



Linux PPPd for LTE CAT M1 Verizon Skywire® on Raspberry Pi 3

NimbeLink Corp

Updated: July 2018

© NimbeLink Corp. 2018. All rights reserved.

NimbeLink Corp. provides this documentation in support of its products for the internal use of its current and prospective customers. The publication of this document does not create any other right or license in any party to use any content contained in or referred to in this document and any modification or redistribution of this document is not permitted.

While efforts are made to ensure accuracy, typographical and other errors may exist in this document. NimbeLink reserves the right to modify or discontinue its products and to modify this and any other product documentation at any time.

All NimbeLink products are sold subject to its published Terms and Conditions, subject to any separate terms agreed with its customers. No warranty of any type is extended by publication of this documentation, including, but not limited to, implied warranties of merchantability, fitness for a particular purpose and non-infringement.

NimbeLink and Skywire are registered trademarks of NimbeLink Corp. All trademarks, service marks and similar designations referenced in this document are the property of their respective owners.

Table of Contents

Table of Contents	2
Introduction	3
Orderable Part Numbers	3
Prerequisites	3
PPP - 4G LTE CAT M1	4
Overview	4
Connect the NL-M1DK	4
Elevate to root	4
Install the "ppp" Package	4
Install a Terminal Program (if necessary)	4
Verify The Skywire is Connected	4
Write PPP Scripts	5
Take Down the Ethernet Interface	7
Bring Up the PPP Interface	7
Troubleshooting	8

1. Introduction

1.1 Orderable Part Numbers

Orderable Device	Description	Firmware	Carrier	Network Type
NL-M1DK	Skywire M1 Development Kit	n/a	Any	Any
NL-SW-LTE-SVZM20-ES	LTE CAT M1, Engineering Sample	30850	Verizon	LTE
NL-SW-LTE-SVZM20	LTE CAT M1	32110	Verizon	LTE
NL-SW-LTE-SVZM20-B	LTE CAT M1	37120	Verizon	LTE

1.2 Prerequisites



This document assumes you have completed the initial setup of your modem and development kit using the Skywire® M1 Development Kit User Manual up to and including Section 4.9:

<http://nimbelink.com/skywire-m1dk/user-manual/>

If you have not completed those steps, refer to the link above and complete the modem setup before proceeding.

If you are using a device that is communicating via SSH, Telnet, or any other type of Ethernet interface, you must connect to it via USB or serial. Section 2.9 requires the Ethernet interface to be taken down.

2. PPP - 4G LTE CAT M1

2.1 Overview

This example has been tested on the following distributions of Linux:

- Raspberry Pi Raspbian 8 Kernel 4.4.xx

using a Skywire® NL-SW-LTE-SVZM20 Verizon CATM1 LTE modem and a Raspberry Pi 3. Though not tested, other versions of the Raspberry Pi should work (with the Raspberry Pi Zero, you may need additional connectors). If you are using a different Linux distribution, you may need additional steps to get PPP working.

2.2 Connect the NL-M1DK

Connect a mini-USB cable from the NL-M1DK's J20 port to one of the Raspberry Pi's USB ports. Connect the Raspberry Pi to the Internet, either via Ethernet cable or Wifi network. Finally, connect to the Raspberry Pi using a keyboard, mouse, and monitor, using SSH, or using a USB-to-Serial Adapter.

2.3 Elevate to root

In order to make the changes necessary, it is necessary to login to the root account. Log in to your Raspberry Pi, and elevate to root user:

```
$ sudo su
```

You will be prompted to enter your password: enter it, followed by the Enter key.

2.4 Install the "ppp" Package

If necessary, install the ppp package by typing the following command:

```
# apt-get install ppp
```

2.5 Install a Terminal Program (if necessary)

If necessary, install a terminal program of your choice. This example uses picocom:

```
# apt-get install picocom
```

2.6 Verify The Skywire is Connected

It is helpful to verify that we can communicate with the modem using AT commands. You can use your favorite terminal program to do this: screen, minicom, picocom, etc. This example uses the built-in picocom, which is a simple terminal program for serial communications.

The syntax for picocom used in this example is:

```
# picocom -b [baudrate] [port]
```

On the NL-M1DK, the Skywire's UART communications use:

```
/dev/ttyUSB0
```

Issue the command:

```
# picocom -b 921600 /dev/ttyUSB0
```

followed by the Enter key, and you will open `miniterm.py`, connected to the Skywire.
Type:

```
AT
```

and you will get a response:

```
OK
```

If you would like to enable echo to see the AT commands, issue:

```
ATE1
```

followed by the Enter key, and you will get the response:

```
OK
```

Now, you can see AT commands as you type them.

2.7 Write PPP Scripts

We need to write two scripts for PPP to reference when initializing the connection.

Note: We have a GitHub page with the necessary PPP files available for customers to use located here:

<https://github.com/NimbeLink/skywire-ppp-scripts>

We highly recommend downloading that repo and following the instructions in the files related to your modem. The other recommended option is to click on the file you want and copy and paste directly from GitHub.

First, clone the repo and navigate to the cloned repo. Next, as superuser (root) copy the file:

```
vzw-SVZM2x
```

to:

```
/etc/ppp/peers/
```

or copy-and-paste the contents of:

```
vzw-SVZM2x
```

from the GitHub repo to

```
/etc/ppp/peers/vzw-SVZM2x
```

The contents of vzw-SVZM2x are shown below:

```
/dev/ttyS4
921600
connect "/usr/sbin/chat -v -f /etc/ppp/peers/vzw-SVZM2x-chat"
noauth
defaultroute
usepeerdns
local
debug
updetach
```

Next, copy:

```
vzw-SVZM2x-chat
```

to:

```
/etc/ppp/peers/
```

or copy-and-paste the contents of

```
vzw-SVZM2x-chat
```

from the GitHub repo to

```
/etc/ppp/peers/vzw-SVZM2x-chat
```

The contents of vzw-SVZM2x-chat are shown below:

```
TIMEOUT 3
ECHO ON
'' AT
OK AT+CGDCONT=3,"IPV4V6","[apn]"
OK AT+CFUN=1
OK AT+CGATT=1
OK AT+CGDATA="ppp",3,1
CONNECT ''
```

Change [apn] to your APN.

*Note: at time of publishing, private network support on M1 network is not in place, so VZWINTERNET is the APN that should be used. When device firmware supports private networks, all go.nimbelink.com data plans use the APN NIMBLINK.GW12.VZWENTP.

2.8 Take Down the Ethernet Interface

A PPP connection requires that any existing Ethernet connection be taken down to automatically come up as the default interface. If you are comfortable using the route command in Linux, you can set the default routes on your own without bringing down the Ethernet connection completely. This guide does bring it down as a proof-of-concept to verify functionality. To bring down the Ethernet connection, type the following command:

```
# ifconfig eth0 down
```

(Optional) To verify that the Ethernet connection is down, type:

```
# ifconfig
```

followed by the Enter key. eth0 should not be listed.

2.9 Bring Up the PPP Interface

To enable the PPP interface, type the following commands:

```
# pon vzw-SVZM2x
```

followed by the enter key. You will see the second script you wrote appear on the screen, followed by the network communication the Skywire modem is going through to get connected.

Once the process is complete, test the connection:

```
# ping -c 2 www.google.com
```

and you should receive a response similar to this:

```
PING www.google.com (216.58.216.196) 56(84) bytes of data.  
64 bytes from ord31s21-in-f4.1e100.net (216.58.216.196): icmp_seq=1 ttl=50 time=47.8 ms  
64 bytes from ord31s21-in-f4.1e100.net (216.58.216.196): icmp_seq=2 ttl=50 time=90.6 ms  
  
--- www.google.com ping statistics ---  
2 packets transmitted, 2 received, 0% packet loss, time 1001ms  
rtt min/avg/max/mdev = 47.818/69.237/90.656/21.419 ms
```

This indicates that your PPP connection is up and connected to the network.

3. Troubleshooting

- If your PPP fails with the error 0x1 in Section 2.9, chances are your vzw-SVZM2x file cannot see your vzw-SVZM2x file. Ensure that the last part of line 3 of the vzw-SVZM2x file:

```
.../vzw-SVZM2x-chat
```

has the same name as your vzw-SVZM2x-chat file. The tool xxd may be beneficial to ensure that there are no extra characters in the file and that the files are named the same:

```
$ xxd vzw-SVZM2x
```

If you are still having issues, rename the last part of line 3 in the vzw-SVZM2x file to:

```
.../vzw
```

and rename vzw-SVZM2x-chat to vzw and try again.

- If your PPP fails with the error 0x3 in Section 2.9, you may have poor signal strength. Move the unit closer to a window for a better signal strength.
- If your PPP still fails with error 0x3 in Section 2.9, your APN may be incorrect. Verify that your APN is correct when you set it in Section 2.7.