

HamGateXX Setup

[44]

A regional hub solution for use with the 44net ham radio TCP/IP network using JNOS

<u>Editor</u>	<u>Notes</u>	<u>Version</u>
Mark Phillips, NI2O	Initial creation	0.1

Mark Phillips, NI2O April 2022

Table of Contents

1 Introduction

2 Assumptions

3 Requirements

3.1 Hardware

3.2 Software

3.2.1 Static public IP address

3.2.2 Secure Shell (SSH) keys

3.2.3 JNOS source code

3.2.4 Wireguard VPN

4 Network Services

4.1 xNOS

4.1.1 Convers

4.2 Routing

4.2.1 Not in the Internet business

4.2.2 Greater network forwarding

4.2.3 RoIP

5 Connection to other hubs

5.1 That encap thing again

5.2 Site-to-site VPN

6 RF solution

7 Installation Walkthrough

7.1 Connect via SSH

7.1.1 Change the hostname

7.1.2 Disable password logins

7.1.3 Create JNOS user

7.2 OS and software updates

7.2.1 Install and configure ampr-ripd and tunnel interfaces

7.2.1.2 Download and configure the maketun package

7.2.2 JNOS software installation

7.2.3 Edit config.h

7.2.4 Sanity check the config.h file

7.2.5 Compile JNOS

1 Introduction

To further aid in the rollout of the 44net Amateur Radio TCP/IP network to licensed ham radio operators, this document proposes a number of solutions that should allow even the most modestly skilled amateur to connect to the network.

To this end a system of regional/State hubs is proposed. Such hubs will exist on a 1 per State basis and be managed by a competent network administrator.

2 Assumptions

The following assumptions have been made;

Static public IP address available (see section 3.2.1)

Support for the IPIP Encap system

Support for both site-to-site and user-to-system VPN support

Secure Shell (SSH) only system access

Internet Access

Debian Linux 11 familiarity

JNOS familiarity

3 Requirements

3.1 Hardware

A modern 64 bit machine or Virtual Server is required for this project. That crappy old i386 you have in the corner gathering dust that you have earmarked for a project “one day” will not suffice. Do the world a favour and send it to the recycling center.

A platform with 512mb of RAM, 1 CPU (any speed over 1GHz), 25GB of disk space and an Ethernet interface is required.

Recycled/Upcycled “thin clients” and Raspberry Pi’s are ideal for this project.

3.2 Software

3.2.1 Static public IP address

A static, non-changing public TCP/IP address is needed. Cable modems and similar Internet connections are ideal as their public addresses rarely change. Various dynamic DNS services exist to keep track of your changing IP address and are recommended.

3.2.2 Secure Shell (SSH) keys

A pair of SSH keys (public and private) will be necessary for system admin level access to the host server. Instructions on how to install and enable SSH on your server platform are beyond the scope of this document.

3.2.3 JNOS source code

We'll be installing JNOS so as to provide mail, convers, nntp, news and other services. It will be compiled from the source code.

3.2.4 Wireguard VPN

This package will be required for remote access to the 44net ampr.org network via this server. It will be installed by the Debian apt package manager.

4 Network Services

4.1 xNOS

4.1.1 Convers

4.2 Routing

4.2.1 Not in the Internet business

4.2.2 Greater network forwarding

4.2.3 RoIP

5 Connection to other hubs

5.1 That encaps thing again

5.2 Site-to-site VPN

6 RF solution

Due to the cloud based nature of this solution no direct RF connectivity is possible. However, it may be possible to add remote RF devices over a VPN

7 Installation Walkthrough

For this walkthrough we have assumed the Hamgate server is installed and available over the network. We will not be using the screen/keyboard any longer.

7.1 Connect via SSH

For security reasons we shall be accessing our server via Secure Shell (SSH) only. Ensure your key pair is correctly installed and SSH is enabled on your server.

Connect to the server as the “root” user with your favourite SSH tool. Windows users may like to use the Putty package; however, configuration of this package is beyond the scope of this document.

7.1.1 Change the hostname

Changing the hostname of the machine by editing the `/etc/hostname` file. Enter only the hostname and not the FQDN. Log out and back in again to enact the change.

7.1.2 Disable password logins

Logins to any hamgate system will be by SSL key exchange ONLY. Make sure that you add your public SSL key to the `/root/.ssh/authorized_keys` file BEFORE continuing.

Edit `/etc/ssh/sshd_config` and change the following parameters

```
PermitRootLogin yes          to      PermitRootLogin prohibit-password  
#PubkeyAuthentication yes  to      PubkeyAuthentication yes
```

Save the file and then restart the SSH daemon with `'/etc/init.d/ssh restart'`. DO NOT CLOSE YOUR ORIGINAL SSH SESSION. Open a second SSH session and try logging in to the server. If successful, you should see the hamgatexx prompt without being asked for a password. You can close the original ssh session ONLY IF THIS WORKED..

7.1.3 Create JNOS user

Create a user called 'jnos'. Ensure the home directory for this user is `/home/jnos`. Set the initial password for this user to 'jnos44Net'.

7.2 OS and software updates

Just like your Windows machine, Linux requires updates. Cut and paste the following commands into the console session. Some of them may take a while to complete.

```
apt update  
apt upgrade -y  
apt install build-essential net-tools traceroute wireguard-dkms screen ntp snmpd wget -y  
apt install daemontools linuxlogo mc -y
```

7.2.1 Install and configure ampr-ripd and tunnel interfaces

Download the latest ampr-ripd package from <http://www.yo2loj.ro/hamprojects/> Untar the package and compile it. Copy the resulting ampr-ripd file into the /home/jnos directory. Change the ownership of this file to 'jnos.jnos' and add the executable bit to +r (chmod +r filename).

7.2.1.2 Download and configure the maketun package

Download the maketun tar file from <http://insert-url.here>

Edit /root/maketun.sh to reflect the 44net requirements of the host system.

7.2.2 JNOS software installation

Download the latest JNOS source code from Maiko Langelaar, VE4KLM's website <https://www.langelaar.net/jnos2/documents/dlcompinst.html> and upload them to your server using your favorite SCP tool. Make sure to obtain the .tar.gz files.

Unzip the source code into a folder like so;

```
mkdir jnos-src  
cd jnos-src  
tar -xvzf ../jnos-source-code.1.2.3.tar.gz  
tar -xvzf ../update-to-jnos.1.2.3.tar.gz
```

7.2.3 Edit config.h

```
cd /root/jnos-src  
cp config.h.default config.h  
nano config.h
```

7.2.4 Sanity check the config.h file

Before we compile the source code we need to check that the config.h file has no mistakes and also that the correct needed additional modules are installed.

Type the following and install any suggested packages.

```
./configure
```

In the below example we are required to install a few packages. Debian will use the Ubuntu command examples. Omit the word "sudo" from the beginning of each command example.

```
root@ni2o-test:~/ROOT/JNOS$ ./configure
```

```
JNOS 2.0 ./configure script Version 1.7 - 30Aug2020, Maiko Langelaar (VE4KLM)
```

```
makefile is compatible with this script
```

```
config.h found, leave it alone
```

The ncurses development package is not installed, use :

RHEL systems : yum install ncurses-devel

UBUNTU systems : sudo apt-get install libncurses5-dev libncursesw5-dev

WINLINK_SECURE_LOGIN is defined in config.h

The openssl development package is not installed, use :

RHEL systems : yum install openssl-devel

UBUNTU systems : sudo apt-get install libssl-dev

Re-run the configure command until all the packages are installed.

When you see the below message you are ready to begin compiling the JNOS source code

root@ni2o-test:/home/jnos/ROOT/JNOS# ./configure

JNOS 2.0 ./configure script Version 1.7 - 30Aug2020, Maiko Langelaar (VE4KLM)

makefile is compatible with this script

config.h found, leave it alone

ncurses development installed

WINLINK_SECURE_LOGIN is defined in config.h

openssl development installed

You should now be able to run 'make' - good luck.

7.2.5 Compile JNOS

Compiling is very simple but depending upon the speed of your chosen machine it could take a while.

Simply enter the work “make” and watch it all go.

make

When the compile is complete you will not see anything telling you it was a successful build. You should assume that if you do not see any error messages everything went well. You will be returned to the system prompt. A successful build looks like this.

```
unix.a ax25.a aprs.a dump.a ppp.a bootp.a inp.a j2curses.a \  
-Incurses -lcrypto  
cc -DNOJ2STRLWR j2pwmgr/j2pwmgr.c j2pwmgr/j2pwrtns.c -o jnospwmgr  
root@ni2o-test:/home/jnos/ROOT/JNOS#  
  
root@ni2o-test:/home/jnos/ROOT/JNOS#
```

Should your build fail, double check the config.h file for spelling errors etc. Otherwise, it is beyond the scope of this document to diagnose and repair compile time errors. Please seek help from a Linux capable system admin.

8 Enable SNMP monitoring

Monitoring is conducted by the Observium application running on HamGateCT. To allow the SNMP monitor to collect data, install the snmpd tools as well as the Observium unix-agent.

```
apt install snmpd
```

Edit the /etc/snmpd.conf file. Ensure the file contains the following data (edit to the required environment)

```
sysLocation Linode, DFW [32.8330, -97.0408]  
sysContact Charles Hargrove <n2nov@n2nov.net>  
agentaddress udp:161,udp6:[::1]:161
```

Follow the below doc to install the Observium unix-agent

https://docs.observium.org/unix_agent/