Network Automation with Python and Ansible (NETPYA)

Modality: Virtual Classroom

Duration: 4 Days

CLC: 42 Units

About this course:

No two networks are the same! Learn to automate your network with a Python and Ansible skillset. Course can be taught across all major (and most minor) network vendors. Course demonstrations can be adapted to best-fit the customer's network to ensure all lessons have maximum relevance.

Course Objective:

This course teaches students to blend Python skillsets with Ansible through the lens of automating networks. Automation techniques for the most popular vendor (incl. Cisco, Juniper, Arista) will be subjects of study, however, students may request examples from vendors within their own environments. Topics begin with a focus on automating networks with Python; this skill set is then folded into a broadening understanding of automating with Ansible. Students will have programmatic experience automating enterprise class networks by the conclusion of this course (includes writing custom Ansible modules with Python).

Class is a combination of lecture, demonstration, and hands-on labs. Students are invited to share their own relevant Python and Ansible scripts with the instructor to ensure class subjects are as relevant as possible.

All notes and scripts will be made available to students by the end of each day via a cloud-share or email. Lab time will be given reinforce that day's topics and demonstrations.

Class is appropriate for those interested in automation, specifically network automation, and those looking to marry a Python and Ansible skillset.

Audience:

The primary audience for this course is as follows:

- Network Administrators
- Ansible devs
- Python devs
- Administrators interested in Automation
- Individuals interested in devops, specifically for networking

Prerequisite:

- There are no prerequisites required for this course

Course Outline:

Day 1 – Critical Python Catchup & Review

- Overview of Python and Ansible
- Python white space rules & best practices
- Printing and more Printing
- Date types and Variables
- Packing and Unpacking Variables
- f Strings
- Conditional expressions
- Relational and Boolean operators
- Lists, Tuples, Dictionaries
- Indexing and slicing
- Built-in functions
- Iterating with Loops (for and while)
- Working with files
- Software Control Management (SCM) (Git, Github, Bitbucket, Cloudshare, etc.)
- Using Python to access REST interfaces
- Working with JSON
- Python, Ansible and Paramiko
- Using Paramiko to SSH with keys and passwords
- RESTful API review
- API keys
- Paramiko Review
- Using Paramiko to SFTP with keys and passwords

Day 02 – Python and Network Automation

- Introduction to Netmiko (automating routers and switches)
- Using Netmiko to send commands to / from network devices
- Working with YAML
- Converting JSON to YAML with Python
- Ansible keywords
- YAML and JSON for data exchange
- Ansible and YAML
- Ansible Playbook components
- Tying together Python and Ansible – Using Python within Ansible
- Ansible Network Modules
- What is new in Ansible (most current updates / release notes)
- Network Agnostic modules
- Writing network playbooks
- Reviewing the construction of network playbooks
- Writing Ansible playbooks that respond to network failures

Day 03 – Blending Python and Ansible Skillsets

- Review how to use Python within Ansible

- Calling Python scripts with Ansible
- Jinja2 Templating Engine for Python (and Ansible)
- Using Templates in Ansible playbooks
- Jinja2 filters, looping, and other useful tricks for automating with Ansible
- Playbook tagging for selective runs
- When to use Python and when to use Ansible
- “Big Picture” options for using Python & Ansible within your Network
- Ansible Roles

Day 04 – Customizing Ansible with Python

- Review – Running Scripts with Ansible
- Prompting for Ansible user input
- Ansible Galaxy & Getting at Roles
- Writing a custom Ansible Module with Python
- Ansible “Engine” vs Ansible “Tower” – marketing hype, capabilities, costs, etc.
- Case Study: Automate your Enterprise Network
- When to use Python and when to use Ansible
- Writing your own Ansible modules in Python
- “Big Picture” options for using Python & Ansible within your Network
- Overview – NETCONF / YANG and what they mean for Python and Ansible
- Molecule – Testing your roles

Lab Outline

- Setting up your development machine
    - Using & Installing Python
    - Complementary Software (Visual Studio Code, Atom, Notepad++)
    - Making a Github account
- Python Lists
    - Data within mixed lists
    - Understanding More about Lists
- Python Dictionaries
    - Getting dir(obj) help() and pydoc
- Managing and Manipulating files and folders
    - Copying Files and Folders
    - Moving and Renaming Files and Folders
- Using Python if statements
- Paramiko
    - Paramiko - SSH with RSA Keys
    - Paramiko - SFTP with UN and PW
- Interacting with API's
    - API's – Cisco DNA Center
    - API's - SDWAN
- Running Netmiko
- Running your first Playbook
    - Debug Module
    - Playbook Prompts

    - register and when
- EOS Get Config and Archive
- Agnostic Network Modules
    - Network Playbook Error Handling
    - Network Playbook Precheck
    - Network Playbooks with Roles and Rollbacks
    - Securing Playbooks with Vault
    - Network Playbooks, Set Fact, and Fail
    - Debug, Loops, and YAML Lists
    - Running a script with Ansible
    - Jinja Filters
    - Ansible, Python Methods, and Jinja Filters
    - Ansible and APIs
    - Python and Ansible
    - Writing an Ansible Module with Python