Systems & Network Training

Python for network engineers

A 5 day Hands on training course

Save on expenses. This course is also available on your premises.

Description

This Python course focusses on teaching Python for use in network automation and network DevOps. We focus on getting delegates up and running with Python and network automation as quickly as possible rather than making them great programmers. In other words we concentrate on enabling delegates to use network automation libraries such as netmiko, NAPALM and Nornir, and APIs such as NETCONF and RESTCONF rather than enabling delegates to produce object oriented programs. Hands on sessions use Cisco and Juniper devices.

Key outcomes

By the end of the course delegates will be able to:

- Run Python programs.
- Read Python programs.
- Write Python programs.
- Debug Python programs.
- Automate network tasks with Python programs.
- Configure network devices with Python.
- Collect data from network devices with Python.

Training Approach

This structured course uses Instructor Led Training to provide the best possible learning experience. Small class sizes ensure students benefit from our engaging and interactive style of teaching with delegates encouraged to ask questions throughout the course. Quizzes follow each major section allowing checking of learning. Hands on sessions are used throughout to allow delegates to consolidate their new skills.

Details

Who will benefit?
Network engineers.

Prerequisites
TCP/IP foundation for engineers.

Duration: 5 days

Customer rating: ★★★★★

Generic Training	Small Class Sizes	Hands On Training	Our Courseware	Customise Your Course
Generic training complements product specific courses covering the complete picture of all relevant devices including the protocols “on the wire”.	We limit our maximum class size to 8 delegates; often we have less than this. This ensures optimal interactivity between delegates and instructor.	The majority of our courses use hands on sessions to reinforce the theory.	We write our own courses; courseware does not just consist of slides and our slides are diagrams not bullet point text.	Please contact us if you would like a course to be customised to meet your specific requirements. Have the course your way.
“*Friendly environment with expert teaching that teaches the why before the how.*” G.C. Fasthosts	*“Excellent course. The small class size was a great benefit...”* M.B. IBM	*“Not many courses have practice added to it. Normally just the theoretical stuff is covered.”* J.W. Vodafone	*“Comprehensive materials that made the course easy to follow and will be used as a reference point.”* V.B. Rockwell Collins	*“I was very impressed by the combination of practical and theory. Very informative. Friendly approachable environment, lots of hands on.”* S.R. Qinetiq

Python for network engineers

Course Content

What is Python?
Programming languages, Why Python? Python in interactive mode, Python scripts, ipython, Python version 2 versus version 3. A simple Python script. Comments. Hands on: Installing Python, Hello world.

A network example
On box vs off box Python. telnet, ssh, NETCONF, HTTP, APIs, manufacturers and API support, analysis of a simple telnetlib program. Hands on: Using Python to retrieve the configuration from a network device. Using wireshark to analyse the actions.

Python basics
I/O, operators, variables and assignment, types, indentation, loops and conditionals. Hands on: Modifying the telnet program, changing configurations on a network devices.

Functions, classes and methods
What are functions, calling functions, builtin functions, useful builtin functions, file handling, classes, objects, creating instances. Hands on: Storing configurations in files, configuring devices from files, using an inventory file to work on multiple devices.

Libraries and modules
Modules, files and packages, import, from-import, Python standard library, other packages, pip install, executing other programs. Managing Python libraries. Hands on: Using pip, installing and using ipaddress, subprocess to access netsnmp. For the more advanced, using the sockets library.

Paramiko and netmiko
SSH, enabling SSH on devices, keys. Paramiko versus netmiko, example scripts. pexpect. Hands on: Configuring VLANs from Python.

pySNMP
Gathering facts using previous methods, SNMP review, pySNMP GET, pySNMP and SNMPv3. easySNMP library. Hands on: Walking a MIB from Python.

NETCONF
What is NETCONF? Enabling NETCONF on devices, A first ncclient script, device handlers, get_config, edit_config, copy_config, delete_config, commit, validate, pyEZ, utils_config, utils.sw. Hands on: Configuration using ncclient and PyEZ. This session is expanded for those interesting in JunOS automation.

Manipulating configuration files
Builtin functions, string handling. Unicode. Sequences, strings, lists, tuples. Dictionaries. TextFSM. Regular expressions. JSON, YAML, XML, YANG, Jinja2, templates. Hands on: Jinja2 templating with Python to configure network devices.

NAPALM
Getters, configuration operations, supported devices, NAPALM transport, Config-replace, Config-merge, Compare config, Atomic changes, rollback. Example NAPLAM scripts. Hands on: Using NAPALM to gather facts, Using NAPALM for configuration management.

REST and RESTCONF
What is REST, HTTP methods, GET, POST, cURL, Postman, Python requests library. RESTCONF, a RESTCONF example. Hands on: Modifying a configuration using RESTCONF.

Scapy
What is scapy, Scapy in interactive mode, Scapy as a module. Hands on: Packet crafting from Python.

Warning
Errors and exceptions, Exception handling, try, except. Memory management. Garbage collection. Context management, With. Hands on: Improving Python code.

Nornir
What is Nornir? A network automation framework, inventories, connection management and parallelization. Nornir architecture and other libraires. Hands on: Setting up nornir, nornir fact gathering, nornir tasks.

Optional
Writing your own functions, Writing your own classes. pyntc. Hands on: Writing reusable code.