

CIS3534C GPA 11: Ansible

In this assignment we will use Ansible to run playbooks against the FishNet server application. As with the client program created in module 10, the playbooks will use APIs to access and manipulate the devices in the simulated network environment.

Assignment Prerequisites

- A VM Workstation Pro machine image will be provided by your instructor. This image contains an Ubuntu 22 image with Python 3.10 preinstalled. NOTE: if you retained the VM image used for the Lab 11, you can use it for this assignment. If you need to recreate it, follow the instructions below to clone the repositories, create the virtual environment, and install the required Python packages.

Instructions:

1. Start the Ubuntu VM and log in as ubuntu (password ubuntu).
2. After logging in you will be in the ubuntu user's home folder
3. Clone the Module 10 (CIS3534CMod10) and Module 11 (CIS3534CMod11) GitHub repositories provided by your instructor to the current folder
4. Create a virtual environment:

```
ubuntu@ubuntu22:~$ python3 -m venv env
```

5. Open two terminal windows and activate the virtual environment in both:

```
ubuntu@ubuntu22:~$ source env/bin/activate
```

6. When activated, the terminal prompt will change to include (env) at the beginning, this is how you can tell you are in the virtual environment.
7. Use pip3 to install the following packages **in the virtual environment**:
 - ansible-core
 - Flask
 - Flask-API
 - requests
8. The CIS3534CMod10 folder contains the FishNet server; in one terminal window cd into that folder and start the server using python3

```
(env) ubuntu@ubuntu22:~$ python3 FishNetServer.py
```

9. In the second terminal window, cd into the CIS3534CMod11 folder and note the files contained in that folder:

```
(env) ubuntu@ubuntu22:~$ cd CIS3534CMod11  
(env) ubuntu@ubuntu22:~$ ls -l
```

10. Modify the **getipaddr.yaml** playbook to retrieve the switch1 configuration instead of the router1 configuration (replace all occurrences of "router1" with "switch1").
11. Execute the Ansible playbook using the following command. Be sure to include the **-c local** option, this will prevent ansible from attempting to use SSH.

```
(env) ubuntu@ubuntu22:~$ ansible-playbook -c local getipaddr.yaml
```

```
"results": {
  "ansible_facts": {
    "discovered_interpreter_python": "/usr/bin/python3"
  },
  "changed": false,
  "connection": "close",
  "content_length": "88",
  "content_type": "application/json",
  "cookies": {},
  "cookies_string": "",
  "date": "Sat, 12 Nov 2022 19:56:51 GMT",
  "elapsed": 0,
  "failed": false,
  "json": {
    "hostname": "switch1",
    "ipaddr": "192.168.2.2",
    "macaddr": "ff:ee:dd:cc:bb:aa",
    "nports": 64
  },
}
```

Note the current configuration of switch1 as displayed by the playbook results, we will compare it to a modified configuration below.

12. Execute the Ansible playbook **gettoken.yaml** using the `ansible-playbook` command. Again, be sure to include the `-c local` option.

```
(env) ubuntu@ubuntu22:~$ ansible-playbook -c local gettoken.yaml
```

13. Use the mouse to highlight, right-click and copy the token from the [debug] output as highlighted below:

```
ansible-playbook -c local gettoken.yaml
PLAY [get FishNet token] *****
TASK [get token from FishNet server] *****
ok: [localhost]

TASK [debug] *****
ok: [localhost] => {
  "results['content']": "1YAqurqQ2odfdXUIoJ81M14HYtswAq9PypvdGRTC76DHNMSj"
}
```

14. Find the POST form body in the **setipaddr.yaml** file and replace the token in that file with your token (note that if your FishNetServer application restarts for any reason you will have to obtain a new token).
15. Also modify the hostname and value in the POST body to change switch1's IP address configuration instead of the router1 configuration (replace all occurrences of "router1" with "switch1").
16. Execute the **setipaddr.yaml** playbook with the `-c local` option to modify the IP address of switch1.
17. Execute the **getipaddr.yaml** playbook again to verify the new configuration.
18. After completing the assignment, Ctrl-C to exit the FishNet server application, then deactivate the virtual environment in both windows by typing **deactivate**.

```
(env) ubuntu@ubuntu22:~$ deactivate
```