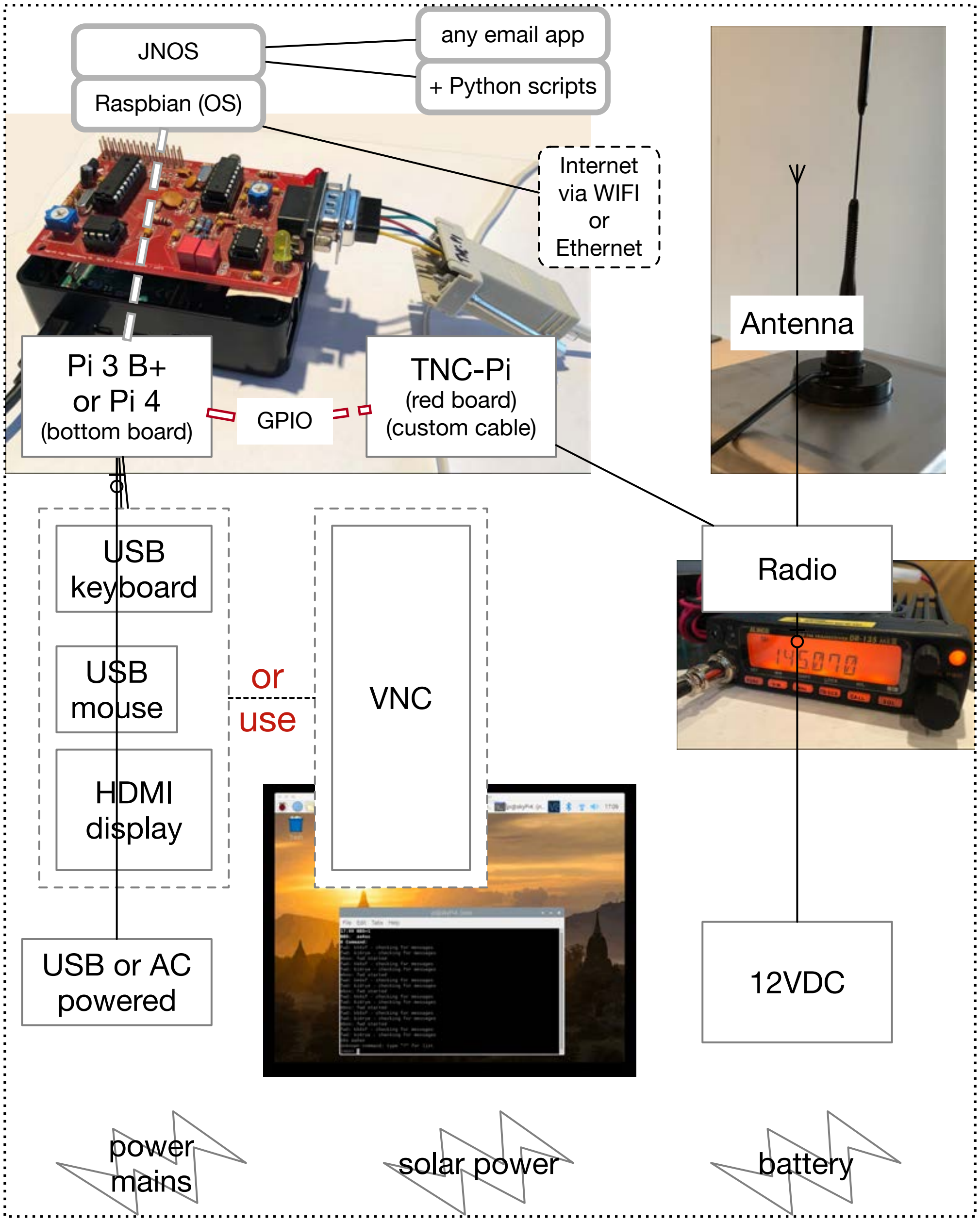
Quick-start guide to “A JNOS-based packet radio BBS with Internet email connection”

In early 2020, I constructed a system capable of sending, receiving, and forwarding (relaying) packet radio text messages to and from amateur radio packet stations, and with an ability to relay these messages to and from standard Internet email addresses. My particular configuration has certain portability improvements over earlier email gateways.

You could think of this as a “radio-based email server” and other radio stations as “client” stations in the sense of client-server. Each of those also requires radio+computer+software (though they could also be a clone of this system) to operate.



There’s another document that describes the ins-and-outs of my process. However, if you have the SD card image of my JNOS install, you can follow these simpler instructions to get up and running within a few hours.

1. The Radio

Get your components set up. You need:

- 2m radio (transceiver) with
- cable from radio to your TNC (yours must match your radio’s pin-outs!)
- USB cable from TNC to Raspberry Pi (should come with the TNC)
- Raspberry Pi (3 or 4) with an SD card pre-burned with Raspbian OS and JNOS software

- Did I say you need an SD card pre-burned with Raspbian and our JNOS?

If you are missing any of these, get ‘em set up before you continue.

Set the radio to 145.09 mHz where you will find N0ARY, KE6JJJ, AA6AX and other stations. Test it with volume on the speakers to be sure you’re hearing data transmission. (You won’t be able to verify the data, or to test transmitting, until JNOS is up and running.)

It’s important to ensure the radio is far enough from the TNC that it doesn’t cause RFI. How would you know? More is better. But if the TNC begins transmitting and doesn’t stop, or if the TNC begins and cuts off immediately after start, then there’s probably RFI involved.

2. The TNC

If you’re using the MFJ or the Kantronics USB TNC, they obtain power from USB. Others might require a separate power supply.

3. The Raspberry Pi

You’ll start from an SD card pre-configured with Raspbian OS and JNOS.

You need to attach an HDMI display, a USB keyboard and a USB mouse for your initial configuration.

Boot up the Pi. It’s configured so you do not need to log in, and you’ll want to change that, however your initial name is ‘pi’ and your initial password is ‘ooo’ … just to get you going.

Set up your Pi to use Wi-Fi or wired network. There is a Wi-fi “fan” logo in the upper right part of the menubar. Click to open and become a client computer on your local wi-fi network. If you’re wired rather than wi-fi, you are most likely already on your network with an IP address assigned via DHCP. (If you are on SFWEM, you may have a Wi-Fi hotspot, in which case you can attach via Wi-Fi to that hotspot.)

Raspberry menu > Preferences > Raspberry Pi Configuration > System

- Use **Change Password** to set a new password. Be sure **Hostname** is set to something appropriate for you and your network. From other computers on your network, you will be able to reach the Pi by addressing hostname.local (where, of course, ‘hostname’ is whatever you specify here). If you are using a SFWEM Wi-Fi hotspot, your network name will be hostname.local.mesh instead.

- The system is set to automatically log in upon power up, which means it recovers following a power failure. The **Splash Screen** is also disabled. You may wish to change those if you have a display, keyboard and mouse permanently attached and/or are in a higher security environment than I am. (VNC will still require a password for remote log-in!)

- To run without a physical display, keyboard and mouse, go to the > **Interfaces** tab and be sure you’ve Enabled **SSH**, **VNC** and **Serial Port**, and have Disabled **Serial Console**. Again, these are already configured this way on our pre-burned SD card.

- Be sure VNC is configured and running. There will be a “VNC” icon in the upper right on the menu bar. Click to open. The VNC control panel will show your local IP address, which you may wish to jot down (though hostname.local should suffice most places).

- To run VNC from a remote location, your local firewall (not the Pi, your local router or firewall) needs to be configured to “port forward” port 5900 from your Pi to the outside world. You’ll have to look up instructions on how to do that. But basically it means that VNC, which listens and responds on port 5900 on the Pi, will be exposed to the outside world via your router or firewall. Any outside connection requires name and password, so there is some protection provided. (Name is ‘pi’ and password is whatever you assigned above.)

Once you have VNC operating, a remote operator can connect to your Pi and help you through the configuration process. If you’re being mentored, this is the time to check out that remote connection!

3a. Confirm your TNC has been discovered by Raspbian

At this point you need to verify where your Raspbian system has “installed” your TNC. WIth the TNC plugged in and operating (lights on?), bring up **Terminal** and run these command-line instructions.

(Terminal can be found under Raspberry menu > Accessories > Terminal)

```
cd /dev
ls -l ttyUSB*
```

This should list one or more serial devices (most likely just ttyUSB0 if you’re under VNC at this point) but could be more than one if you have other USB devices (like keyboards). Most likely the highest numbered USB device is your TNC. As I run via VNC and do not have a keyboard or mouse, my TNC is ttyUSB0. You will need to know this to configure JNOS!

4. JNOS configuration

You must configure JNOS with your callsign and some other tweaks that depend on which TNC you’re using. JNOS uses all TNC’s in ‘KISS” mode, meaning that you don’t have to configure the TNC directly, and the TNC does not interpret any data traffic (e.g. the TNC's built in BBS is disabled when running under JNOS).

If you're comfortable editing at commend line (using vi or such editors) you can do that, but GUI editing is going to probably be faster, so the examples will utilize the **mousepad** editor.

In these examples there are two 'constants' to remember. One is your callsign, which you shoudl insert lowercase. The other is your 'area' password. This is case-sensitive. And is and should be different from your 'pi' password. This password gets you access to JNOS as the JNOS administrator. I will call this your JNOS 'area' password.

You need to edit a few files. Use the **mousepad** editor from **Terminal**:

`/jnos/autoexec.nos`

Hostname and call:

Find the line 'hostname' and put your callsign in (upper case). Below that are several 'ax25' lines. Similarly use your callsign in place of the example callsign. Your **bbscall** must have '-1' after the callsign. Insert your location (if desired) and URL in the 'bctext' line, which is the 'broadcast' text that your station will send periodically on the air.

Interface to Linux:

Just note that the line 'ifconfig tun0' creates a 'tunnel' from JNOS out to your Raspbian operating system using the IP address 192.168.2.2. If for some reason that's not an appropriate address for you to use, you can change this. Laster on you will use **Terminal** to attach to the JNOS command line and do some testing. This is the address you will use at that time. This does not have to be a routable IP address — meaning that your wi-fi network does not need to do anything with this address at all. So no worries about that! There's another line 'shell ifconfig' later on which would have to be modified in case you decide to change where JNOS appears on the local network.

Set up TNC:

Find the line 'attach asy ' and modify to use the USB device's attachment point (the /dev listing we discovered above). Mine is ttyUSB0, and yours most likely is too, but you might have to modify.

Set the baud rate of the interface in that line. My TNC-X by MFJ uses 1200 baud externally on the radio as well as 1200 baud on the USB-to-TNC connection. Some interfaces may self-discover this rate, and it's probably best to use whatever their default rate is instead of letting them discover the rate.

Logging:

I really like have a full log of all in/out traffic heard on the radio. So lower in the file there's a line that starts 'trace vhf' … what this line does is enable full logging of all packets into a local file. You could comment out the logging if you wish. Also, if you are using KPC3+ you probably used a different interface name ('kpc3p' instead of 'vhf' I used) and you'll want to log that interface by name.

```
sudo mousepad /jnos/ftpusers
```

Modify to use your callsign and your JNOS area password.

Kantronics TNC users only:
```
sudo mousepad /jnos/onexit.nos
```

If you are using a Kantronics KPC3+ TNC, remove the # from the two lines that contain “shell” commands. The files onexit.nos is executed as JNOS terminates, and these lines will return your KPC3+ TNC to its normal mode after it has been in KISS mode (during the JNOS interaction).

Kantronics TNC users only:
```
sudo mousepad /jnos/jnos.exit
```

Similar to above, remove the leading # on all lines below the comment about ‘turn off KISS mode’ … currently there are two such lines. Note that ttyUSB1 is our assumption about which device is the KPC3+ and that may be incorrect, which we will debug later.

```
sudo mousepad /jnos/popusers
```

Modify to use your callsign (lowercase) and your JNOS ‘area’ password instead of the example callsign and password.

5. Run JNOS

To run JNOS, you must open **Terminal** and run JNOS. It does not automatically start upon a reboot, which is probably how you should leave it. You don’t want it to just grab the radio and start operating unless you as control operator are available.

Raspberry menu > Accessories > Terminal

```
cd /jnos
sudo ./jnos -d /jnos
```

Note that ‘sudo’ means to switch to become the ‘root’ or master user. You may be asked to type your root password (which you changed already above). JNOS must run as the root user or it will not function properly.

This will launch JNOS. You’re up and running at this point and can test your radio.

It can be very helpful to see what’s going on on the radio. To do this, bring up a new **Terminal** session and type:

```
sudo tail -f /jnos/vhf.log
```

(or if you renamed the logging point within autoexec.nos [see above] then that log might have a different name)

This will display the log as new entries are written, in real time. You can just leave this running and put the window off to the side somewhere. If you see nothing, then JNOS isn’t running, or the TNC is deaf, or the radio is off, or the antenna is loose, or something else is really wrong.

6. Claws Mail

You need to configure the Claws Mail program to send/receive messages using your JNOS account ("area") and your remote email account. In theory you can use any email program to do this, as long as you give it the proper JNOS 'area' credentials for your JNOS account and JNOS is running.

Launch **Claws Mail** (use the Raspberry menu > Internet > Claws Mail)

Create/configure an email account. In the Configuration > Create New Account

Basic (tab): **Set as Default** and use any name you like.
    Mail Address: area@BBSNAME
    (my own 'area' is aa6ax and my BBSNAME is AA6AX)
Server for Receiving: 192.168.2.2
SMTP server (send): 192.168.2.2
User ID: (use your own area name/callsign)
Password: (use your own JNOS area password)

Receive (tab): Do not use **secure auth**. Do "**Remove messages on server when received**" but do it after some time (I use 1 day) in case there are bugs. **Check for mail** you can check as often as you wish — I check every minute. JNOS is dedicated and Claws is running and your Pi is probably loafing along, so you can check every minute if you wish. You can also set up filters and folders as much as you want, but perhaps do that later on?

Send (tab): Turn off all the **Header** options. Turn on **SMTP Authentication. (Important)** Your **User ID** is your callsign (the JNOS area name) and **Password** is your JNOS area password. Do not **Authenticate with POP**.

Compose (tab): **Signature** — don't. This takes up air time.

SSL/TLS (tab): Be sure these are set to **Don't use** for POP and SMTP both. That's because JNOS doesn't do SSL or TLS.

Advanced (tab): These should probably just default. They'll be 25 for SMTP and 110 for POP.

7. Testing/Initializing locally

At this point if you're going to be using Claws or another email program, you want to configure your JNOS'area' properly.

Using **Terminal**, log into JNOS and configure your JNOS area.

```
telnet 192.168.2.2
```

login: callsign
password:

Hit "X" return until your prompt line is abbreviated (shortest prompt).

Turn off 'paging' behavior:

```
XM 0
```

Exit from JNOS by typing

```
bye
```

8. Testing on the air

To perform a brute-force initial test, with JNOS running and with the ‘tail’ running in **Terminal**, you can do this in a new **Terminal**:

```
telnet 192.168.2.2
```

You should see a request for ‘login’ (which is asking for your callsign). Type that, then you’ll be prompted for a password (your JNOS area password), which you should type.

At that point, you can test your TNC and radio. In the example ‘vhf’ is the interface name (which was specified in autoexec.nos as ‘vhf’ in my case). And I am connecting to a local BBS KE6JJJ-1 in San Francisco.

You should see an entry in your log file (via **tail** above) and you should see your radio transmit and any transmit light on the TNC should light up briefly. If you are monitoring the frequency on another radio, you should hear your radio transmit the ax25 information.

```
login: aa6ax

Password:

[JNOS-2.0m-B1FHIM$]

You have 0 messages.

Area: aa6ax (#0) >

connect vhf ke6jjj-1

Trying...  The escape character is: CTRL-T

*** connected to KE6JJJ-1

[ARY-4.1.2-H$]


Hello Sky,

Welcome to the Bernal Heights, San Francisco packet BBS

 Help Level: 2

 Homebbs: KE6JJJ

1) AA6AX de KE6JJJ [0]>
```

You are probably a first-time user of this BBS, so you may be asked your name and some other questions (by the remote BBS). Go ahead and answer those questions. If you're comfortable having world-wide packet traffic for your callsign directed thru that remote BBS, you can select it as your 'home BBS'. This will not cause any problems with JNOS.

There is additional configuration you will want to do so our automated message pickup and delivery will work. That will be described later.

xxxxxxxxxx

xxxxxxxxxx

xxxxxxxxxx