











LANforge Scripting Cookbook

The LANforge Scripting Cookbook provides examples of how to run LANforge scripts and LANforge script development. The scripts are written in Python and Perl. The Python scripts are written to support functionality available in Python 3.7 for support of LANforge systems running Fedora 27

LANforge Scripting Cookbook

Running LANforge Python Scripts

1. Getting Started: A Simple Script: sta_connect2.py  
2. Getting Started: Basic: Layer 3 Traffic Generation: test_l3.py  
3. Getting Started: Basic: Layer 4-7 HTTP Traffic Generation: test_l4.py  
4. Getting Started: Basic: Layer 4-7 FTP Traffic Generation: test_l4.py  
5. Getting Started: Configure and Run Dataplane Test  

[Under development] Intermediate: Layer 4-7 Traffic Generation: test_l4.py

[Under development] Advanced: Automate Chamber View test: create_chamberview.py,
create_chamberview_dut.py, lf_wifi_capacity_test.py

LANforge Python Scripts Using LANforge Client

[Under development] Start Here: Modify wiphy radio spatial streams: lf_modify_radio.py







Setup and Run a Test Suite

6. Suite: Start Here: Initial Setup to Run Scripts Test Suite for AP Testing  

[Under development] Intermediate: Run multiple scripts in test suite



















Additional Script Documentation

Scripting General Information

7. Basics: LANforge Entity IDs  
8. Basics: LANforge GUI Introduction  
9. Basics: LANforge Scripting Introduction  

Python Script Development Instructions

























10. Python: Querying the LANforge JSON API using Python  
11. Python: Managing WANlinks using JSON and Python  
12. Python: Create Test Scripts With the Realm Class  

13. Python: Create Layer 4 Test Scripts With Python  
14. Python: Create Generic Test Scripts With Python  
15. Python: Create VAP Test Scripts With Python  
16. Python: Load Scenarios And Control Test Groups With Python  
17. Python: Record the results of a test as CSV from the REALM monitor script  
18. Python: Record the results of a test as an Excel file from the REALM monitor script  
19. Python: Define and Demonstrate Docstring Usage in Candellatech Python Scripts  
20. Python: Scan for SSIDs, BSSIDs, and Signals of wireless APs  
21. Python: Probe Ports for Information  

LANforge JSON API

22. JSON: Querying the LANforge Client for JSON Data  

Perl Scripts

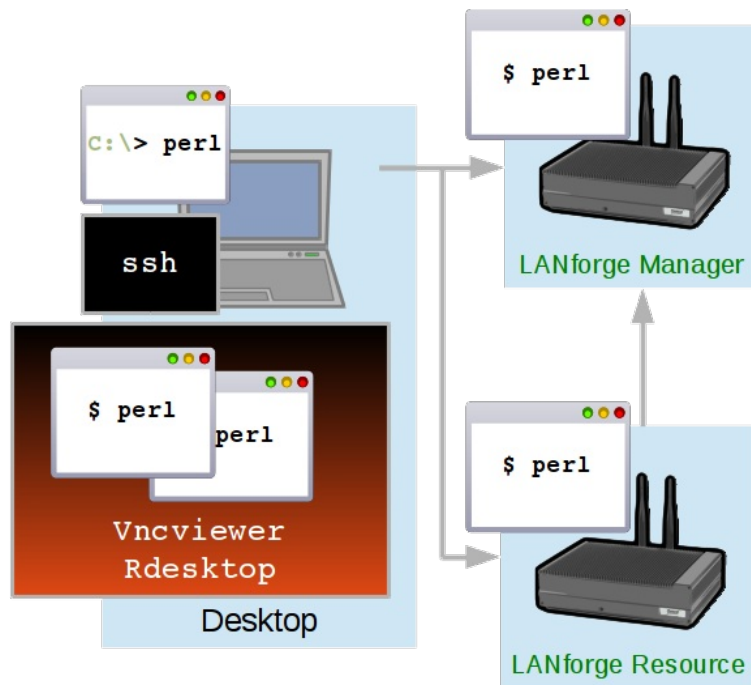
23. Perl: Perl CLI Scripts Introduction  
24. Perl: Operating LANforge scripts from Windows  
25. Perl: Monitor and Reset Ports with the portmod Script  
26. Perl: Cross Connects and Endpoints Tutorial  
27. Perl: Creating Connections with the FIREmod Script  
28. Perl: Operating Endpoint Hunt Scripts with the CLI  
29. Perl: Generating WiFi Traffic with the Associate Script  
30. Perl: Changing Station WiFi SSID with the CLI API  
31. Perl: Generic Endpoint Scripts  
32. Perl: Chamber View: Automated tests with script  
33. Perl: Control a chamber with the If_chamber.pl Script  
34. Perl: Emulating Video Transmission with Layer 3 connections  

CLI Concepts

35. CLI: Changing Station POST_IFUP Script with the CLI API  
36. CLI: Scripting Attenuation with CSV data  
37. CLI: Station CLI Operations  

Places to Run CLI Commands

You do not need to operate scripts directly from the LANforge server, and this allows you to code scripts in your preferred text editing environment. Likewise, you do not need to run a copy of LANforge Server on your desktop. Scripts will create a plain-text connection to the LANforger server you specify.



Windows Desktop

You can install a copy of the LANforge Server on your windows desktop (without a license) so that you have access to the Perl scripting libraries. Edit scripts and run them from your `C:\Program Files\LANforge-Server\scripts` directory.

Linux Desktop

You can copy the LANforge scripts folder directly from your LANforge server to your Documents directory with `scp`.

SSH or VNC connection to LANforge Server

Using `vncviewer`, `rdesktop` or `ssh` are all fine options to connect to the LANforge server to write and operate scripts. The LANforge server comes with a basic Linux desktop and you can use `emacs`, `vim`, `pluma`, or `gedit` text editors installed by default. When editing scripts on the LANforge server itself, be careful to back up your work before you upgrade LANforge. The LANforge install process will over-write scripts of the same name in the scripts directory.

Requirements for Scripts

Your desktop (or other computer) running CLI scripts needs to have a reliable (wired) connection to the management port of your LANforge server. If you are engaging in long running tests, you might consider running the scripts from the LANforge manager itself if your desktop machine needs to be powered off.

Script Libraries

CLI scripts are written using Perl. They require the libraries in `/home/lanforge/scripts/LANforge` Users may write scripts in other programming languages, such as python, but in that case, they will not be able to take direct advantage of the Perl scripts included in LANforge.

On Windows

LANforge is more fully featured on Linux, but basic support exists on Windows as well.

You can run CLI scripts from any Windows desktop as long as you have Perl installed. You can use [ActiveState Perl](#) or [Perl from the Cygwin project](#). We also highly suggest installing [PuTTY ssh client](#) to access your LANforge server.

On Linux/OS X

Most Linux distributions come with an `ssh` client and Perl already installed.

LANforge Server Requirements

The following examples will create test scenarios that work on LANforge Linux systems running the LANforge software with the LANforge kernel and a sufficient license. If you are running LANforge server using another Linux kernel, you may not be able to operate some of the examples. (Features like Armageddon,

operation of WiFi-AC radios, and WanLinks all require drivers included only in Candela provided kernels.)

Please contact us at support@candelatech.com if you have any questions.

Before Starting LANforge-CLI Traffic Generation

Before attempting the examples below, ensure that you have successfully followed these software installation guides:

- [LANforge-GUI Installation](#)
- [LANforge Server Installation](#)

If you have any questions or suggestions, email support@candelatech.com.

It is also recommended that you back up your current running LANforge Server database so that you may safely return to your current operating state.

For instance:

```
su - root
cd /home/lanforge
tar -cvzf my_db_backup.tar.gz DB
```

- [All cookbook examples in one page. Good for printing.](#)

If you have any questions or suggestions, email support@candelatech.com.

Candela Technologies, Inc., 2417 Main Street, Suite 201, Ferndale, WA 98248, USA
www.candelatech.com | sales@candelatech.com | +1.360.380.1618