PortSlave How−To using the Linux Router

Tom McKellips

tom@computechnology.com

Revision History

Revision v1.00	8 October 2000

I wrote this as a beginning for using Portslave with the Linux router project. It seems that Portslave is a widely used program with no documentation. Since I just dove into using Portslave I am sure this document can be improved by others over time. After several days of fighting Portslave, I figured out how incredibly easy to use it really is.

Table of Contents

1. Introduction

Share this with all; sell it to no one.

First, I just want to thank all the Linux programmers out there. Your contributions have made a difference. I hope this small contribution is useful to the Linux users out there.

I wrote this as a beginning for using Portslave with the Linux router project. It seems that Portslave is a widely used program with no documentation. Since I just dove into using Portslave I am sure this document can be improved by others over time. After several days of fighting Portslave I figured out how incredibly easy to use it really is.

The trouble I ran into most of the time was PAP Authentication failure. This was really tough thing to work out (I thought). I also had a few other errors, but if you follow this How To it should at least get you started (or start you to the insane asylum).

1.1. Copyright Information

1.2. Disclaimer

No liability for the contents of this documents can be accepted. Use the concepts, examples and other content at your own risk. As this is a new edition of this document, there may be errors and inaccuracies, that may of course be damaging to your system. Proceed with caution, and although this is highly unlikely, the author does not take any responsibility for that.

All copyrights are held by their respective owners, unless specifically noted otherwise. Use of a term in this document should not be regarded as affecting the validity of any trademark or service mark.

Naming of particular products or brands should not be seen as endorsements.

You are strongly recommended to take a backup of your system before major installation and backups at regular intervals.

2. The Procedure

First, you need a running version of LRP – either build the disk yourself or you can grab an image of mine at: http://www.computechnology.com/pslave1440.img. This is a copy of my working disk and you can directly write an image of it to your disk. It will get you started. All you will then have to do is change to the appropriate IP numbers and network card drivers. I won't get into building a lrp disk here because there seems to be fairly good documentation on that subject available.

Assuming you have your disk built, and you computer running, here is what we need to do to configure your disk.

First, *DELETE* (yes, I said *DELETE*) all `options.tty??` files, the options file, and pap–secrets files located under `/etc/ppp-radius` and `/etc/ppp`. If you have an `/etc/ppp`, you probably installed `ppp.lrp`. Also remove `ppp.lrp` from your disk and `/etc/ppp` will go away.

Next, go to `/etc/portslave` and adjust the `pslave.conf` file accordingly. I will now take you through that file line–by–line. I don't know what all of it means but I made it work so you can to.

```
#
# pslave.conf Here is the sample server configuration file.
#
# Version: 1.17 03-Nov-1998
#

#
# Hostname of the system.
#
# This is my routers name. Your routers name will be different

conf.hostname hma2.cpty.net

#
# IP address - if left empty, uses the IP address of the system (hostname).
#
# This is used as the "local" address for SLIP and PPP connections.
# This is my routers IP address yours will be different use your routers
# IP number here

conf.ipno 10.0.0.4

#
# Lock directory - on FSSTND compliant systems it's /var/lock.
#
# No need to change this

conf.lockdir /var/lock

#
# Where to find the rlogin binary that accepts the "-i" flag.
#
# No need to change this

conf.rlogin /usr/bin/rlogin-radius

#
# Where to find our patched pppd that has radius linked in.
#
```

```
# No need to change this

conf.pppd /usr/sbin/pppd-radius

#
# Where to find telnet. This can just be the system telnet.
#
# This can stay or go

conf.telnet /usr/bin/telnet

#
# If you set this to "1", you can always login locally by putting a '!'
# before your loginname. Useful for emergencies when the RADIUS server is down.
# Make this either 0 or 1 as mentioned above

conf.locallogins 1

#
# Logging stuff - this program can use a remote syslog daemon if needed.
#
# If you want to log locally leave the "syslog" field empty. The facility
# field is an integer between 0 and 7 and sets the syslog facility to
# local0-local7.
#
# For now I log local to my router that is why I do not have anything
# after syslog

conf.syslog

conf.facility 6

#
# Stripnames - if you set this to "1", leading "P", "S", "C", "L" or "!"
# characters and trailing ".slip", ".cslip" and ".ppp" strings will be
# stripped from the username before it is recorded in the system
# utmp and wtmp files (if sysutmp or syswtmp are turned on of course)
#
# No need to change this

conf.stripnames 0


##
## The all entry is used as a template for all others. This means that
## setting all.debug to 0, you set s0.debug, s1.debug, s2.debug etc.
## to 0. It also means that all these settings can be overridden on a
## per-port basis below.
##
## The "all." stuff is the default for everything unless you specifically
## override it. I'll show you that at the end of this.


#
# Debugging output to syslog. Set to 0 or 1. "1" is pretty verbose.
# This can be 0 or 1. I like 1 because it gives lots of info

all.debug 1

#
# Authentication type - either "radius" or "none".
#
```

```
# Leave this as it is

all.authtype radius

#
# Authentication host and accounting host. We can have 2 of both. The
# first one is always tried three times before switching to the second one.
# They are alternately tried after that, up to maximum 10 times in total.
# Timeout is 5 seconds per query.
#
# These are the names of my RADIUS servers; name your RADIUS servers here

all.authhost1 cody.cpty.net
all.accthost1 cody.cpty.net

#all.authhost2 backuphost.someisp.com
#all.accthost2 backuphost.someisp.com

#
#
# The shared secret for RADIUS.
#
# Put your shared secret here, this must match the shared secret in
# your RADIUS servers clients file for the IP number or name of this router.

all.secret superagentman

#
# Default protocol and host. This is for rlogin sessions.
#
# Just change the all.host to the IP number of your router this should
# match what you have at the top of this file

all.protocol rlogin
all.host 10.0.0.4

#
# Default IP stuff. If you end the "ipno" with a "+", the portnumber will
# be added to the IP number. The IP number of a port is used when the RADIUS
# server doesn't send an IP number, or if it tells us to use a dynamic ipno.
#
# Leave the netmask at 255.255.255.255, unless you really know what
# you're doing.
#
# This seemed a little confusing, but since I went with static IP numbers
# this was easy. I do not have "+" after my IP number because I directly
# assign the IP number to a MODEM at the end of this file.
#
# I modified the netmask to match that of my network. And I left MTU alone

all.ipno 10.0.0.4
all.netmask 255.255.255.0
all.mtu 1500

#
# Standard message that is issued on connect.
#
# No need to change this

all.issue \n\

Cistron Internet Services \n\
```

```
POP Alphen aan den Rijn \n\

Welcome to terminal server %h port S%p\n

#
# Login prompt.
#
# No Need to change this

all.prompt Cistron login:

#
# Terminal type, for rlogin/telnet sessions.
#
# No need to change this

all.term vt100

#
# If you want portslave to update the utmp and/or wtmp files just
# like a regular getty/login, set these to 1.
#
# I set both of these to 1; you can do what you want here

all.sysutmp 1

all.syswtmp 1


##
## Options for the serial port.
##

#
# Porttype (passed to Radius for logging).
# 0 = async, 1 = sync, 2 = ISDN, 3 = ISDN-V120, 4 = ISDN-V110
#
# Use 0 if its a modem

all.porttype 0

#
# Speed. All ports are set to 8N1.
#
# Set this to just beyond your max modem speed

all.speed 115200

#
# Use this to initialize the modem.
#
# I had to make this AT38;F to reset the modem to it's defaults
# each time ATZ did not work.  If you are not familiar with scripting
# then some of this won't make sense to you. That's ok; I didn't get u
# it either until I played with it.

all.initchat "" \d\dAT38;F OK\r\n-AT38;F-OK\r\n

#
# You can use either waitfor or aa.
#
```

```
# No need to change this

all.waitfor RING

#
# Chat string to get the modem to connect after waitfor.
#
# The @ sign matches (.*)[\r\n] in regexp code, the match is logged
# to Radius as Connection-Info.
#
# No need to change this

all.answer "" ATA CONNECT@

#
# Auto answer - if you set this to "1", the system will just wait for
# the DCD line to get high (this is not well tested). You won't get
# the connection info either.
#
# No need to change this

all.aa 0

#
# You can use this chatstring to regulary check if the modem is still alive.
#
# NOT IMPLEMENTED YET.
#
# I don't know if this line works or not. Let me know if you find anything
# out about it.  Just leave it the same and portslave will work.

all.checktime 60
all.checkchat "" AT OK\r\n

#
# Flow control on this serial port:
#
#    hard - hardware, rts/cts
#    soft - software, CTRL-S / CTRL-Q
#    none
#
# No need to change this

all.flow hard

#
# Use the DCD line or not (this sets CLOCAL if on). This means that the
# session will get hung up if the modem hangs up. Can be set to 0 or 1.
#
# No need to change this

all.dcd 1

#
# PPP options - used if we autodetect a PPP session.
#
# Note that we set mru and mtu both to the MTU setting.
#
# Look at these lines close; this is what worked for me.
# These parameters are sent to the ppp daemon when it
# is called. I think the autoppp is called first then
# after you are authenticated I think the second ppp is
```

```
# called. I don't know for sure that this is how it worked,
# but it appears that way to me

all.autoppp proxyarp modem asyncmap 0 %i: \

noipx noccp login auth +pap -chap \

mtu %t mru %t \

ms-dns 208.206.143.35 ms-dns 208.206.143.36 \

uselib /usr/lib/libpsr.so


#
# PPP options - User already authenticated and service type is PPP.
#

all.pppopt proxyarp modem asyncmap 0 %i:%j \

noipx noccp \

mtu %t mru %t netmask %m idle %I \

ms-dns 208.206.143.35 ms-dns 208.206.143.36 \

uselib /usr/lib/libpsr.so


##
## Tty names are s0...s63. For every port we need to define a tty port, and
## an IP number for when radius tells us to pick one ourself. Unless you
## use the IP pool option mentioned above (IP number with "+" appended).
##
## Note that you can change _all_ of the above settings that start
## with all.xxxx on a per-port basis, such as issue, prompt etc.
##
## This is where you can set options to a specific modem. sX.tty is
## for portslaves use you assign it to a real tty device.
## In the /etc/inittab you will see the lines portlsave 0 or 1 etc..
## this 0 or 1 is the tty device number portslave already knows it is
## "tty something" so all it requires is the last digits.
##
## Since my modem is on COM 4 (DOS) that means s3.tty is ttyS3

s3.tty ttyS3

# Now I can set options for that modem
#
# Here is it's IP number statically assigned

s3.ipno 10.0.0.202

# Here is the protocol to use on that modem. PAY CLOSE ATTENTION HERE!
# This is the line that finally made portslave work perfectly for me.
# You must tie the ppp protocol to your modem. Simple huh?

s3.protocol ppp13;
```