Network Automation with Ansible – Part 2

April 27, 2018

Agenda

Part 1/2 [2 hr.]

- Ansible Concepts
- Lab
- Basic Playbooks
- Lab

Lab [90 min. home-work]

Part 2/2 [90 min.]

- Roles
- Lab – [In-Session and home-work]

Reference

Acknowledgement

Appendix

Ansible Recap

- Ansible is Open source, agentless and connects through SSH
- Ansible is easy to start and simple; can scale down and up
- Ansible.cfg, inventory file and Yaml
- Playbook is a script; Push-based & runs to completion
- Wide adoption and driven through community support
- Services opportunity seen with development of modules, roles, and playbooks

Yaml Recap

- Space indentation is important
- List
  - Ordered Data
  - Always starts with “-”
- Dictionary
  - Key: Value pairs
- List of dictionaries used for roles

```
# Lists with dictionary

---
router_hostname:
   - { hostname: router1 }
   - { hostname: router2 }
   - { hostname: router3 }
...
```

```
# List with dictionary with many variables

---
router_variables:
      - hostname: router-rtr1
        timezone: EST
        timezone_dst: EDT
        timezone_offset: -5

      - { hostname: router-rtr2, timezone: EST,
timezone_dst: EDT, timezone_offset: -5 }
...
```

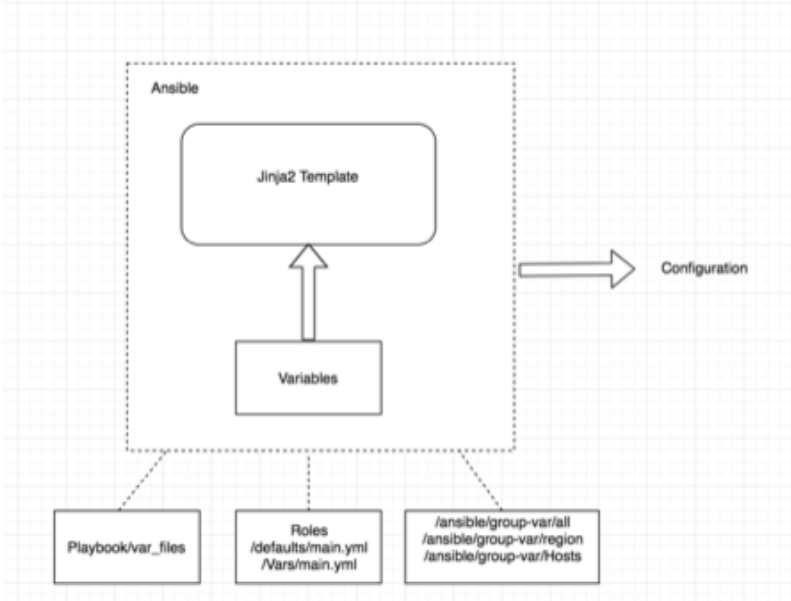
Roles

- Organize a large playbook into reusable file structures/multiple files
- Creates a separation of functions; variables, tasks, & templates in unique directories
- Expects files main.yml, and .j2 files in respective folders
- File structure can be created manually or automatically via ansible CLI – "**ansible-galaxy**"

```
[roles/
├── xr-config  >> Name of this role
│   ├── defaults >> default variables for the role
│   │   └── main.yml
│   ├── files >> contains files which can be deployed via this
role
│   ├── handlers >> contains handlers, can used by this role or
anywhere outside this role
│   │   └── main.yml
│   ├── meta >> defines some meta data for this role
│   │   └── main.yml
│   ├── README.md
│   ├── tasks >> contains the main list of tasks to be executed
by the role
│   │   └── main.yml
│   ├── templates >> contains templates which can be
deployed via this role
│   └── vars>> contains variables used in this role
```

Templating in Ansible

- Templates contain common and device/role specific elements
- Ansible uses Jinja 2 Templating language for access to variables and logic/dynamic expression
- Jinja 2 template files end with .j2 ext
- Ansible can automatically access the Jinja2 templates through its Python API



Role with lists with single variables – Example 1

- Creating a role to generate configuration across multiple devices

```
# Playbook to execute the role for XR
- name: Create a config for router` from template
XR
  hosts: localhost
  gather_facts: no

  roles:
    - xr-config

# playbook for executing role of xr-config
```

```
# Executes main.yml in xr-config/tasks/main.yml
- name: Generate the configuration from templates
  template: src=xr-config-template.j2
dest=/home/cisco/{{item.hostname}}.txt
  with_items:
    - "{{ router_hostname }}"

# tasks file for xr
```

```
# Variable defined in xr-config/vars/main.yml

---
router_hostname:
   - { hostname: router-rtr1, timezone: EST,
timezone_dst: EDT, timezone_offset: -5 }
   - { hostname: router-rtr2, timezone: EST,
timezone_dst: EDT, timezone_offset: -5 }
```

```
# Leverages j2 template for standard and variable config
hostname {{item.hostname}}
service timestamps log datetime msec
service timestamps debug datetime msec
clock timezone {{item.timezone}} {{item.timezone_offset}}
clock summer-time {{item.timezone_dst}} recurring
```

Jinja2 Template – For loop

- For Loop is a continuous loop until it runs out of inputs variables
- For Loop is invoked using {% for x in y %} syntax and ends with {% endfor %} syntax

```
# /template/template.j2
{% for interface in interface_list %}
interface {{interface}}
cost 1
!
{% endfor %}
  !

# /vars/main.yml
Interface_list:
  - GigabitEthernet0/0/0/0
  - GigabitEthernet0/0/0/1
```

Hierarchical templates and Block configs

- Base template *.J2 is pulled to specific template through {% extends ”base_config_template.j2" %} knob
- Configurations from specific template are inserted through block configs that being with { % block x %} and end with { % endblock % }

```
## Config lines from lsr_config referring base
template

{% extends "ler_lsr_config_template.j2" %}
```

```
#/templates/ ler_lsr_config_template.j2
hostname {{item.hostname}}
service timestamps log datetime msecservice
timestamps debug datetime msectelnet vrf default
ipv4 server max-servers 10telnet vrf Mgmt-intf
ipv4 server max-servers 10domain name
virl.infodomain lookup disablecdp
{% block rsvp %}
{% endblock %}
!,,
```

```
#/templates/ lsr__config.j2

{% block rsvp %}
!
rsvp
{% for interface in interface_list_ler %}
 interface {{interface}}
  bandwidth percentage 100
 !
{% endfor %}
{% endblock %}
```

Lab Exercises

- Exercise A – Create a playbook using role and Jinja2 template
  - Utilize roles to generate simple config by passing template and variable
- Exercise B – Create a playbook utilizing looping function
  - Utilize roles and Jinja2 template to create a config with looping function
- Exercise C – Create BGP generation for different device types
  - Utilize the templates and variables for config generation for different OS type
- Exercise D - Hierarchical Template
  - Utilize Hierarchical Template model for config generation

Conclusion

- Ansible is an open-source, agentless automation tool
- Automate repetitive tasks with Ansible
- With increasing support of modules, it is possible to automate even more network functions through Ansible.
- AS Services opportunity around developing modules, roles and playbook

Reference

- Ansible user guide URL & installation guide URL
- YAML resources
  - Version 1.2 Specs: http://www.yaml.org/spec/1.2/spec.html
  - http://docs.ansible.com/ansible/latest/YAMLSyntax.html
  - http://www.yaml.org
  - https://www.youtube.com/watch?v=cdLNKUoMc6c
  - https://www.youtube.com/watch?v=U9_gfT0n_5Q
- Ansible Training
  - Ansible for the Absolute Beginner @Udemy Click here
  - Ansible for Network Engineers @Udemy Click here
  - Kirk Byers Ansible training Jive page
- Jinjia2 Templating: http://jinja.pocoo.org/docs/dev/templates/#
- Ansible Up and Running – Lorin Hochstein

Acknowledgements

- Some material in this session are sourced from Ansible docs
  - http://docs.ansible.com/ansible/latest/index.html

Thank You