BPQ

Willem A Schreüder AC0KQ
willem@prinmath.com

RMHAM University
March 12, 2022

What is BPQ?

- NET/ROM compatible Packet Switch
  - Multiple ports
    - Serial, USB, I2C, IP
    - You need to provide the TNCs
  - Multiple protocols
    - Packet, Pactor, AXIP, VARA, ...
  - Multiple functions
    - BBS, Chat, APRS

What can we use BPQ for?

- Access point (packet, VARA, pactor, ...)
  - BPQ is typically what you connect to
- Cross band, cross mode gateway
- Bulletin Board System (BBS)
- Radio Message Server (RMS)
  - BPQ forms the backbone of the NTS-D system
- APRS Internet Gateway
- Application Gateway

How does it work?

- BPQ is a software program
  - Runs on most computers
  - Somewhat complex configuration file
- Connects to radio via Terminal Node Controller (TNC)
  - Usually serial, USB or I2C
- Interconnects via IP
- Built-in BBS, iGate, Chat server, ...

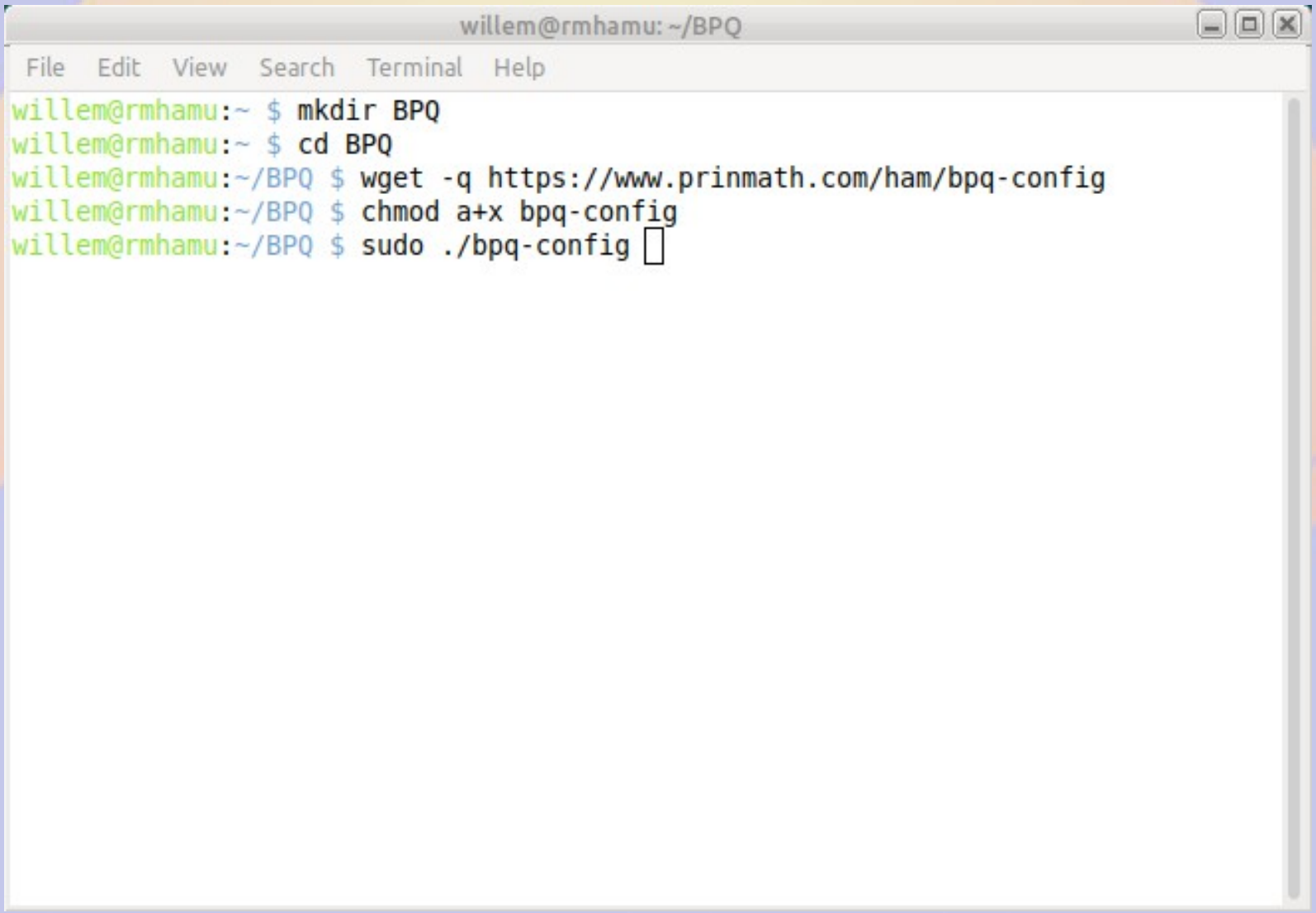
Complaint: Hard to set up BPQ

- BPQ is very sophisticated, and that necessarily adds a lot of complexity
- Solutions:
  - Use *bpq-config* to get started
  - Web interface once configured
  - Join a support group

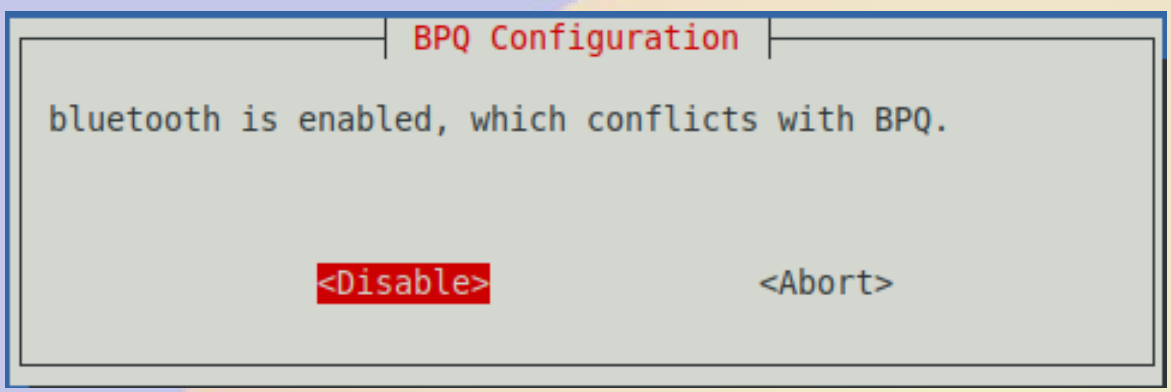
Brief history of BPQ

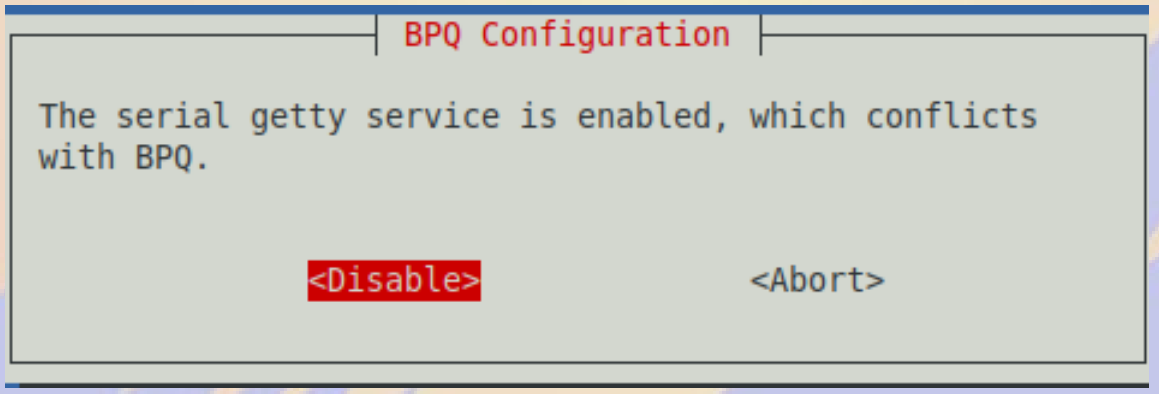
- Written by John Wiseman G8BPQ
- Originally called BPQCODE
- Became BPQ32 in late 90s
- Ported to OSX/Linux in 2000s
- Ported to Raspberry Pi/TNC-PI and Beagle Bone Black/TNC-Black

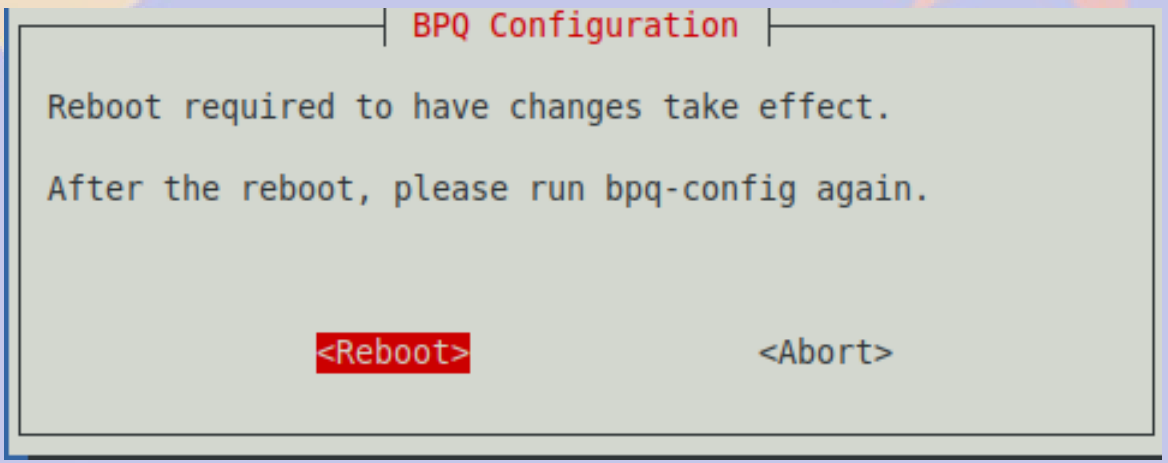
Download and run bpq-config



Disable conflicting stuff

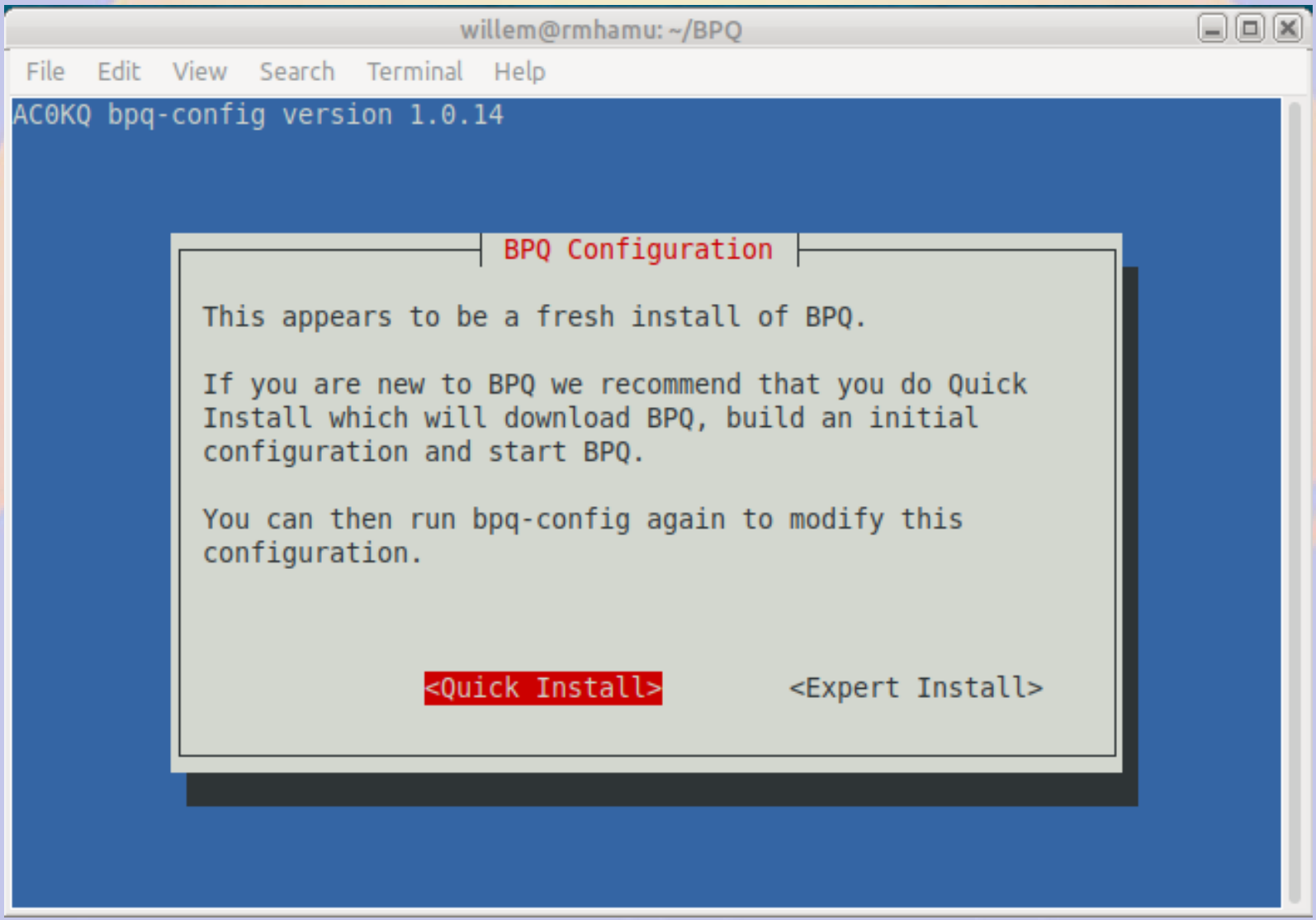






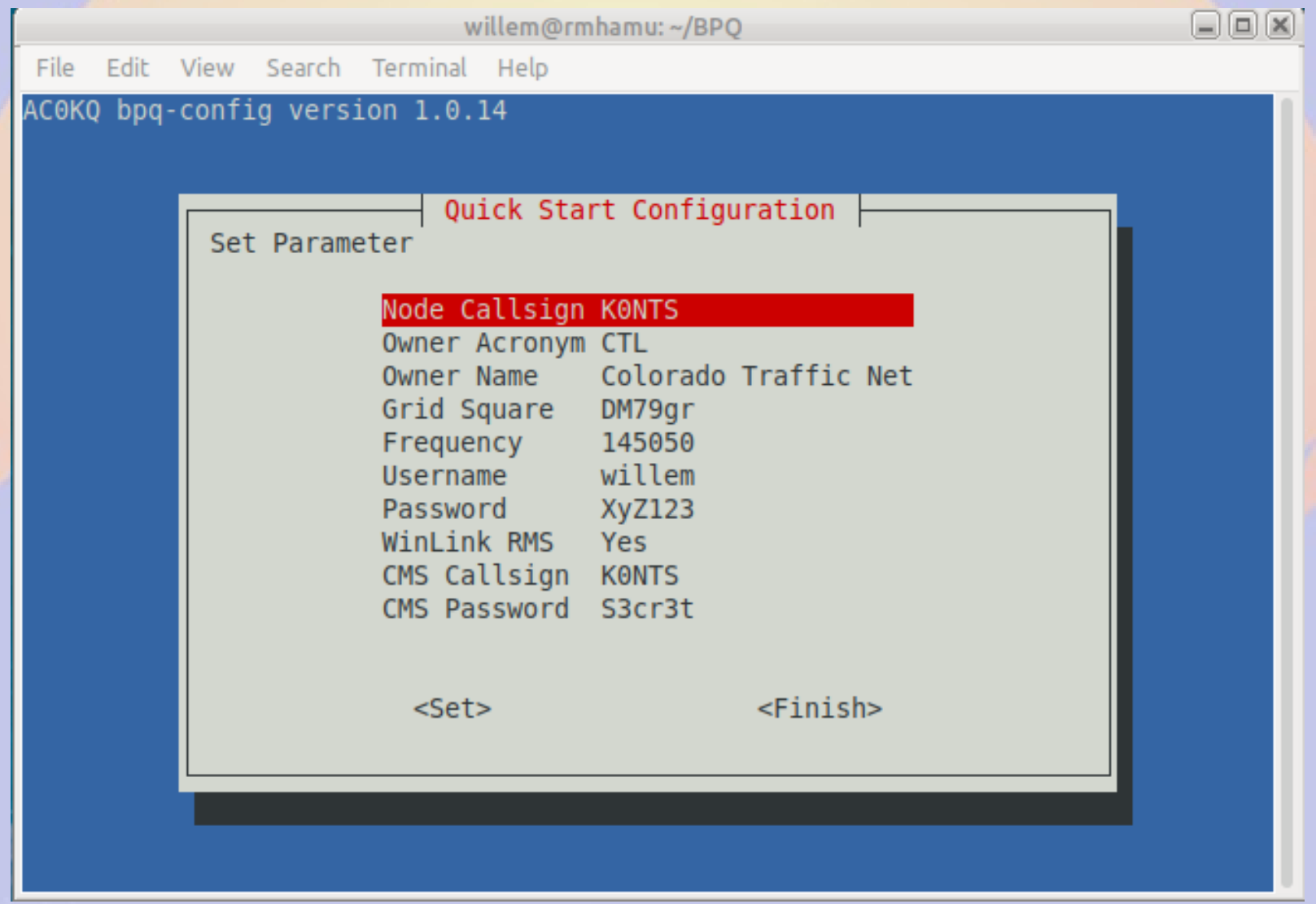
Do the Quick Install

This downloads a BPQ and everything else needed

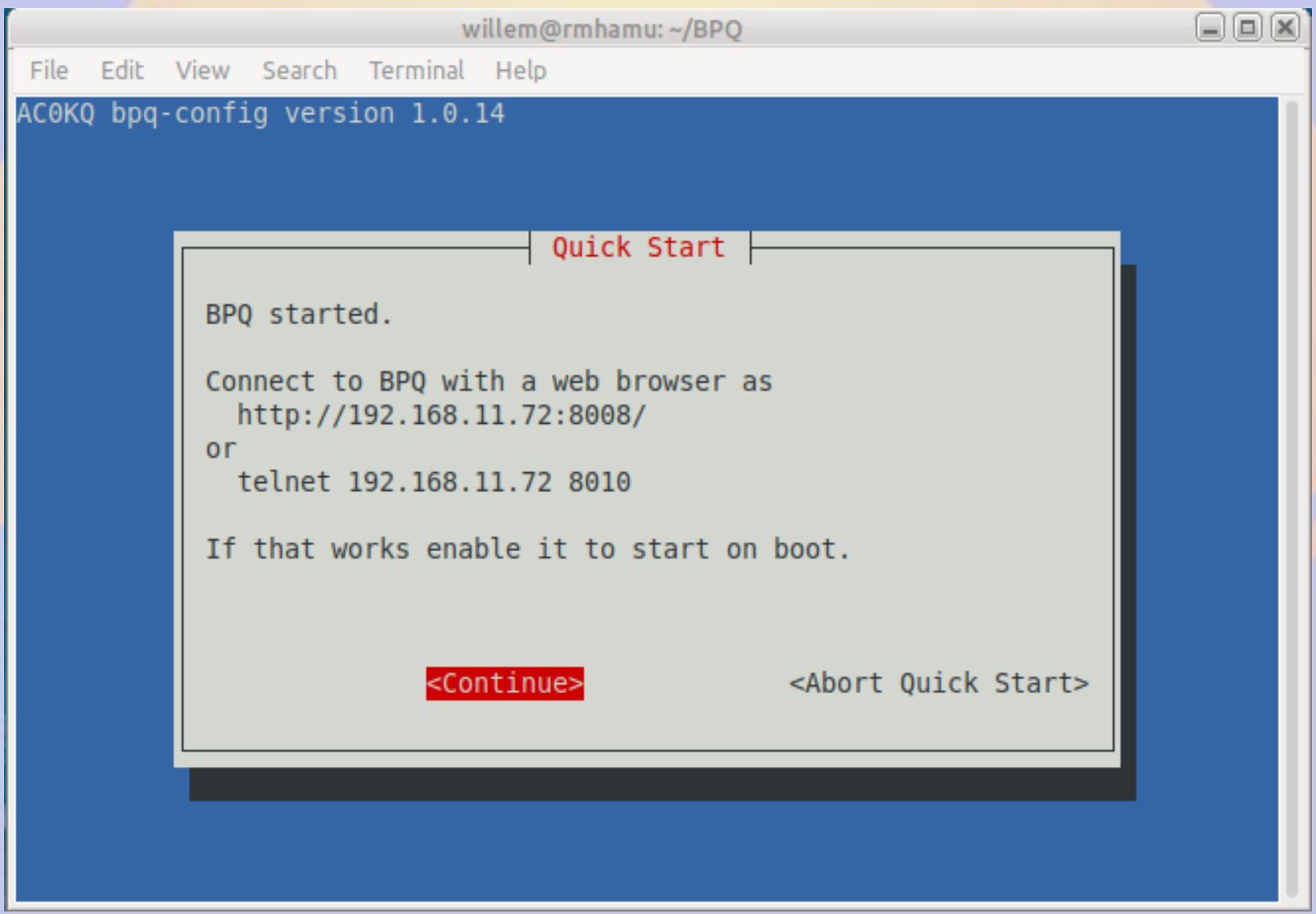


Quick Start Configuration

Fill in all the fields with your values

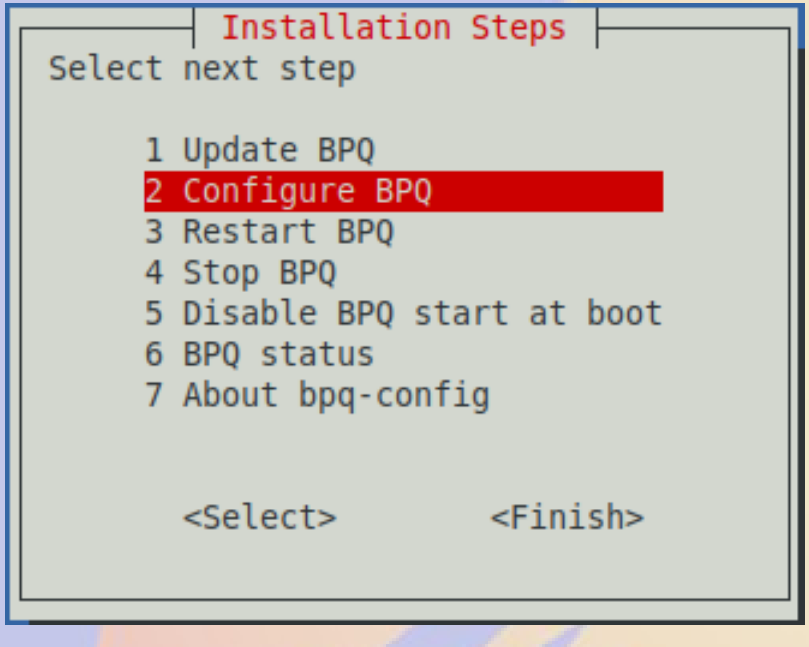


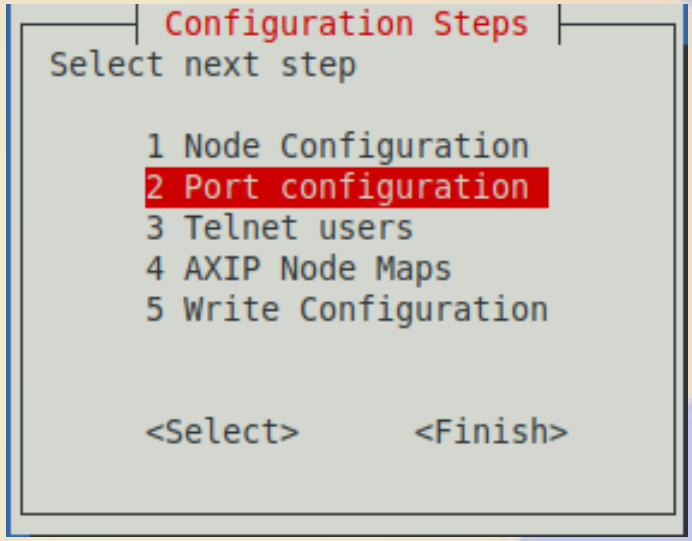
Where to find BPQ

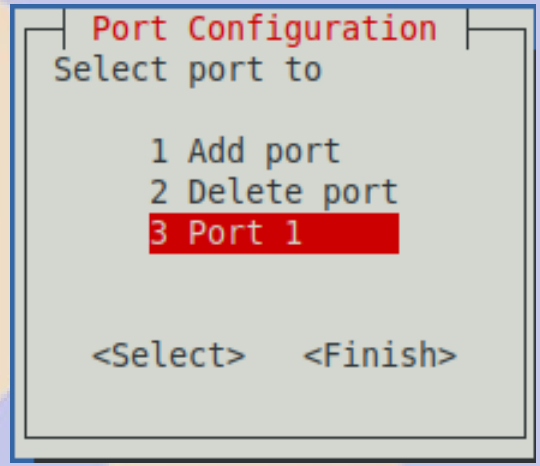


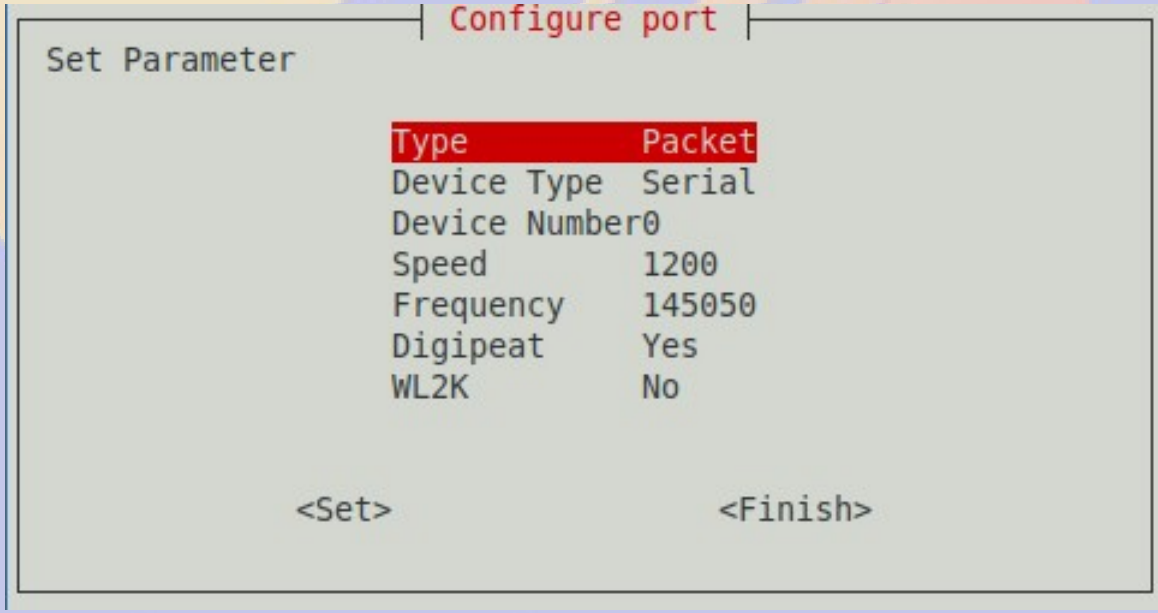
Port Configuration

Quickstart defaults to first serial port – edit if necessary

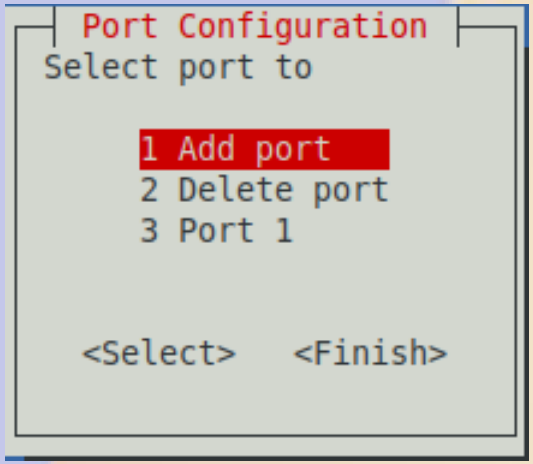


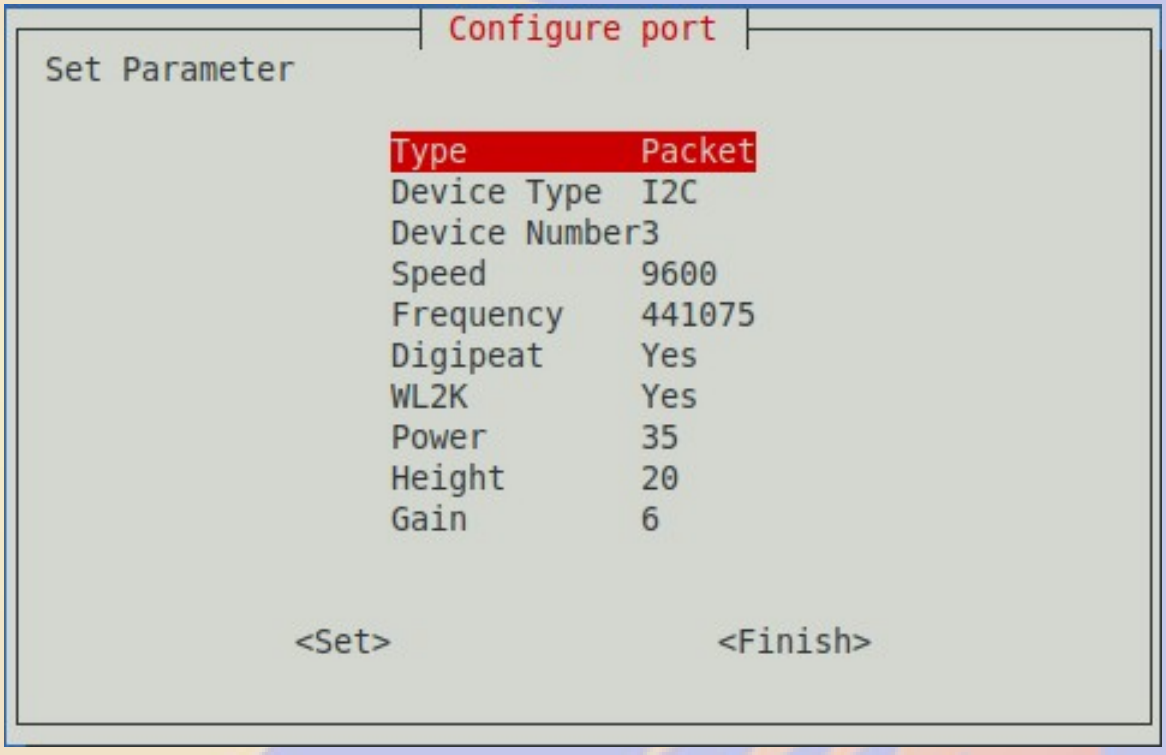


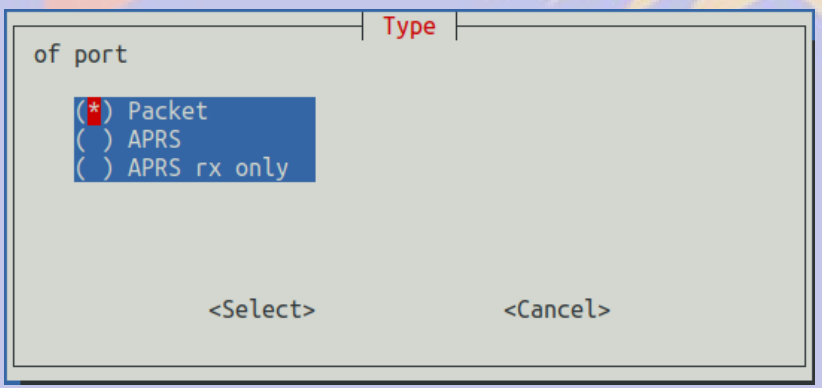


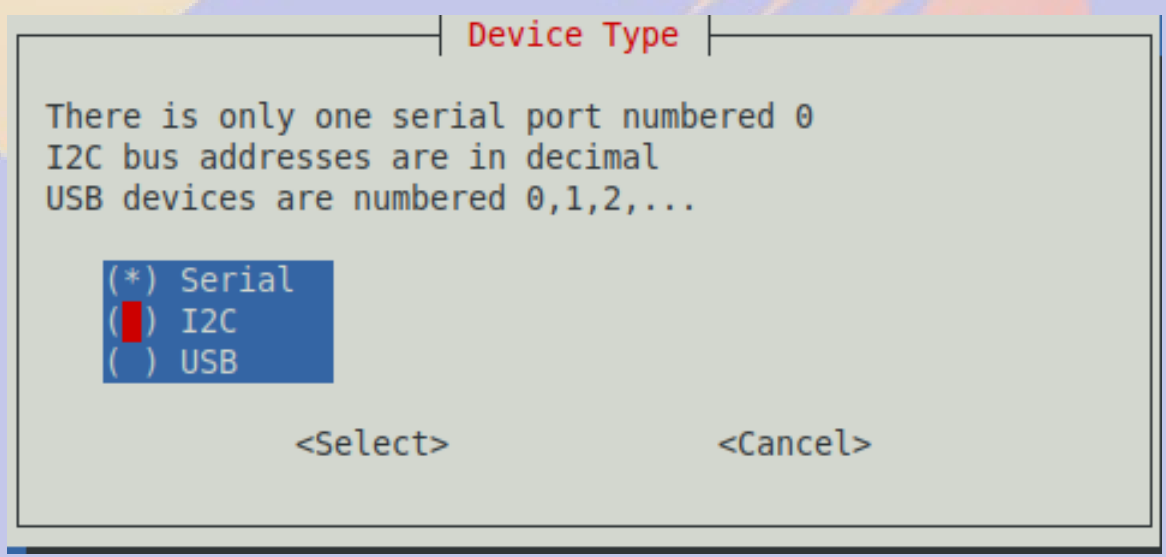


Adding more ports



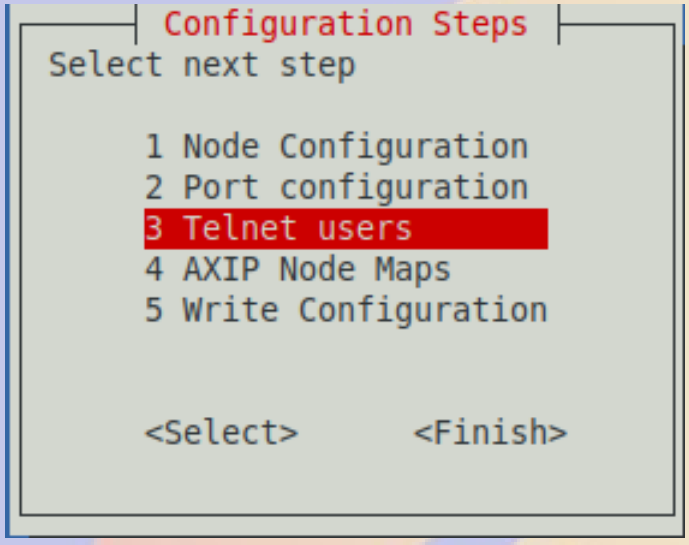


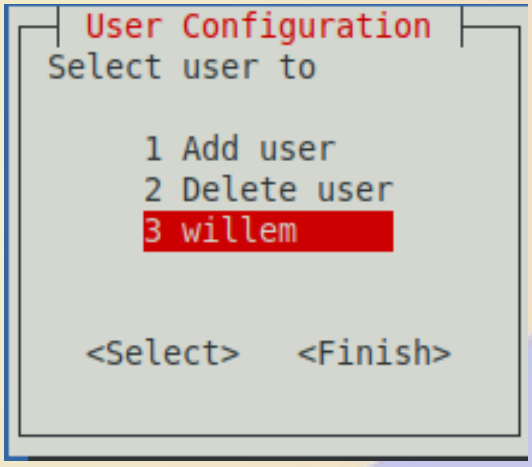


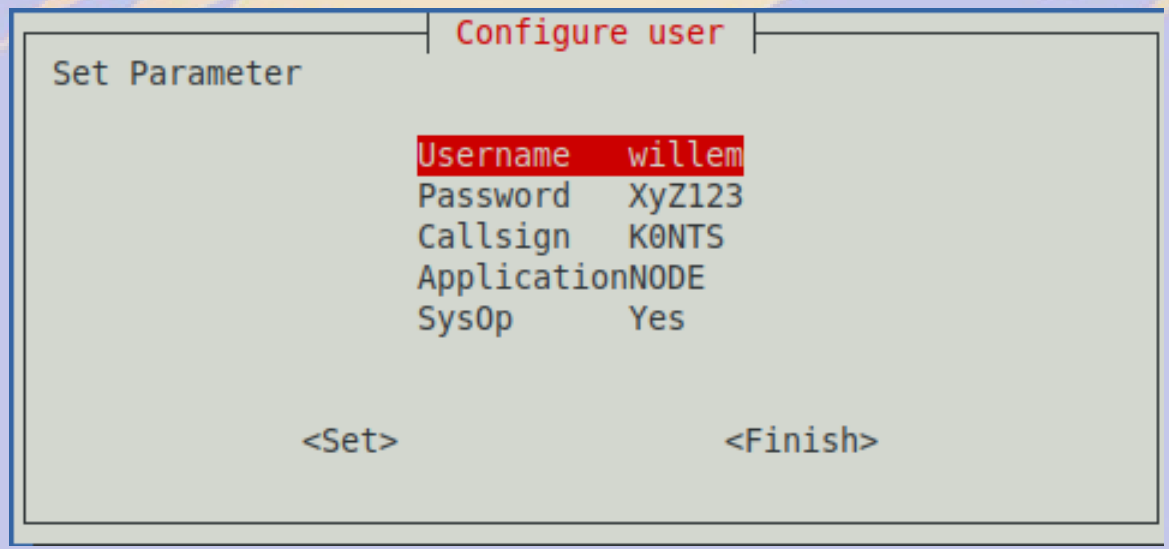


Add Telnet users

Quickstart adds a system user



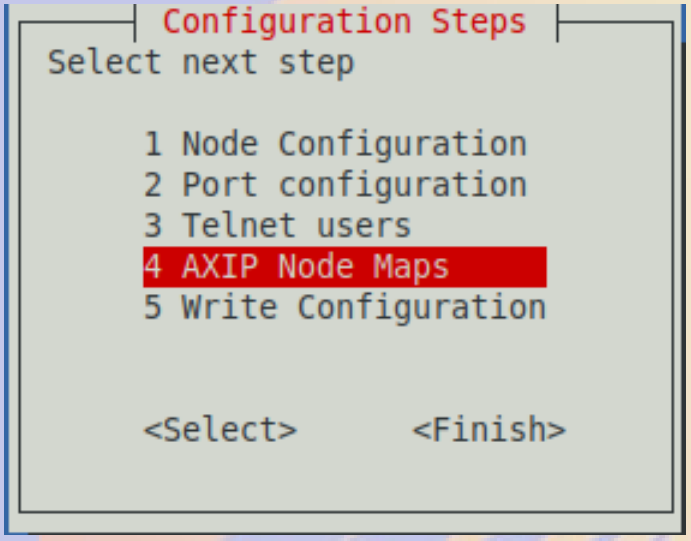


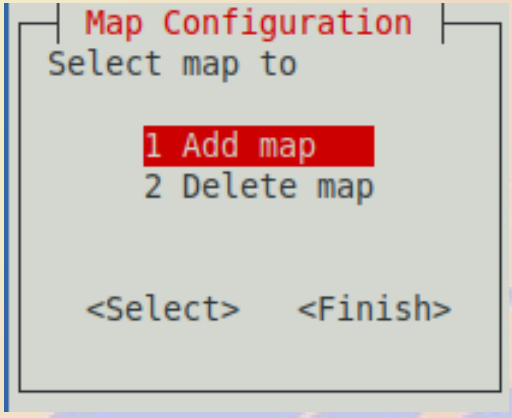


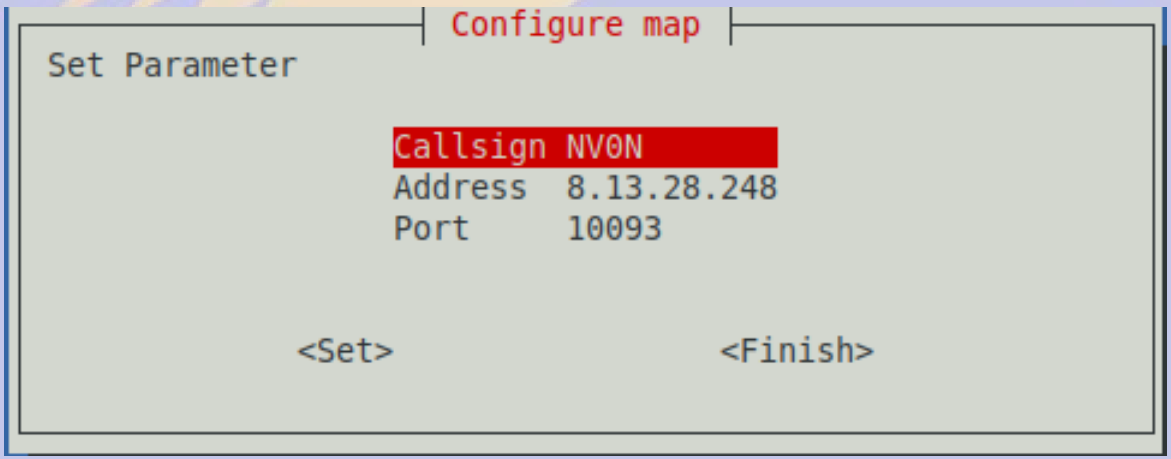
Configure AXIP Node Maps

Uses UDP packet to encapsulate packets

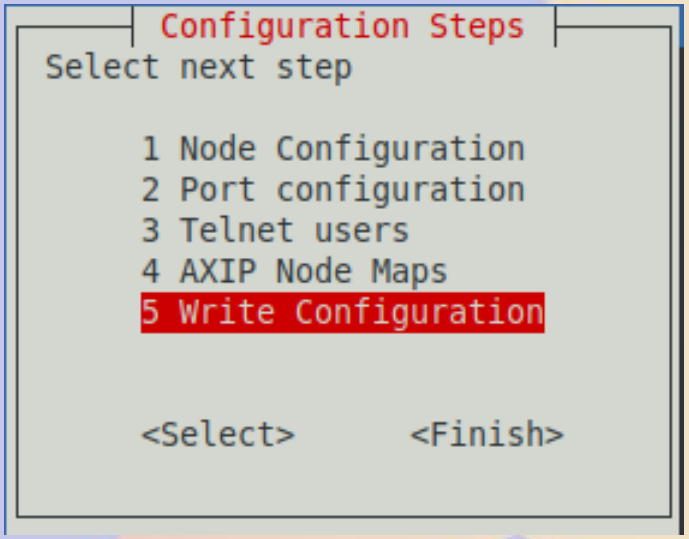
May need to allow this in your router firewall

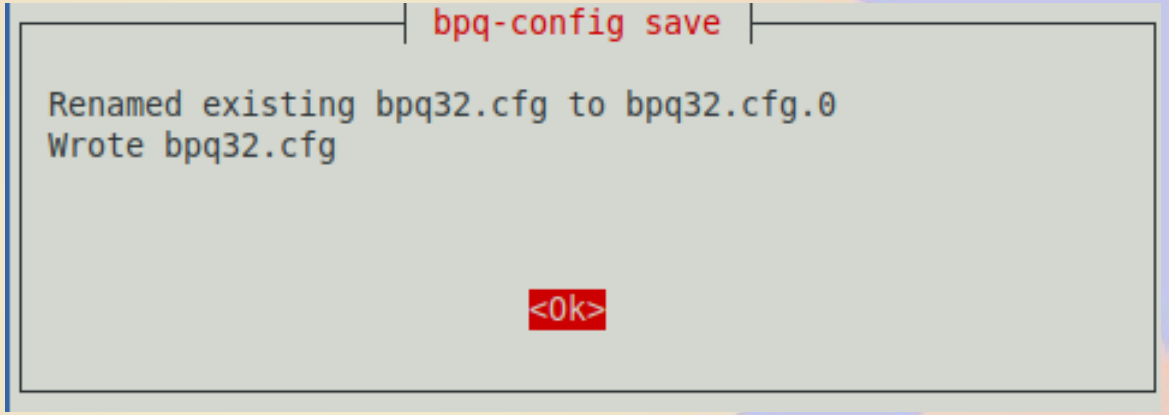


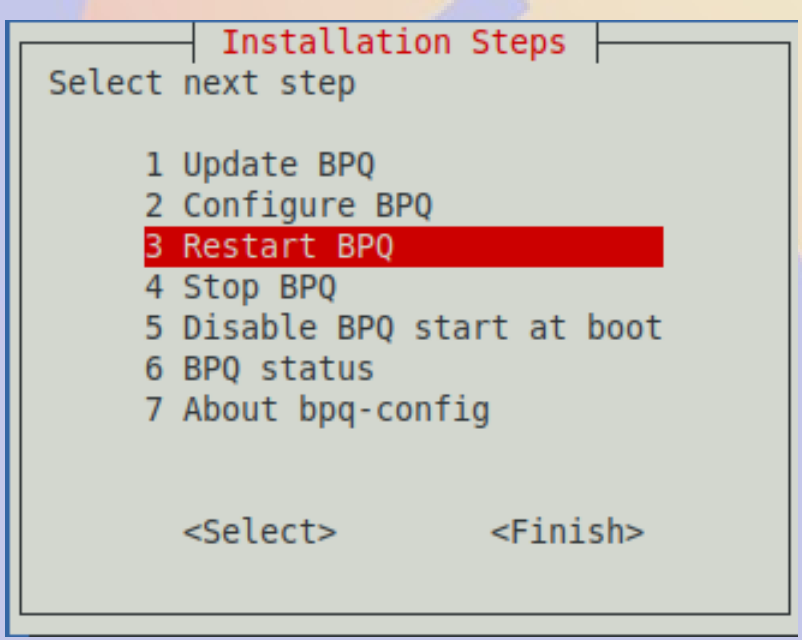


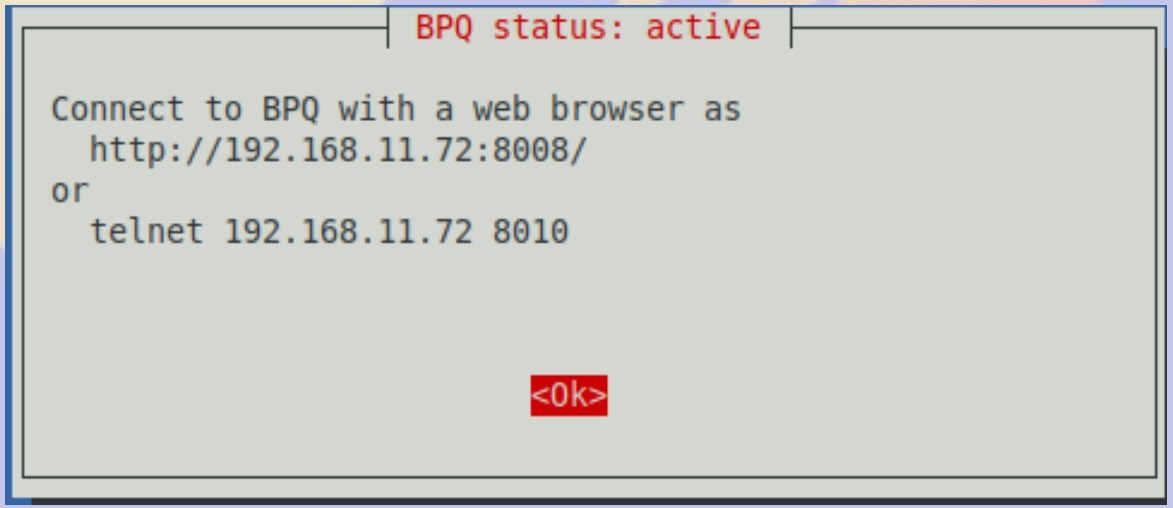


Write configuration and restart



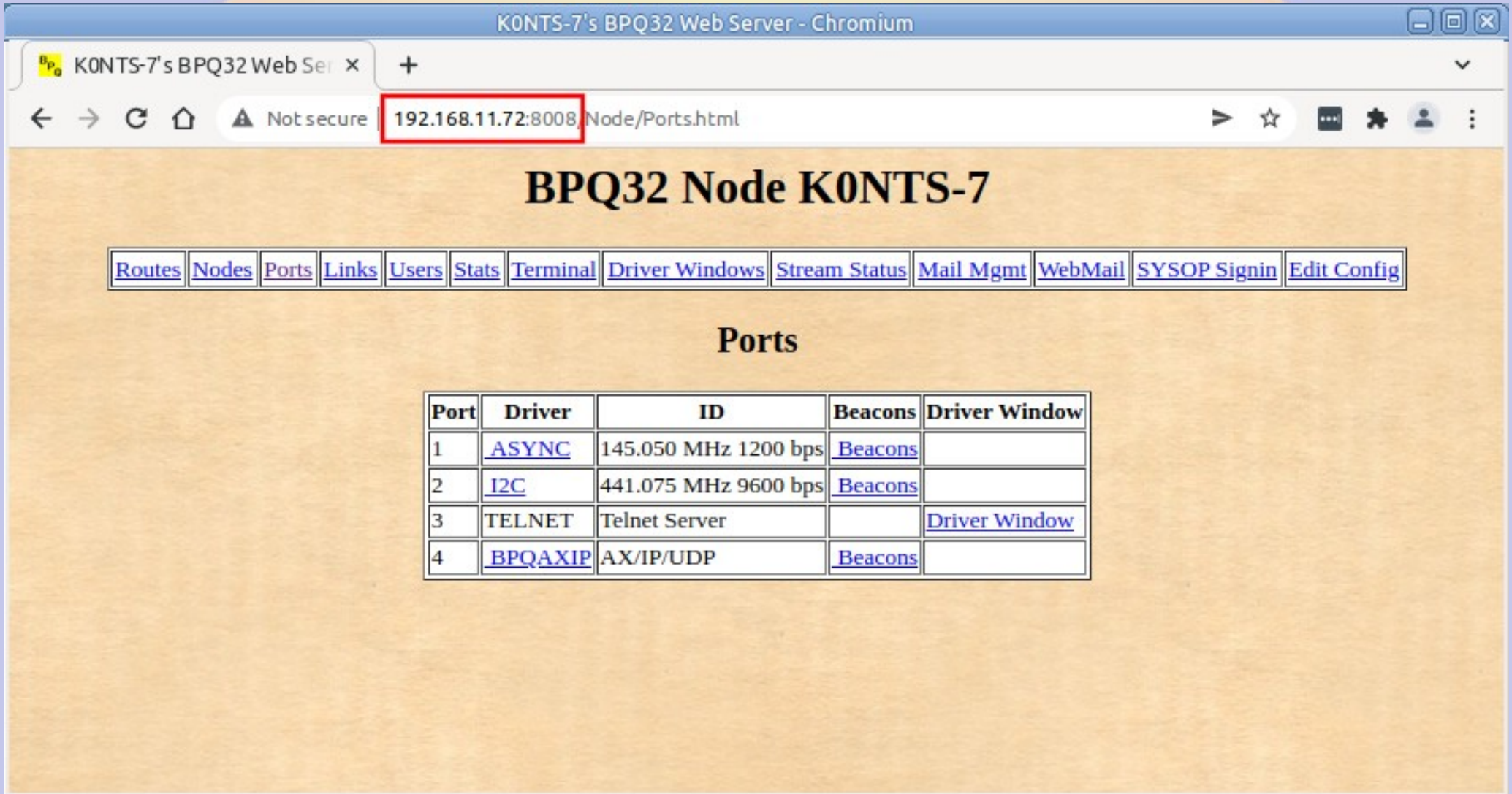




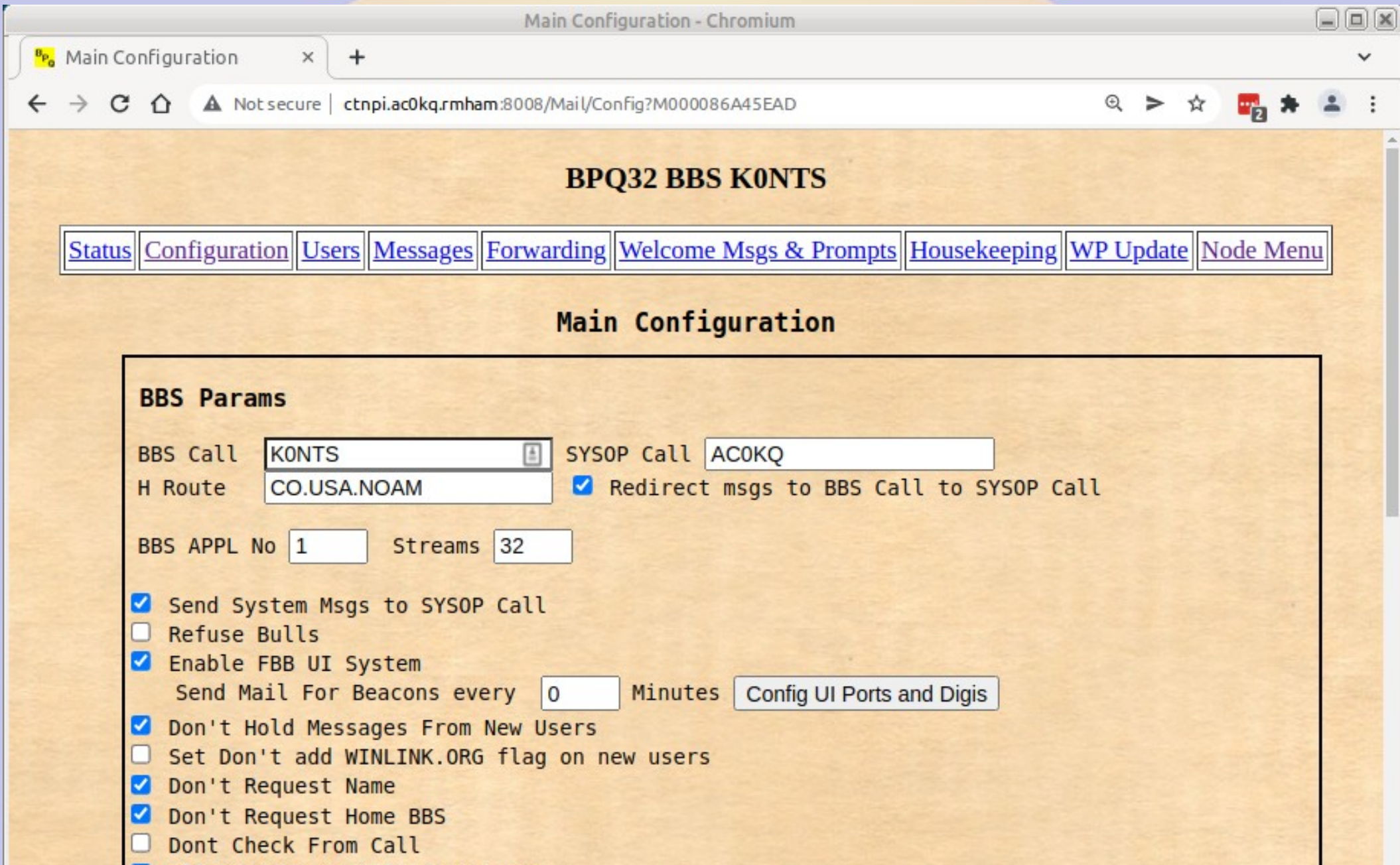


Browse to BPQ node port 8008

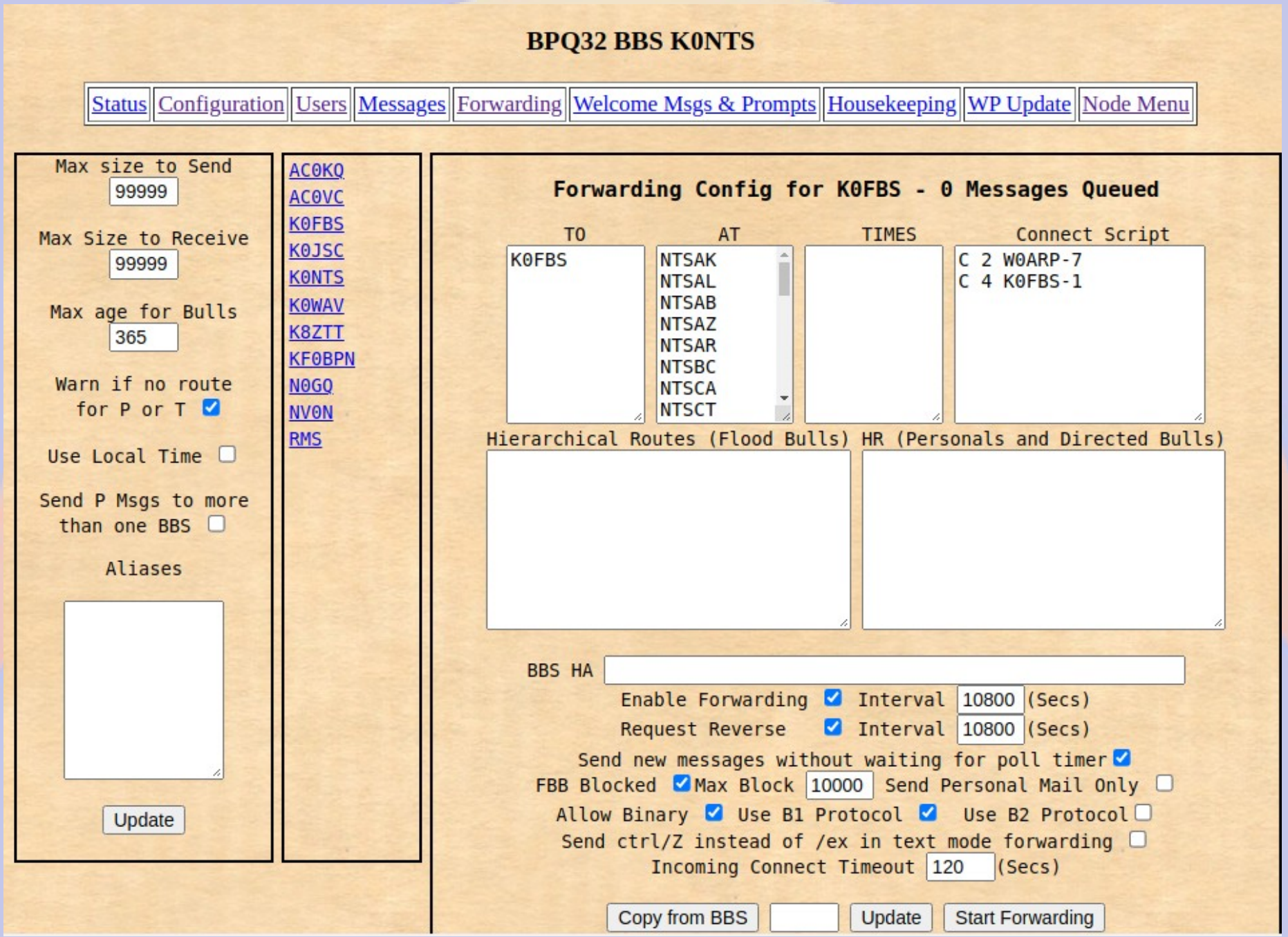
(Use the IP and port on the status page)



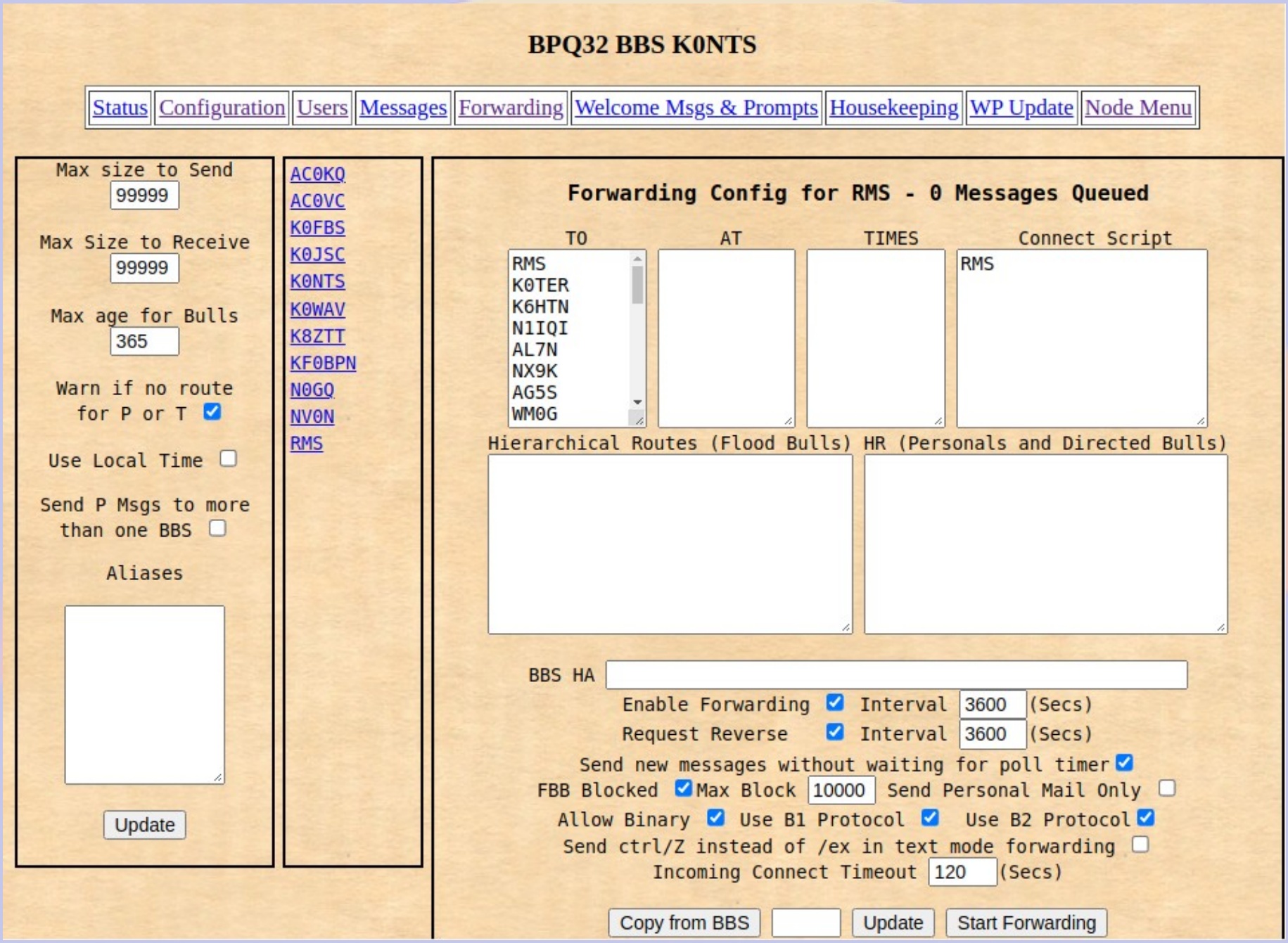
BBS Configuration
(bpq-config set most of these in linmail.cfg)



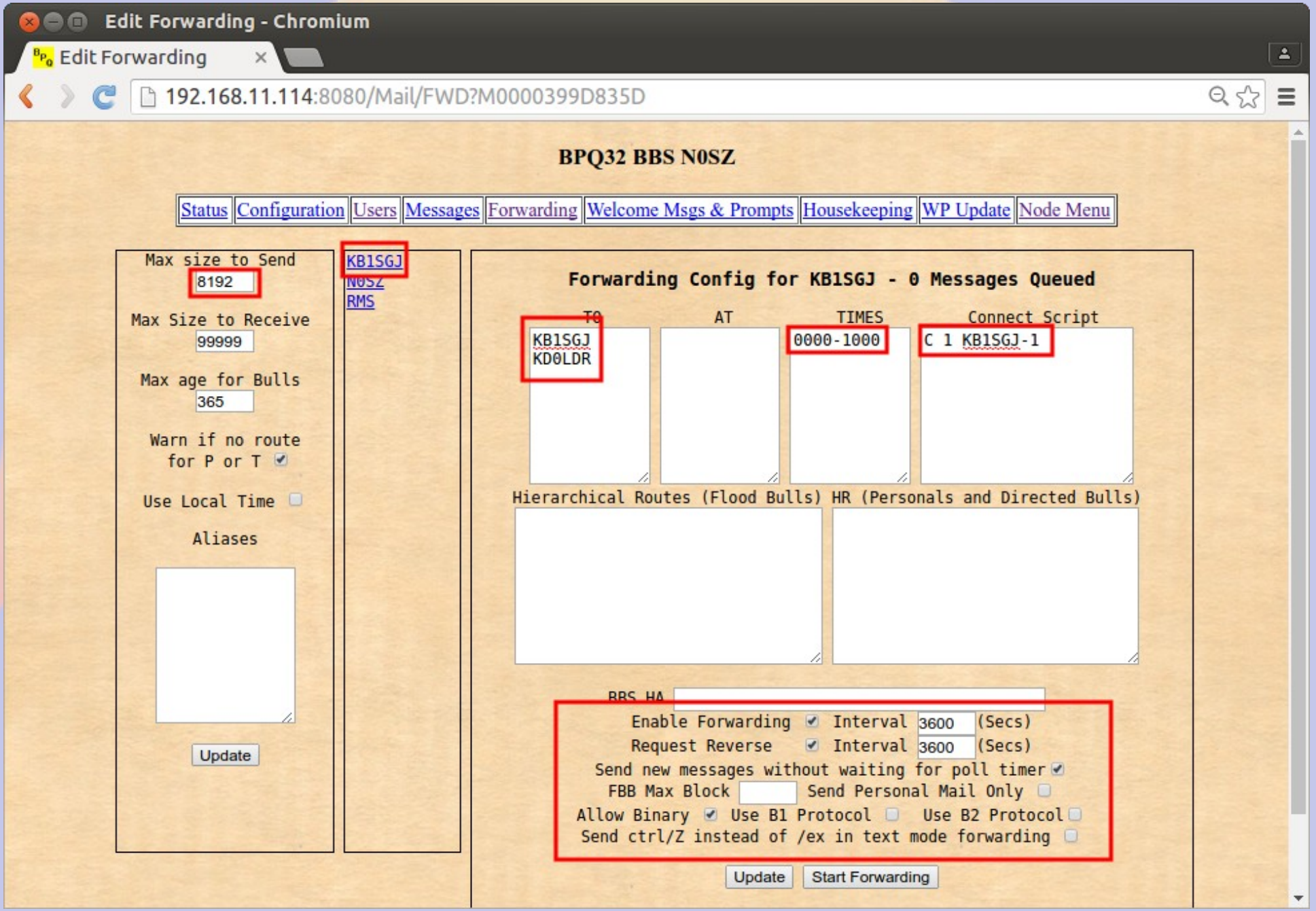
Forwarding to NTS-D



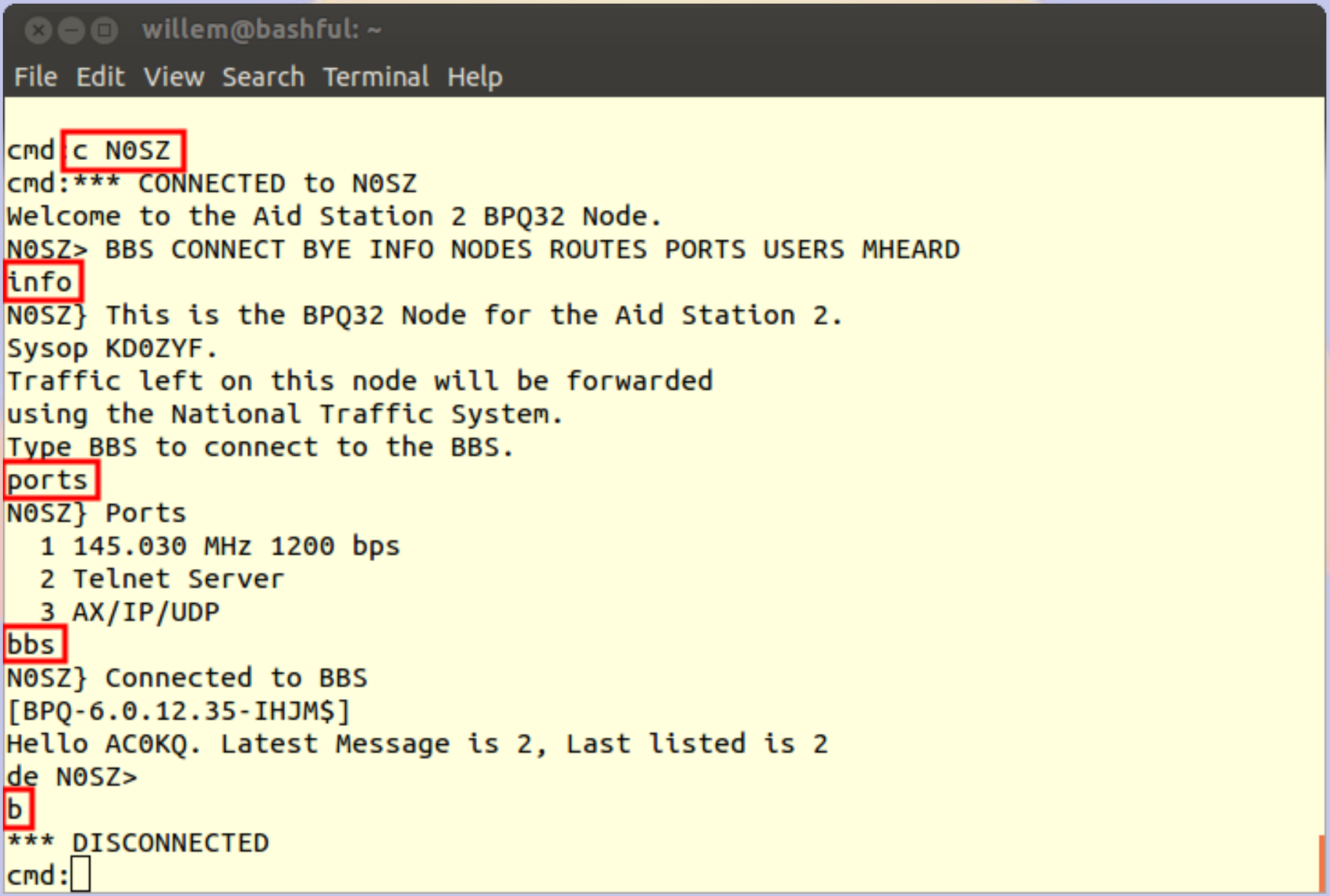
Forwarding to WinLink



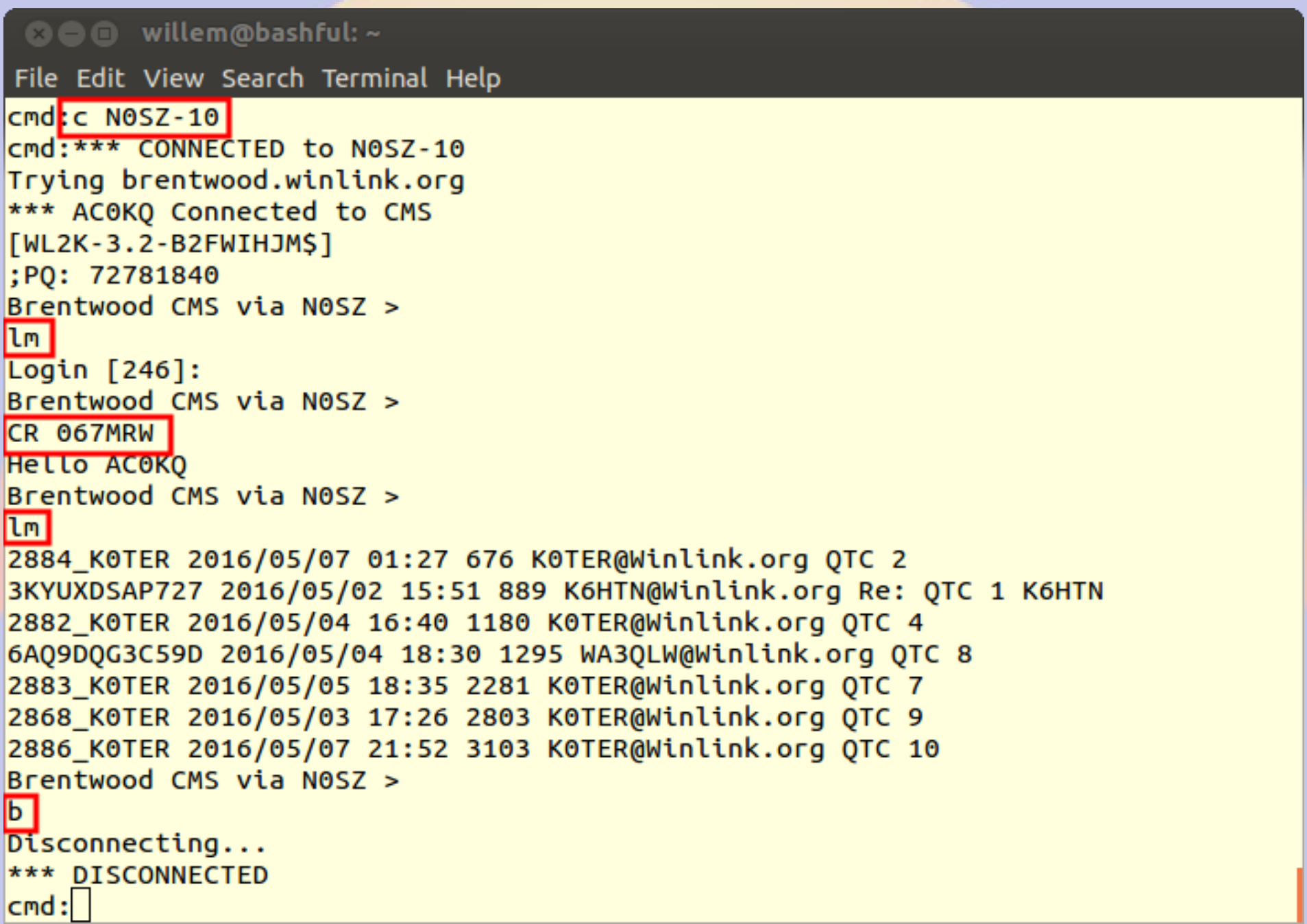
Forwarded Users



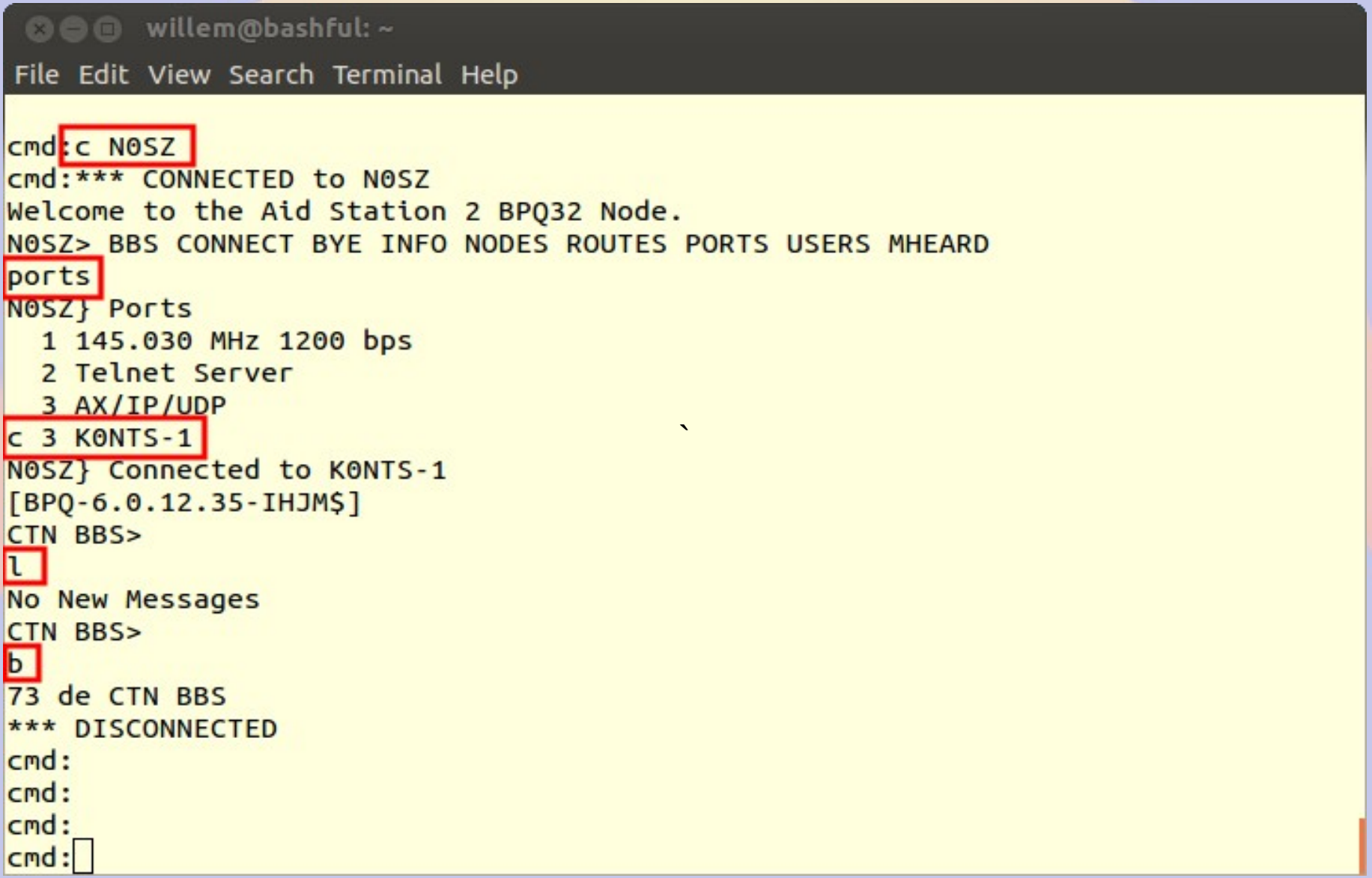
Connect in via RF (as AC0KQ)



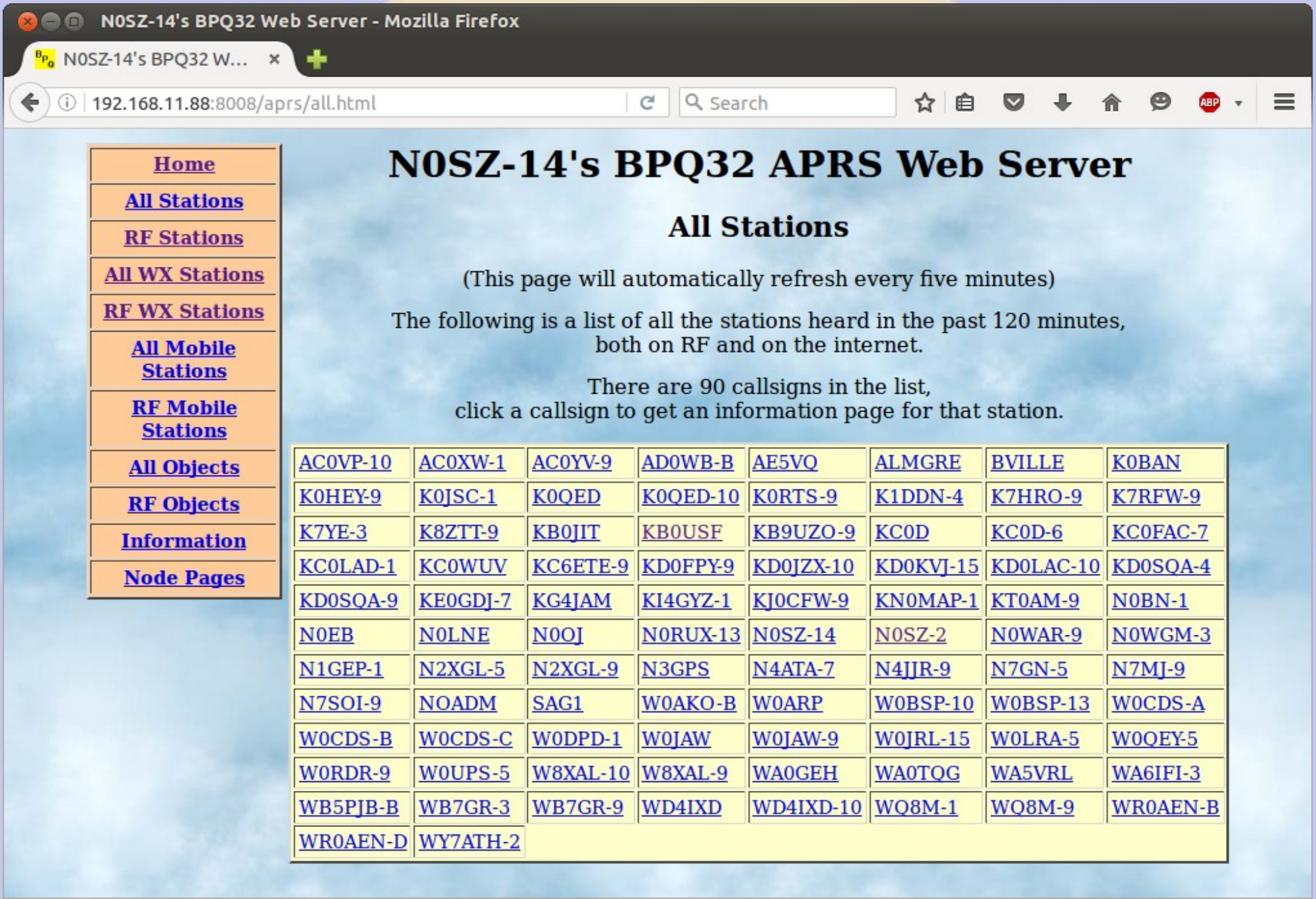
Connect via RF to WinLink



RF > N0SZ & AXIP > K0NTS-1



APRS Page


N0SZ-14's BPQ32 Web Server - Mozilla Firefox
N0SZ-14's BPQ32 W...
192.168.11.88:8008/aprs/all.html
Search
Home
All Stations
RF Stations
All WX Stations
RF WX Stations
All Mobile Stations
RF Mobile Stations
All Objects
RF Objects
Information
Node Pages
N0SZ-14's BPQ32 APRS Web Server
All Stations
(This page will automatically refresh every five minutes)
The following is a list of all the stations heard in the past 120 minutes, both on RF and on the internet.
There are 90 callsigns in the list, click a callsign to get an information page for that station.
AC0VP-10 AC0XW-1 AC0YV-9 AD0WB-B AE5VQ ALMGRE BVILLE K0BAN
K0HEY-9 K0JSC-1 K0QED K0QED-10 K0RTS-9 K1DDN-4 K7HRO-9 K7RFW-9
K7YE-3 K8ZTT-9 KB0JIT KB0USF KB9UZO-9 KC0D KC0D-6 KC0FAC-7
KC0LAD-1 KC0WUV KC6ETE-9 KD0FPY-9 KD0JZX-10 KD0KVJ-15 KD0LAC-10 KD0SQA-4
KD0SQA-9 KE0GDJ-7 KG4JAM KI4GYZ-1 KJ0CFW-9 KN0MAP-1 KT0AM-9 N0BN-1
N0EB N0LNE N0OJ N0RUX-13 N0SZ-14 N0SZ-2 N0WAR-9 N0WGM-3
N1GEP-1 N2XGL-5 N2XGL-9 N3GPS N4ATA-7 N4JJR-9 N7GN-5 N7MJ-9
N7SOI-9 NOADM SAG1 W0AKO-B W0ARP W0BSP-10 W0BSP-13 W0CDS-A
W0CDS-B W0CDS-C W0DPD-1 W0JAW W0JAW-9 W0JRL-15 W0LRA-5 W0QEY-5
W0RDR-9 W0UPS-5 W8XAL-10 W8XAL-9 WA0GEH WA0TQG WA5VRL WA6IFI-3
WB5PJB-B WB7GR-3 WB7GR-9 WD4IXD WD4IXD-10 WQ8M-1 WQ8M-9 WR0AEN-B
WR0AEN-D WY7ATH-2

Station Map

N0SZ-14's BPQ32 Web Server - Mozilla Firefox

N0SZ-14's BPQ32 W...

192.168.11.88:8008/aprs/find.cgi?call=N0SZ-2

Search

- Home
- All Stations
- RF Stations
- All WX Stations
- RF WX Stations
- All Mobile Stations
- RF Mobile Stations
- All Objects
- RF Objects
- Information
- Node Pages

N0SZ-14's BPQ32 APRS Web Server

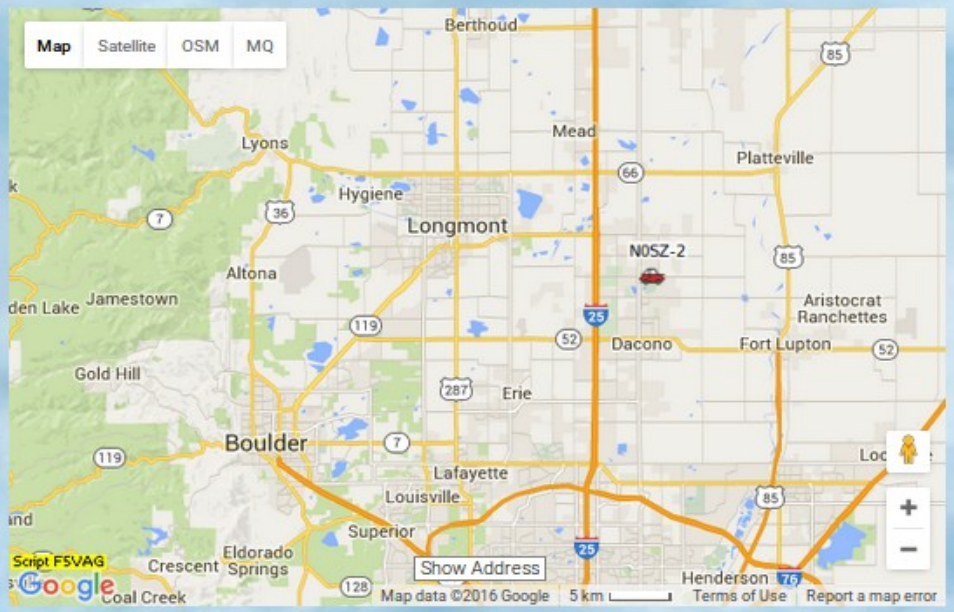
(This page will automatically refresh every five minutes)

Information about N0SZ-2

Click the callsign to look it up on qrz.com
Location: 40°07.91'N 104°55.75'W
The bearing from N0SZ-14 to N0SZ-2 is 028 degrees, the distance is 47.9 Miles

Last posit: APTT4,W0UPS-5,WIDE1,KC0D,WIDE2*

Status: /TinyTrak4 Alpha
Last heard 00:54:36 ago



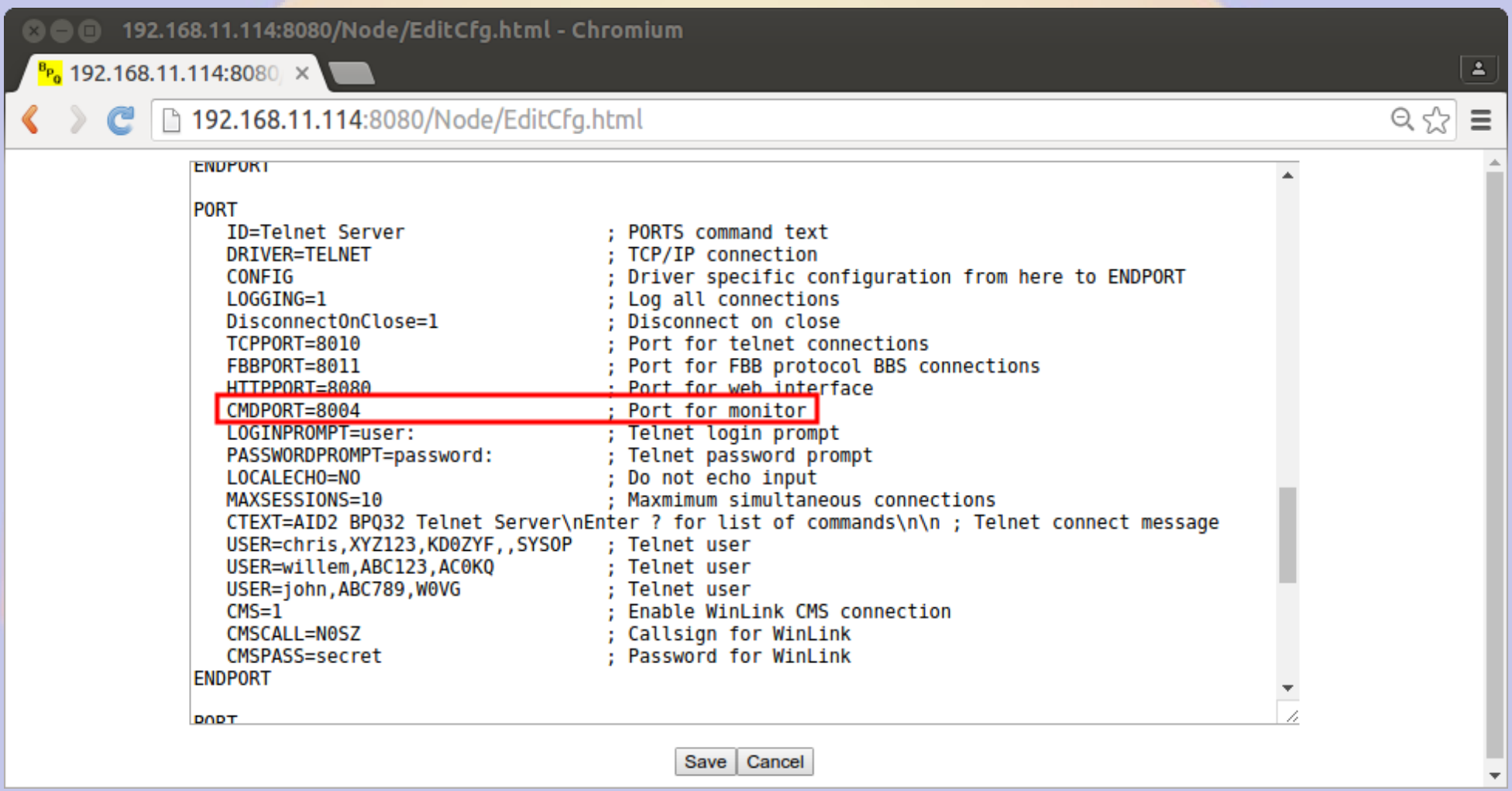
Adding a new service

- **BPQ adds new services via TCP/IP**
- **BPQ connects to local port**
- **Received stream piped to *stdin***
- **Transmits output from *stdout***
- **First line is connected station call**

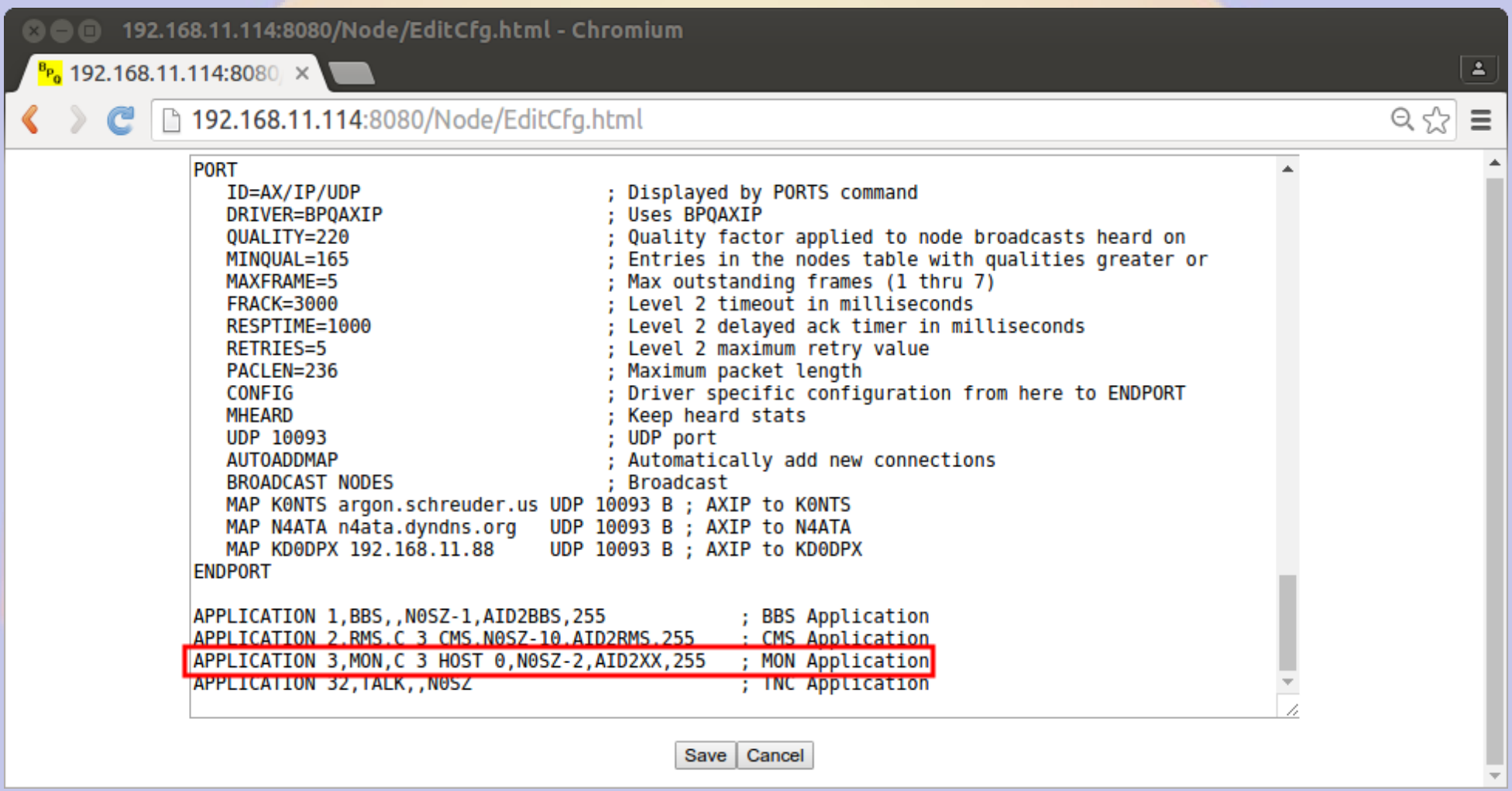
Monitor service

- Written in Perl
- Interprets commands
- Used to get system time and disk
- Bye to exit
- Mapped to N0SZ-2 and port 8004

Command Port



Application



systemd Socket Service

/lib/system/systemd/mon.socket

```
[Unit]
Description=MON
socket

[Socket]
ListenStream=8004
Accept=yes

[Install]
WantedBy=sockets.targ
et
```

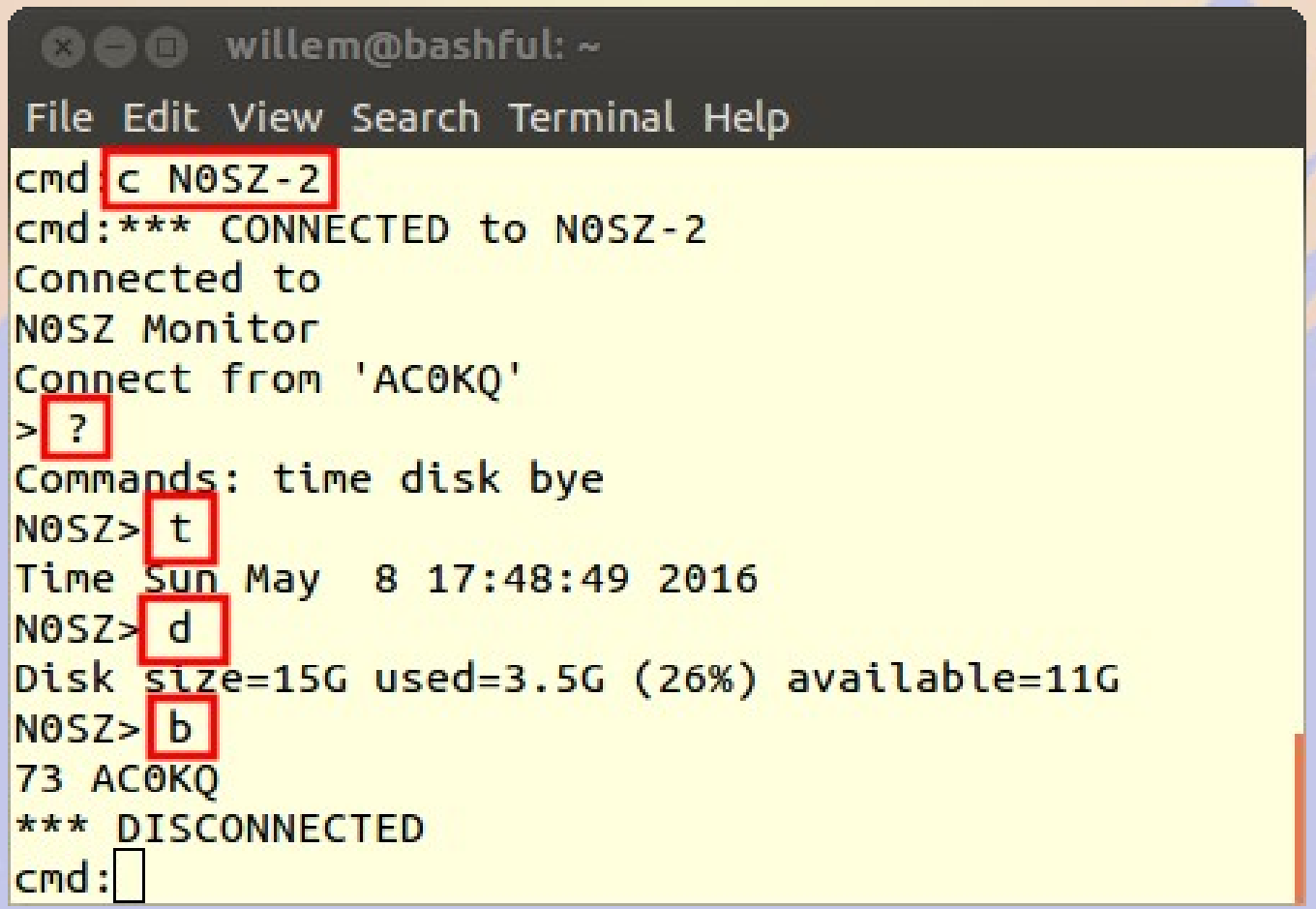
/lib/system/systemd/mon@.service

```
[Unit]
Description=MON server

[Service]
ExecStart=/home/willem/BPQ/
mon.pl
StandardInput=socket
```

mon.pl

```perl
#!/usr/bin/perl -w -T

#  Monitor server
#  Willem AC0KQ

use strict;
$ENV{PATH} = '/bin';

#  Set autoflush
my $old_fh = select(STDOUT);
$| = 1;
select($old_fh);

#  Read callsign of connecting station
my $call = <STDIN>;
$call =~ s/^\s*//;
$call =~ s/\s*$//;

#
#  Match command
#
sub match
{
   my ($cmd,$pat) = @_;
   $pat = substr($pat,0,length($cmd));
   return ($pat eq $cmd);
}

#  Welcome message
print "N0SZ Monitor\nConnect from '$call'\n> ";
```

```perl
#  Read lines
while(my $line = <STDIN>)
{
   #  Split on whitespace
   my ($cmd,@line) = split(' ',$line);
   $cmd = defined($cmd) ? lc $cmd : '';

   #  Do nothing
   if ($cmd eq '')
   {
   }
   #  Help
   elsif($cmd eq '?')
   {
      print "Commands: time disk bye\n";
   }
   #  Bye
   elsif (match($cmd,'bye'))
   {
      die "73 $call\n";
   }
   #  Print time
   elsif (match($cmd,'time'))
   {
      print "Time ".localtime(),"\n";
   }
   #  Print disk usage
   elsif (match($cmd,'disk'))
   {
      my @disk = `df -kh /`;
      my (undef,$size,$used,$avail,$pct) = split(' ',$disk[1]);
      print "Disk size=$size used=$used ($pct) available=$avail\n";
   }
   else
   # Unknown command
   {
      print "Unknown command $cmd args @line\n";
   }
   #  Print command prompt
   print  "N0SZ> ";
}
```

Enable the service

- **Enable and start the service**
  - **sudo systemctl enable mon.socket**
  - **sudo systemctl start mon.socket**