Systems & Network Training

Advanced Python for network engineers

A 5 day Hands on training course

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

Description

This course caters to network engineers aiming to enhance both their Python proficiency and network automation skills. Delving deeper into key areas such as netmiko, Nornir, and ncclient, we also focus on automating network testing and validation. Participants gain greater confidence working with Python functions, classes, objects, and error handling. The course additionally introduces more libraries like Scrapli, TTP, pyATS, Genie, pybatfish, and Suzieq, which cover parsing strategies, automation testing, validation, network analysis, observability, and telemetry. The curriculum also encompasses concurrency techniques.

Key outcomes

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

- ✔ Write Python modules and functions.
- ✔ Evaluate techniques to parse unstructured data.
- ✔ Use NETCONF filters.
- ✔ Handle Python errors effectively (try, assert…).
- ✔ Use postman.
- ✔ Automate testing and validation of the network.
- ✔ Use scrapli, Genie, batfish and Suzieq.

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
Python for network engineers.

Duration: 5 days

Customer rating: **New course**

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

Advanced Python for network engineers

Course Content

Review

CLI, NETCONF, RESTCONF, structured versus unstructured data, gNMI and when to use which. PEP 8. Naming conventions. Packages, modules, Classes and methods. The scrapli library. Netmiko versus scrapli. **Hands on:** scrapli, Dictionaries versus Regular Expressions.

Modules and Functions

Writing your own modules, containers versus packages, virtual environments. Best practices, calling functions, writing your own functions. Parameters, arguments. Named arguments, dictionaries as arguments. Builtins. Docstrings. Main. __name__, __main__ . Program arguments. **Hands on:** Getting interfaces, showing interface status using Netmiko and functions. Using dictionaries as arguments. Writing your own modules.

Parsing strategies

Turning unstructured data into structured data. textfsm, PyATS Genie parser, NAPALM getters, Template Text Parser. **Hands on:** Genie parser, TTP. Accessing structured data with lists and dictionaries.

Classes, objects and Python

Python classes in Genie, PyEZ and others . **Hands on:** studying network automation classes, objects, methods and attributes.

Configuration management – more nornir, ncclient, requests

Nornir tasks. Nornir results, Nornir functions, Nornir plugins. Nornir processors. YANG, YANG models, pyang. NETCONF hello. Capabilities. Schemas. Filters. Subtrees. XPATH. Exploring available YANG data models. NETCONF and network wide transactions. Asserting NETCONF capabilities. Configuration types. Locking configurations, commits. NETCONF data stores. Netconf-console. RESTCONF differences from NETCONF. URI construction. Postman. More XML and JSON. Git and configuration versions. **Hands on:** Nornir and Jinja2. Exploring available models, NETCONF filters. Using postman.

Python error handling and debugging

Context handlers, try, assert, logging, pdb, pytest, unit testing, chatgpt. **Hands on:** Writing code with each of the error handling methods, investigating what happens on an error. Use chatgpt to debug your code.

Python Automation Testing

Testing and validation. pyATS, Genie. Testbed file. Genie parse, genie learn, genie diff. Genie conf, Genie ops, Genie SDK, Genie harness. Xpresso. **Hands on:** Using Genie for state comparisons of the network.

Network analysis

Batfish, pybatfish, configuration analysis, analysing routing, analysing ACLs. Pandas. Pandas dataframe. Filtering and selecting values of interest. **Hands on:** Use Batfish to analyse network snapshots, find network adjacencies, flow path analysis.

Network observability

Suzieq, using docker, using as a package. Sq-poller, suzieq-gui, suzieq-cli, sq-rest-server. Namespaces and seeing devices, network state and Asserts. Time based analysis, snapshots and changes. **Hands on:** Suzieq: Gathering data from the network, analysing data from the network. Network state assertion.

Telemetry

gRPC, gNMI. CAP, GET, SET. Subscriptions. Model Driven telemetry. **Hands on:** Analysing telemetry data with Python.

Concurrency

asyncio, threads, processes. Nornir concurrency. Scrapli and netmiko concurrency. **Hands on:** Multiple SSH connections to devices at same time. Scarpli asyncio.