← Previous Next →

Using Ansible like library programming in Python

Posted on 2015/01/21 by **Oriol Rius**

Reading time: 2 – 4 minutes

Ansible is a very powerful tool. Using playbooks, something like a cookbook, is very easy to automate maintenance tasks of systems. I used Puppet and other tools like that but IMHO Ansible is the best one.

In some cases you need to manage dynamic systems and take into advantage of Ansible like a Python library is a very good complement for your scripts. This is my last requirement and because of that I decided to share some simple Python snippets that help you to understand how to use Ansible as a Python library.

Firstly an example about how to call an Ansible module with just one host in the inventory (test_modules.py):

```
#!/usr/bin/python
import ansible.runner
import ansible.playbook
import ansible.inventory
from ansible import callbacks
from ansible import utils
import json

# the fastest way to set up the inventory


# hosts list
hosts = ["10.11.12.66"]
# set up the inventory, if no group is defined then 'all' group is used by default
example_inventory = ansible.inventory.Inventory(hosts)

pm = ansible.runner.Runner(
    module_name = 'command',
    module_args = 'uname -a',
    timeout = 5,
    inventory = example_inventory,
    subset = 'all' # name of the hosts group
    )

out = pm.run()

print json.dumps(out, sort_keys=True, indent=4, separators=(',', ': '))
```

As a second example, we're going to use a simple Ansible Playbook with that code (test.yml):

```
- hosts: sample_group_name
  tasks:
    - name: just an uname
      command: uname -a
```

The Python code which uses that playbook is (test_playbook.py):

```
#!/usr/bin/python
import ansible.runner
import ansible.playbook
import ansible.inventory
from ansible import callbacks
from ansible import utils
import json

### setting up the inventory

## first of all, set up a host (or more)
example_host = ansible.inventory.host.Host(
    name = '10.11.12.66',
    port = 22
    )
# with its variables to modify the playbook
example_host.set_variable( 'var', 'foo')

## secondly set up the group where the host(s) has to be added
example_group = ansible.inventory.group.Group(
    name = 'sample_group_name'
    )
example_group.add_host(example_host)

## the last step is set up the invetory itself
example_inventory = ansible.inventory.Inventory()
example_inventory.add_group(example_group)
example_inventory.subset('sample_group_name')

# setting callbacks
stats = callbacks.AggregateStats()
playbook_cb = callbacks.PlaybookCallbacks(verbose=utils.VERBOSITY)
runner_cb = callbacks.PlaybookRunnerCallbacks(stats, verbose=utils.VERBOSITY)

# creating the playbook instance to run, based on "test.yml" file
pb = ansible.playbook.PlayBook(
    playbook = "test.yml",
    stats = stats,
    callbacks = playbook_cb,
    runner_callbacks = runner_cb,
    inventory = example_inventory,
    check=True
    )
```

```
# running the playbook
pr = pb.run()

# print the summary of results for each host
print json.dumps(pr, sort_keys=True, indent=4, separators=(',', ': '))
```

If you want to download example files you can go to my github account: github.com/oriolrius/programming-ansible-basics

I hope it was useful for you.

This entry was posted in **System administration, Databases, Messaging and Security**, **technology** and tagged **ansible**, **delivery**, **deploy**, **programming**, **python**, **sysadmin**, **system-administrator**, **upgrade** by **Oriol Rius**. Bookmark the **permalink [http://oriolrius.cat/blog/2015/01/21/using-ansible-like-library-programming-in-python/]** .

About Oriol Rius

Pure G33k

View all posts by Oriol Rius →