Templates
 - Roles

Inventory

```
[cluster:children]
```

```
webservers
```

```
dbservers
```

```
proxies
```

```
[webservers]
```

```
ws1.example.com http_port=80
```

```
ws2.example.com http_port=8080
```

```
[dbservers]
```

```
db[01:02].example.com
```

```
[dbservers:vars]
```

```
Dbuser=joe
```

Ad Hoc Commands

- Run a single, one-line command on one or more hosts

```
$ ansible all -m ping
```

```
$ ansible all -m command -a "date"
```

```
$ ansible cluster -a "date"
```

```
$ ansible dbservers -a "service postgresql restart"
```

Modules

- Perform well-defined tasks and accept arguments to define the desired state
- Modules perform their actions in an idempotent fashion, except for the special command module.
- Modules can be used in playbooks as well as in ad-hoc one-liners

Playbooks

- Allow expressing complex configuration management
- A playbook contains one or more plays
- All tasks in one play are applied to the same set of hosts
- Each task is applied to all the hosts in the play, potentially concurrently across several hosts
- Tasks are executed in sequential order

Variables: Example

servicex:

location: '/path/to/install/dir'

files:

- {file: 'file1.j2', target: '/etc/servicex/file1.sh'}
- {file: 'file2.j2', target: '/etc/servicex/file2.sh'}

options: ['opt1','opt2']

user: {name: 'servicex', group: 'services'}

Roles

- Create groups of reusable tasks
- Support reusability of Ansible tasks

- apache
- artifactory
- git
- Iptables
- jdk-7
- jenkins
- nginx
- postgres
- tomcat

References

- <http://docs.ansible.com/ansible/latest/>
- <https://github.com/ansible/ansible-examples>
- <https://galaxy.ansible.com/>