Installation of DXSpider on a RPi - v3.10

Bill Shell, N6WS   28 January 2017

General information

This installation procedure is for the latest version (January 2017) of Raspbian Jessie with Pixel or Raspbian Jessie Lite running on a Raspberry Pi 3. The procedures have also been used for a CuBox and Orange Pi PC. After installation and configuration of the OS, these procedures can be followed to install DXSpider on your system.

Installation of DXSpider on a Raspberry Pi

1. Begin by installing a few tools

```
$ sudo apt-get install mc
```

- mc is a good navigational tool with editor

```
$ sudo apt-get install wget
```

- wget is a good tool for downloading files off the Web
- wget may already be installed on some versions of Raspbian

```
$ sudo apt-get install git-core
```

- git is the tool to retrieve DXSpider and future updates

2. Setup system to install DXSpider

```
$ sudo groupadd spider
$ sudo useradd -m sysop -G spider
$ sudo usermod -g spider sysop
```

- follow the instructions to set the full name and password
- if you are not prompted for a password enter the following and add a password:

```
$ sudo passwd sysop
$ sudo adduser sysop sudo
```


```
$ sudo hostname n6ws-pi
```

- any hostname will do

3. Begin downloading perl libraries.

```
$ sudo apt-get install libtimedate-perl
$ sudo apt-get install libnet-telnet-perl
$ sudo apt-get install libcurses-perl
$ sudo apt-get install libdigest-sha-perl
$ sudo apt-get install libdata-dumper-simple-perl
```

- Adds the necessary perl modules for DXSpider

4. Finalize the Sysop setup and restart the RPi.

```
$ sudo su
# cd ~sysop
# ln -s /home/sysop/spider /spider
# shutdown -r now
```

- after restart login as sysop

5. Load DXSpider software

```
$ cd ~
$ mkdir spider
$ git clone git://scm.dxcluster.org/scm/spider spider.new
$ cp -a /home/sysop/spider.new/.git /spider
$ rm -rf spider.new
$ cd /spider
$ git reset --hard
```


6. Setup permissions on DXSpider files

```
$ sudo su
# cd /home/sysop
# chown -R sysop:spider spider
# find . -type d -exec chmod 2775 {} \;
# find . -type f -exec chmod 775 {} \;
# exit
```


7. Setup DXSpider initialization and parameter files

```
$ cd /spider
$ mkdir local
$ mkdir local_cmd
$ cp perl/DXVars.pm.issue local/DXVars.pm
$ cp perl/Listeners.pm local/Listeners.pm
$ cd local
$ mc
```

- edit DXVars.pm following the instructions in the comments
- edit Listeners.pm to remove "#" from the line for the port, and add additional ports as needed:

```
@listen = (
    ["0.0.0.0", 7300],
);
```

- exit mc

8. Begin setup of DXSpider

```
$ cd /spider/perl/
$ ./create_sysop.pl

$ cd /tmp
$ wget http://ftp.w1nr.net/usdbraw.gz
```

9. Begin Setup of usdbraw file

```
$ /spider/perl/create_usdb.pl /tmp/usdbraw.gz
```

- The step can take a few minutes

10. Compile DXSpider client

```
$ cd /spider/src
$ make
```

11. Launch DXSpider

```
$ cd /spider/perl
$ ./cluster.pl
```

12. On a separate ssh or system console (Ctrl-Alt-F2), log into the system as sysop.

```
$ /spider/perl/console.pl
```

13. From the command line of the console enter 'load/usdb':

```
N6WS de N6WS-6 16-Mar-2014 2106Z dxspider > load/usdb
```

14. Configure the system to allow for initiation of DXSpider on boot, and respawning of the process in case of failure.

```
$ sudo nano /etc/init/dxspider.conf
```

Add the following into the /etc/init/dxspider.conf file and save the file:

```
#DXSpider startup script
#
# This service maintains a getty on tty6 from the point the system is
# started until it is shut down again.


start on runlevel [2345]
stop on runlevel [!2345]

respawn

exec /bin/su - sysop -c "/usr/bin/perl -w /spider/perl/cluster.pl" >/dev/tty6
```

```
$ sudo nano /etc/init.d/dxspider
```

Add the following into the /etc/init.d/dxspider file and save the file:

```
#! /bin/sh

### BEGIN INIT INFO
# Provides:          dxspider
# Required-Start:    $local_fs
# Required-Stop:
# Default-Start:     1 2 3 4 5
# Default-Stop:
### END INIT INFO

/spider/perl/cluster.pl

case $1 in
        start|restart|force-reload)
        /spider/perl/cluster.pl
        ;;
        stop)
        ;;
        status)
        exit 0
        ;;
        *)
        echo "Usage: $0 {start|stop|status|restart|force-reload}" >&2
        exit 3
        ;;
esac

exit 0
```

Initiate dxspider as a service, and reboot the system.
- after restart login as sysop

```
$ sudo rm /etc/init/tty6
$ sudo chmod 755 /etc/init.d/dxspider
$ sudo update-rc.d dxspider defaults
$ sudo reboot
```

15. Logon to DXSpider console.

```
$ /spider/perl/console.pl
```

Installation of DXSpider on a RPi - v3.10

Bill Shell, N6WS   28 January 2017

Running your DXSpider node

To receive DX Spots you will have to arrange with another Sysop to connect to another node.

1. Contact a potential link partner, and arrange for a connection into the DX spotting network. You can do this by soliciting a link partner on the Dxpsider-support list. The list information is at http://mailman.tobit.co.uk/mailman/listinfo/dxspider-support. Once the partner is found and the adjacent node is configured to accept your connection, you can configure your DXSpider node to connect into the spotting network. There are excellent documented instructions available at http://wiki.dxcluster.org/ in the "DXSpider Installation Manual" Paragraph 6.5.

To configure your node you must set the adjacent node by node type:

For connection to a DXSpider node use the set/spider command from the console

```
$ /spider/perl/console.pl
—------
N6WS de N6WS-6 16-Mar-2014 2107Z dxspider > set/spider <adjacent_node_callsign>
```

- Enter your link partner as <adjacent_node_callsign> after set/spider

For a connection to an ARCluster node

```
$ /spider/perl/console.pl
—------
N6WS de N6WS-6 16-Mar-2014 2108Z dxspider > set/arcluster <adjacent_node_callsign>
```

2. The next step is to create a connect script to allow connection to your adjacent node partner. Detailed information on connect scripts is containe in the "DXSpider Installation Manual" Paragraph 6.6.

```
$ touch /spider/connect/<adjacent_node_callsign>
$  mcedit /spider/connect/<adjacent_node_callsign>
```

- creates the connect script file and enters the editor

Your connect script should look similar to this script

```
# /spider/connect/<adjacent_node_callsign>
#
# Create a timeout for the connection attempt in seconds
timeout 15
#
# Initiate telnet connections via internet using telnet connection information
# This connection may be using IP address or hostname followed by port number
connect telnet cluster.xyz.com 7300
#
# substitute the callsigne of your DXSpider node as <your_node_callsign>
'login:' '<your_node_callsign>'
#
# If you require a password to connect to your adjacent node partner include the following
# substitute the muturally arranged password for <password>
'Password:' '<password>'
#
# End of /spider/connect/<adjacent_node_callsign>
```

3. To initiate a connection to the your new adjacent node partner

```
$ /spider/perl/console.pl
------
N6WS de N6WS-6 16-Mar-2014 2108Z dxspider > connect <callsign>
```

- You should see the connection confirmation and DX spots should start appearing.

4. Information on setting up a crontab file in /spider/local_cmd can be found in the DXSpider Installation Manual, General Information – The crontab file.

5. It is a good idea to establish connections with more than one adjacent node partner. A redundant connection with a possible diverse path will ensure the failure of a single node does not interrupt the supply of DX spots to your node. After completion of the sequence of steps to establish the initial connection, repeat the steps for any additional connections.

Summary

That is pretty much all you have to do to get the basic DXSpider up and running on a Raspberry Pi. If you would like to add a TNC for AX25 connections, I have included basic instructions in Installation of TNC-Pi on DXSpiderPi for adding a TNC-Pi.