# https://lh4.googleusercontent.com/YAkJn7VqqmAS8Vv8-JktR7RmdchyEDaNQUZSWUg3c1E1rKm9PSRoZwMP1pBLNrGNZ_sdrzO83F1Kg_RJHQis9STLyEw8TN12v56np9ZB7Qmwb_WAsfmGdxKFLcSFMUnIe-5ZJczX **BOOK TITLE: Network Automation Cookbook**

***Subtitle: Over 100 recipes to effectively configure and manage network infrastructure***

# **AUTHOR: Karim Okasha**



# **ABOUT THE AUTHOR**

In order for us to write your author bio we need a few details. Please remember that your answers should be ***relevant*** to the book. Your bio helps sell the book so please *only* include *relevant* information:

|  |  |
| --- | --- |
| Full name | Karim Ahmed Adel Okasha |
| What is your job title? | Network Consultant |
| What is the name of the organisation you work for? | Equifax |
| What is your skillset (as relevant to the book)? | Network Consultant with special focus on Network Automation, CLoud and SDN |
| In which industry do you work? | Telecom |
|  |  |
| What University degree do you have? | Bachelor Degree in Telecommunications |
| What professional qualifications do you have? | CCIE, JNCIE, RedHat Ansible, AWS, LFCS |
| PLease detail your *relevant* work history | Design and Implementation for large scale Networks |
|  | across large Telco and Enterprise Customers |
| *Relevant* projects projects you have worked on: |  |
| 1 |  |
| 2 |  |
| 3 |  |
|  |  |
| Would you like to include your social media details (optional)? |  |

# **Can you recommend a good technical reviewer for your book?**

Mohamed Radwan , email: [abobakr.mohamed@gmail.com](mailto:abobakr.mohamed@gmail.com)

PART ONE: BACKGROUND RESEARCH

# **TARGET AUDIENCE**

Describe your target reader: what you assume about their knowledge of the topic, related topics, and technical topics generally; why they want to learn the technology; what will they want to do with it.

***Answer the following****:*

Who is your audience?

|  |  |
| --- | --- |
| 1 | Network Engineers and Devops Engineers who would like to automate common network tasks |
| 2 | Good network knowledge and basic Linux knowledge |

What is important to them?

|  |  |
| --- | --- |
| 1 | Automating network tasks |
| 2 | Adopting DevOps practices for automating the network |
| 3 | Understanding how to use Ansible and Python to simplify managing network devices |

# **COMPETITIVE BOOK TITLES**

What is unique about your book? You will need to look on Amazon at books that have been well-received – what are the top three market leading books that your book will compete with? Examine the description, table of contents and book reviews.

**List the books here**:

|  |  |
| --- | --- |
| 1 | <https://www.amazon.com/Network-Programmability-Automation-Next-Generation-Engineer/dp/1491931256> |
| 2 | <https://www.amazon.com/Practical-Network-Automation-automating-optimizing/dp/1789955653/ref=sr_1_5?keywords=Network+Automation&qid=1555049998&s=books&sr=1-5> |

# 

Please ensure that you have looked at the **description**, **table of contents** and **book reviews** for each of these books.

PART TWO: BOOK OVERVIEW

Having looked at your **target audience** and **competitor's products**, now decide, what is the **purpose** of your book:

# **OVERVIEW**

|  |  |  |
| --- | --- | --- |
| **TEMPLATE** | **EXAMPLE: Docker Cookbook** | **Your turn...** |
| Explain / Introduce the tech | Docker is an open source platform for building, shipping, managing, and securing containers. | Network Automation is a key requirement and both Ansible and Python are key tools to automate common network tasks |
| Why would a developer want to learn it? | It enables you to bring traditional and cloud native applications together advancing dev to ops collaboration and reducing time to market. | How to use both Ansible and Python to build robust network automation solutions |
| Product Breakdown: In 2 sentences, describe the “journey” the book takes the reader on. Look at your section headings for help | The book begins by setting up Docker in different environments, and explains how to work with Docker containers and images. You will gain an understanding of Docker use cases, orchestration, network and data management, security, ecosystems, and hosting platforms so that your applications facilitate effective collaboration and are easy to build and deploy. | The book begins by reviewing the basic Ansible concepts and then explains how to use Ansible to automate common network tasks across various vendor equipments in different environments like Enterprise, Data Center and Service Providers.Then, we move towards using python for automating Network devices and outline open source libraries like NAPALM and PyEZ to simplify Network Automation. |
| By the end of this book you will... | By the end of this book, you will be able to package and deploy end-to-end, distributed applications with Docker 1.8 xx and have an understanding of best practice solutions for common problems. | By the end of this book, the reader should be comfortable in using Ansible in automating network devices as well as using python Libraries like NAPALM and PyEZ for network automation |
| Anything else you would like to add? |  |  |

# 

# **LEARNING OUTCOME - WHAT WILL THE READER LEARN AND DO?**

Consider the competing books; in particular the **description**, **table of contents** and **book reviews**. Decide what the key learning objectives will be for your book. List them below:

|  |  |
| --- | --- |
| 1 | Using infrastructure as code concepts in the design and build of network solutions |
| 2 | Using Ansible in automating network devices like Cisco, Juniper, Arista, F5 and Palo Alto |
| 3 | Using Ansible to automate network resources in AWS Cloud |
| 4 | How to use Python with Netmiko and NAPALM to automate network devices |
| 5 | Introduce Nornir and how to use it with python to automate network devices |
| 6 | Using PyeZ and JSNAPy to configure and validate Juniper Devices |

PART THREE: BOOK STRUCTURE

Using your **overview**, and **learning outcomes** now decide on the structure of your book? What are your start and end points?

# **GENERAL STRUCTURE**

**Divide the book into approximately 7-12 chapters**.The **learning outcomes** you listed previously will help to inform these. **Each chapter should have a clear focus**. Each chapter title should clearly state what aspect of the overall topic the chapter deals with.

*PLEASE NOTE: Chapter titles appear on Amazon*

|  |  |
| --- | --- |
| 1 | Mastering Ansible |
| 2 | Managing Cisco IOS Devices using Ansible |
| 3 | Automating Juniper Devices in the Service Providers using Ansible |
| 4 | Building Data Center Networks with Arista and Ansible |
| 5 | Automating Application Delivery with F5 LTM and Ansible |
| 6 | Administering Multi Vendor Network with NAPALM and Ansible |
| 7 | Deploying and operating AWS networking resources with Ansible |
| 8 | Using Netmiko and Python for SSH |
| 9 | Managing Network Devices with Python and NAPALM |
| 10 | Network Automation Using PyEz and JSNAPy |
| 11 | Configuration Creation with Nornir |

PART FOUR: DETAILED OUTLINE

### **CHAPTER 1:**Mastering Ansible - [30] pages

### 

### DESCRIPTION:

This chapter serves as a refresher for you regarding the main building blocks of Ansible and how to use the different Options within Ansible in order to best use it for Network Automation. This is the foundation that you will see throughout the first section of the book in terms of using Ansible to automate network devices.

### 

### Level: *Basic,*

### Recipes (7-12 main headings):

What will the reader learn during the chapter?

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Tip Number** | **Recipe Title** | **Level** | **Description**  **(What will the reader do?)** | **Pages** |
| 1 | Building Ansible inventory | B | How to define and construct Ansible inventory | 2 |
| 2 | Building a playbook | B | How to build an Ansible playbook | 2 |
| 3 | Defining Ansible variables | B | How to use Ansible variables and how to organise them | 2 |
| 4 | Controlling Ansible options | B | How to control playbook execution and how Ansible connects to devices | 2 |
| 5 | Using Ansible conditionals | B | How Ansible conditionals work | 3 |
| 6 | Using Ansible loops | B | How Ansible loops work | 4 |
| 7 | Securing secrets with Ansible vault | I | How to secure sensitive information with Ansible vault | 3 |
| 8 | Using Jinja2 templates with Ansible | I | How to use Jinja2 templates within Ansible | 5 |
| 9 | Using Ansible network filters | I | Ansible network filters | 5 |
| 10 | Using Ansible tags | A | How to use Ansible tags | 2 |

### 

### **CHAPTER 2:** Managing Cisco IOS Devices using Ansible [30] pages

### DESCRIPTION:

This chapter outlines how to manage Cisco IOS devices using Ansible modules and illustrates how Ansible connects to Cisco IOS Devices and helps you configure and collect operational data from Cisco IOS Devices. Its main focus is to automate network tasks within the enterprise environments.

### Level: [*Basic*]

### Recipes (7-12 main headings):

What will the reader learn during the chapter?

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Tip Number** | **Recipe Title** | **Level** | **Description**  **(What will the reader do?)** | **Pages** |
| 1 | Building Network Inventory | B | How to build an Ansible Network Inventory | 2 |
| 2 | How to connect to Cisco IOS devices | B | Connecting from Ansible to Cisco IOS devices | 2 |
| 3 | Configuring Basic System Configuration | B | Configure basic system options with Ansible on IOS devices | 2 |
| 4 | Configuring Basic Interface Information | B | How to use ios\_interface module to configure basic Interface information on IOS Devices | 2 |
| 5 | Configuring L2 VLANs | B | How to use ios\_vlan to configure L2 VLANs on IOS Devices | 2 |
| 6 | Configuring Trunk and Access Interfaces | B | How to use ios\_l2\_interface to configure Trunk and Access Ports on IOS Devices | 2 |
| 7 | Configuring Interface IP addresses | B | How to use ios\_l3\_interface module to configure IP addresses on Interfaces | 2 |
| 8 | Configuring OSPF | B | How to configure OSPF on IOS Devices using ios\_config module | 2 |
| 9 | Collecting device facts | B | How to collect IOS Devices facts using ios\_facts module | 2 |
| 10 | Validating Network Reachability | I | How to use ios\_ping to validate Network Reachability | 3 |
| 11 | Retrieving Operational commands | I | How to use ios\_commands module to run arbitrary operational commands on IOS devices | 2 |

### 

### **CHAPTER 3:** Automating Juniper Devices in the Service Providers using Ansible - [30] pages

### 

### DESCRIPTION:

This chapter outlines how to manage Juniper Devices using Ansible modules and illustrates how Ansible connects to Juniper Devices and helps you to configure and collect operational data from Juniper Devices. Its main focus is to automate network tasks within a service provider environment.

### Level: Basic

### Recipes (7-12 main headings):

What will the reader learn during the chapter?

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Tip Number** | **Recipe Title** | **Level** | **Description**  **(What will the reader do?)** | **Pages** |
| 1 | Building the Network Inventory | B | How to build an Ansible Inventory | 2 |
| 2 | Connecting to Juniper devices from Ansible | B | How to connect to Juniper devices from Ansible using SSH Keys | 2 |
| 3 | Enabling NETCONF on JunOS devices using Ansible | B | How to enable NETCONF on Juniper devices | 2 |
| 4 | Configuring Generic System Options on Juniper Devices | B | How to use create Users and setup DNS on Juniper Devices | 2 |
| 5 | Configuring Juniper interfaces using Ansible | B | Configure interface information on Juniper devices | 2 |
| 6 | Configuring OSPF On Juniper Devices using Ansible | B | How to generate OSPF configuration for Juniper devices | 2 |
| 7 | Configuring MPLS On Juniper Devices using Ansible | I | How to generate MPLS configuration for Juniper devices | 3 |
| 8 | Configuring BGP On Juniper Devices using Ansible | I | How to generate BGP configuration for Juniper devices | 3 |
| 9 | Deploying configuration to Juniper devices | I | How to use junos\_config to configure Juniper devices | 3 |
| 9 | Configuring L3VPN on Juniper Devices | A | How to configure L3VPN on Juniper Devices | 2 |
| 10 | Validating Network Reachability on Juniper Devices | I | How to execute Ping to validate network reachability on Juniper nodes | 4 |
| 11 | Retrieving Operational Data from Juniper Devices | A | How to retrieve and validate operational state of Juniper nodes | 3 |

### **CHAPTER 4:** Building Data Center Networks with Arista and Ansible - [25] pages

### 

### DESCRIPTION:

This chapter outlines how to manage Arista Devices using Ansible modules and illustrates how Ansible connects to Arista Devices and helps you to configure and collect operational data from Arista Devices. Its main focus is to automate network tasks within a ***Data Center*** environment.

### Level: Basic

### Recipes (7-12 main headings):

What will the reader learn during the chapter?

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Tip Number** | **Recipe Title** | **Level** | **Description**  **(What will the reader do?)** | **Pages** |
| 1 | Connecting to Arista devices from Ansible | B | How to connect to Arista devices from Ansible | 2 |
| 2 | Creating users and using passwords for authentication | B | How to use username and passwords to authenticate with Arista devices | 2 |
| 3 | Creating users and using ssh-keys for Authentication | B | How to Use ssh-keys to authenticate users | 2 |
| 4 | Enabling eAPI on Arista devices using Ansible | B | How to enable eAPI on Arista devices | 2 |
| 5 | Configuring Arista interfaces using Ansible | B | Configure Interface Information on Arista devices | 2 |
| 6 | Creating VRFs on Arista devices | B | Building VRFs on Arista devices | 2 |
| 7 | Setting advanced routing On Arista using template module | I | How to configure advanced routing BGP on Arista using Ansible | 3 |
| 8 | Pushing custom configuration to Arista devices | I | How to use ***eos\_config*** to configure Arista devices | 3 |
| 9 | Collecting Arista device facts using Ansible | B | How to collect device facts using Ansible | 2 |
| 10 | Running arbitrary commands on Arista devices | I | How to execute operational commands on Arista devices | 3 |

### **CHAPTER 5:** Automating Application Delivery with F5 LTM and Ansible - [30] pages

### 

### DESCRIPTION:

This chapter outlines how to manage F5 Devices ,with a strong focus on F5 LTM appliances, using Ansible modules and illustrates how Ansible connects to F5 Devices and how you can configure and collect operational data from F5 Devices. Its main focus is to automate network tasks related to F5 LTM devices in both enterprise and data center environments .

### Level: Basic

### Recipes (7-12 main headings):

What will the reader learn during the chapter?

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Tip Number** | **Recipe Title** | **Level** | **Description**  **(What will the reader do?)** | **Pages** |
| 1 | Connecting to F5 devices from Ansible | B | How to connect to F5 devices from Ansible | 2 |
| 2 | Creating users on F5 devices | B | How to create local user accounts on F5 devices | 2 |
| 3 | Configuring system components on F5 devices | B | How to setup system level parameters on F5 devices | 2 |
| 4 | Configuring F5 interfaces using Ansible | B | How to configure F5 trunk ports using Ansible | 2 |
| 5 | Configuring tcp and HTTP monitors on F5 LTM | B | How to setup TCP and HTTP monitors on F5 LTM nodes | 2 |
| 6 | Configuring load balancing pools in F5 LTM | I | How to configure pool members and load balancing pools on LTM | 3 |
| 7 | Configuring virtual servers in F5 LTM | I | How to setup virtual servers for application load balancing on F5 LTM | 3 |
| 8 | Collecting device facts from F5 LTM | I | Collecting devices facts from F5 LTM devices | 4 |

### 

### 

### **CHAPTER 6:** Administering Multi Vendor Network with NAPALM and Ansible - [25] pages

### 

### DESCRIPTION:

This chapter explores an Open Source library for Network Automation called NAPALM and outlines how it can be used to simplify network automation in a multi-vendor environment. We then outline how this library can be integrated with Ansible to perform various network tasks like collection operational data, validation and configuring network devices.

### Level: *Intermediate*

### Recipes (7-12 main headings):

What will the reader learn during the chapter?

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Tip Number** | **Recipe Title** | **Level** | **Description**  **(What will the reader do?)** | **Pages** |
| 1 | NAPALM overview | B | Understanding NAPALM scope and its support matrix | 2 |
| 2 | Installing NAPALM with Ansible | B | How to use NAPALM with Ansible | 2 |
| 3 | Retrieving network facts with NAPALM | B | How to retrieve different operational state using NAPALM | 5 |
| 4 | Retrieving routing information from network devices | I | How to get route information from networking devices | 4 |
| 5 | Configuring network devices with NAPALM | I | How to configure network devices with NAPALM | 3 |
| 6 | Validating and auditing network with NAPALM | I | How to validate and audit network configuration with NAPALM | 4 |
| 7 | Building reports with NAPALM and Ansible | I | How to build operational reports with Ansible and NAPALM | 4 |

### 

### **CHAPTER 7 :** Deploying and operating AWS networking resources with Ansible - [20] pages

### 

### DESCRIPTION:

This chapter explains how to deploy and operate AWS networking resources using Ansible. We will outline how to deploy VPCs, subnets, route tables and also security enforcements like security groups and network ACLs.

### Level: *Intermediate*

### Recipes (7-12 main headings):

What will the reader learn during the chapter?

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Tip Number** | **Recipe Title** | **Level** | **Description**  **(What will the reader do?)** | **Pages** |
| 1 | Connecting to AWS API from Ansible | B | How to connect to AWS API using Ansible | 2 |
| 2 | Building a new VPC in AWS account | B | How to build a VPC using Ansible | 2 |
| 3 | Building subnets in a VPC using Ansible | B | How to configure and setup subnets in a VPC using Ansible | 2 |
| 4 | Building internet gateway in a VPC | I | Building an internet gateway within a VPC | 3 |
| 5 | Adjusting subnet route table | I | Creating and adjusting route table within a VPC | 2 |
| 6 | Building a VPN gateway in a VPC | I | Creating a VPN gateway within a VPC | 2 |
| 7 | Configuring security groups | I | Configuring and adjusting security groups | 2 |
| 8 | Configuring network ACLs | I | Creating and configuring network ACLs | 2 |
| 9 | Gathering VPC and subnet facts | I | Gathering VPC and subnets information | 3 |

### 

### 

### 

### 

### **CHAPTER 8 :** Using Netmiko and Python for SSH - [25] pages

### 

### DESCRIPTION:

This chapter introduces you to the Netmiko Python library and outlines how to use it to establish SSH connection to various network devices. You will use Netmiko to issue operational commands and configure Network devices in various tasks.

### Level: *Intermediate*

### Recipes (7-12 main headings):

What will the reader learn during the chapter?

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Tip Number** | **Recipe Title** | **Level** | **Description**  **(What will the reader do?)** | **Pages** |
| 1 | Installing Netmiko and setting up the environment | B | Prepare the python environment for installing Netmiko | 2 |
| 2 | Connecting with Netmiko to network devices | B | How to establish SSH sessions to different network devices | 2 |
| 3 | Sending operational commands to network devices | B | How to use Netmiko to send arbitrary network commands from Netmiko | 5 |
| 4 | Configuring network devices using Netmiko | B | Using Netmiko to configure network devices | 4 |
| 5 | Parsing operational commands using TextFSM | I | Parsing Operational data using TextFSM with NetMiko | 4 |
| 6 | Advanced NetMiko configuration options | I | Explore advanced options to tweak Netmiko | 2 |

### 

### 

### **CHAPTER 9:** Automating Multi Vendor Networks with Python and NAPALM - [30] pages

### 

### DESCRIPTION:

In this chapter, you will start to explore how to use python for Network Automation using the open source NAPALM library that help to manage network devices in Multi-Vendor environments. We outline how to use this library and what features and methods it provides in order to simplify network automation within python.

### Level: *Intermediate,*

### Recipes (7-12 main headings):

What will the reader learn during the chapter?

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Tip Number** | **Recipe Title** | **Level** | **Description**  **(What will the reader do?)** | **Pages** |
| 1 | Explore NAPALM python library | B | Setting the environment to use python NAPALM library | 2 |
| 2 | How to connect to network devices using NAPALM | B | How to connect to network devices using NAPALM | 2 |
| 3 | Gathering network device facts using NAPALM | I | Gathering operational data and device facts using NAPALM | 4 |
| 4 | Advanced facts collection using NAPALM | I | Advanced fact collection with NAPALM | 4 |
| 5 | Validating network reachability using NAPALM | I | Testing network reachability using NAPALM | 4 |
| 6 | Configuring network devices with NAPALM Part 1 | I | Configuring network devices using NAPALM | 3 |
| 7 | Using Jinja2 templates with NAPALM to configure network devices | I | How to use JINJA2 templates with NAPALM | 3 |
| 8 | Auditing network devices with NAPALM | I | How to use NAPALM to validate and audit network configuration | 3 |

### 

### 

### **CHAPTER 10:** Network Automation Using PyEz and JSNAPy - [25] pages

### 

### DESCRIPTION:

In this chapter, we explore two open source python Libraries from Juniper for managing and configuring network devices. PyEZ is a generic python library for managing and configuring Juniper devices and JSNAPy is a validation framework for Juniper Devices.

### Level: Advanced

### Recipes (7-12 main headings):

What will the reader learn during the chapter?

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Tip Number** | **Recipe Title** | **Level** | **Description**  **(What will the reader do?)** | **Pages** |
| 1 | Installing both PyEZ and JSNAPy | B | How to install PyEZ and JSNAPy | 2 |
| 2 | Connecting to Juniper devices using PyEZ | I | How to connect to network devices using PyEZ | 2 |
| 3 | Sending operational commands using PyEZ | I | How to collect operational commands using PyEZ | 3 |
| 4 | Using Tables and View with PyEZ | I | How to Use Tables and View in PyEZ to extract information from Juniper devices | 5 |
| 5 | Configuring JunOS Devices using PyEZ | I | How to configure Juniper devices using PyEZ | 3 |
| 6 | Taking network snapshot using JSNAPy | I | How to perform snapshot to Juniper devices using JSNApy | 3 |
| 7 | Validating network state using JSNAPy | I | How to validate network state using JSNAPy | 4 |

### **CHAPTER 11:** Configuration Creation with Nornir - [20] pages

### 

### DESCRIPTION:

In this chapter, we introduce another Open Source and Multi Vendor python Library to manage network devices called Nornir. We outline how it is different from NAPALM and NetMiko. You will use it to execute common network tasks across multiple Network vendor devices.

### Level: *Intermediate,*

### Recipes (7-12 main headings):

What will the reader learn during the chapter?

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Tip Number** | **Recipe Title** | **Level** | **Description**  **(What will the reader do?)** | **Pages** |
| 1 | Installing Nornir | B | How to install Nornir | 2 |
| 2 | Building Nornir inventory | B | How to build Nornir inventory to manage Network devices | 2 |
| 3 | Gathering facts using Nornir | I | Gathering network facts using Nornir | 4 |
| 4 | Sending arbitrary network commands using Nornir | I | How to send arbitrary operational commands using Nornir | 4 |
| 5 | Configuring network devices using Nornir | I | Configuring network devices using Nornir | 4 |

### 

### 