Class: TCOM570 - Network Automation

Final Project: Python L3VPN Provisioning in Juniper MX Devices

By: Elijah Curtis

https://github.com/e-i-c

Date: 22 April 2024

Project Objective and Scope:

The purpose of this project is to demonstrate the power of Python in automating repetitive network provisioning tasks, particularly in bare-bones environment absent of any network automation suites. The idea for this project stems from my experience at my current job. We have no working network automation suites for tasks such as: new router configurations (thin configs, thick configs), network hardening, service provisioning to customers (i.e. VPN provisioning to customer equipment). Due to this, repetitive tasks such as L3VPN creation/provisioning are time consuming and very prone to human error if done manually. Additionally, my environment does not allow for the install of python libraries that are not included with a base python install. Thus, libraries such as paramiko and netmiko are off-limits. This means the program needed to utilize what ships with a new python install, and do so in a way that is intuitive to the end-user. The environment also restricts scripts and tools from interacting directly with network equipment (no SSH or direct login). The script itself will generate configurations based on user input, then a user can easily copy and paste the config into any Juniper MX-series router to complete the configuration.

Technology Stack and Tools:

The technology used in this project is Python. No outside networking tools, Python frameworks (other than what is included with Python) or third-party APIs are used.

Installation and Setup Instructions:

Because the intent of this project/script are to provide a script that does not require outside/third-party tools, the set-up is very simple. You must have a computer, rights to utilize the command prompt/terminal, and the .exe file I’ve created. This allows the script to be utilized in virtually any environment.

Usage Examples and Case Studies:

I have used this code in practice at work to provision VPNs for customers in production networks. I have also tested and verified the code and the generated config functions as expected. In a Service Provider environment, much of your configuration will be deploying new services, VPNs, connections etc. for customers. Using this script, I can much more efficiently deploy these configurations. My 4 coworkers have also used it over the past 2 weeks to deploy configurations to production devices. It has also led to a sharp decrease in manual configuration errors, as these configurations were previously hand-generated using Notepad ‘copy + replace’. Please see the script in use in the screenshots below.

Figure 1 - Initial Screen of Script

A screenshot of a computer

Description automatically generated

Figure 2 - Script Output/Generated Configuration

A screenshot of a computer screen

Description automatically generated

Security Considerations:

Due to the script using internal Python libraries and the fact that it does not interact directly with network equipment, there are very few security concerns. The main security concern with the script is ensuring none of the script’s code has been altered maliciously. To verify this, we can check the hash of the script prior to use. Please see below.



The full checksum is:

5e9a05a6fc1252a346a10d606ca6658b3011110cdb58b0991d64438178e7ff42

Prior to script use, users should verify that the hash has not changed from the original above.

Additionally, users should still manually comb over the generated configurations to ensure validity of the configuration before deployment to a device.

Future Development and Sustainability:

As the script serves a purpose for a very specific environment, it is not as scalable as other tools. However, this also means there is much less maintenance long-term for the script. Barring any significant changes to Junos commands and configurations, this script can be utilized for years with no significant changes required. However, it is my intention to follow up with my coworkers utilizing the script and ask them for feedback. The main feedback I have heard to this point is that I should develop more scripts for other purposes (thin configuration, thick configuration, DNS, TACACS etc.) I will be looking into this further.