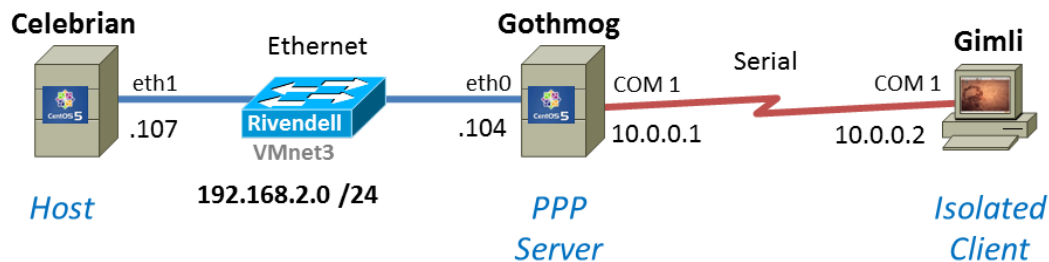


## CIS 192 Linux Lab Exercise

### Lab X2: PPP Connections (Extra Credit) Fall 2011 2010

#### Lab X2: PPP Connections (Extra Credit)

The purpose of this lab is to connect a standalone computer to a LAN using a serial connection with PPP. Whereas the serial connection is usually established with analog modems, in this lab, you will establish the serial connection with a virtual null-modem cable connecting the serial ports of two computers. You will first create the PPP link by hand, and then automate the connection with PPP scripts. In the following discussion, Gimli will be the isolated client, Gothmog will refer to the computer acting as the PPP server, and Celebrian will refer to another system on the Rivendell network.



#### Supplies

- 192 VMs: Celebrian, Gothmog\* and Gimli\*

\*Only available in CIS VLab pods.

#### Preparation

- Revert to the “Pristine” snapshot on all four VMs.
- On Gimli, install:
  - minicom
- On Opus, make a copy of the labX2 report template file in `/home/cis192/depot` in your home directory. Edit the header of this file with your own information and record all the information requested.

#### Forum

Use the forum to ask for help, post tips and any lessons learned when you have finished. Forum is at: <http://opus.cabrillo.edu/forum/viewforum.php?f=39>

#### Background

PPP has replaced SLIP as the preferred method for establishing serial connections between computers to emulate a network connection. Operating at the network interface layer of the TCP/IP stack, PPP relies on an external component for establishing the serial link and then manages that connection to act as an IP connection so that network applications will work transparently across the link. At first you will use minicom, a small communications utility, to establish a serial connection. Second, you will use the chat utility with PPP scripts to both establish the connection and start the interface. PPP is a peer-to-peer protocol that requires the pppd daemon to be running on both sides of the connection. The client is simply the computer that initiates the connection; the server responds to it. Once a connection is established, you must perform any necessary routing changes to make use of the new interface.

The new commands we will be using for this lab are:

- pppd
- minicom
- chat

Configuring the pppd daemon requires root access since PPP modifies network settings. Note that although you will be using three machines, you don't need the CIS Lab network.

## Procedure

### Part 1

In this part of the lab, the VMs will be cabled according to the diagram above. Celebrian and Gothmog will be connected over the usual Ethernet LAN. Gothmog will be connected to Gimli however with a serial cable. With physical computers, this serial cable could be extended using dial-up modems. For this lab, the serial connection is made using a virtual null-modem cable plugged into the (COM 1) port of each computer. To implement this "cable" in VMware, a "named pipe" will be used.

1. Revert to the "pristine" snapshot of both Gimli and Gothmog.
2. **On Gothmog**, use the VMware Hardware Wizard to add a serial port and specify the "Output/connect to named pipe" option:
  - a. Name the Pipe "**null-modem-cable**"
  - b. Near end should be "**Server**"
  - c. Far end should be "**A virtual machine**"
  - d. Take the defaults for everything else.
3. **On Gimli**, use the VMware Hardware Wizard to add a serial port and specify the "Output/connect to named pipe" option.
  - a. Name the Pipe "**null-modem-cable**"
  - b. Near end should be "**Client**"
  - c. Far end should be "**A virtual machine**"
  - d. Take the defaults for everything else.
4. When you have virtually "connected the two computers with the null-modem cable", power up Gothmog and logon as root.
5. Verify that the serial port exists and is active:  
**ls -l /dev/ttyS?**  
**setserial /dev/ttyS0**  
Note that the COM1 serial port is accessed through the device file /dev/ttyS0.

Verify that the serial chip is a UART 16450A or 16550A which can support baud rates in excess of 14400.

6. Enable this port for logins by editing `/etc/inittab`. Add the following line near the bottom of the file:

**s1:35:respawn:/sbin/agetty 38400 ttyS0**

7. Tell init to re-read the inittab file:

**telinit q**

8. Allow the `pppd` daemon to be executed by normal users by setting the user id bit:

**chmod u+s /usr/sbin/pppd**

Verify the "s" bit is set (r\_sr\_xr\_x) with:

**ls -l /usr/sbin/pppd**

## Part 2

Bring up the client Gimli and manually connect to the server Gothmog using `minicom`.

1. Log in to Gimli as root.
2. Disable NetworkManager on Gimli with: **service network-manager stop**
3. Disconnect Gimli from the CIS Lab network and use **ifconfig eth0 down** to disable the interface.
4. Verify that `eth0` is down using the **ifconfig** command.
5. You are now going to configure the `minicom` utility which runs in character mode using the `curses` library. Configure the `minicom` utility to use the COM1 (`/dev/ttyS0`) at 38400 baud. The default 8 data bits, no parity, one stop bit, and hardware flow control is already configured by default.

**minicom -s**

Use letters, ESC, up and down arrows to navigate this utility until your configuration matches what is show below:

```
+-----+
| A -   Serial Device       : /dev/ttyS0
| B - Lockfile Location    : /var/lock
| C -   Callin Program     :
| D - Callout Program      :
| E -   Bps/Par/Bits       : 38400 8N1
| F - Hardware Flow Control : Yes
| G - Software Flow Control : No
|
|   Change which setting?
+-----+
| Screen and keyboard
| Save setup as dfl
| Save setup as..
| Exit
| Exit from Minicom
+-----+
```

Use Esc key to exit the submenu.

Select **Save setup as dfl** (for default). Note this saves the config file in `/etc/minicom/minirc.dfl`.

Select **Exit from Minicom**

6. Connect to the server and log in as user `cis192`:  
**minicom -o**

Note: after minicom comes up, press the enter key once or twice until you get the login prompt.

(The -o option forgoes the modem initialization sequence)

7. Verify that you are on the server using the **hostname** command.
8. Logout from the cis192 account on the server.
9. Exit the minicom utility with the following key sequence:  
**Ctrl-A z q**  
and confirm your decision to **Leave without reset** as Yes.

### Part 3

You will now create and configure a guest account on the server Gothmog to start the pppd daemon automatically when it is logged into. The client Gimli will then initiate the login session and start PPP on the client side thereby establishing a network session.

1. Create a guest account on **Gothmog** using:  
**useradd -c "Guest account for serial access" guest  
passwd guest**
2. Edit `.bash_profile` in guest's home directory by appending the following command line to the bottom of the file:  
**/usr/sbin/pppd -detach crtscts proxyarp 10.0.0.1:10.0.0.2 /dev/ttyS0 38400**
2. On the client Gimli, use the minicom utility (with the -o option) to **login as guest** to **Gothmog**. When you log in, you should see some "garbage" characters spewed out to the screen. This is pppd talking to you.
3. Quit the minicom session with the following keystrokes:  
**Ctrl-A z q**  
and **quickly** type the following command at the prompt:  
**pppd -detach crtscts /dev/ttyS0 38400 &**
4. You should get a connection, assuming you didn't take too long to type the above command.
5. Use **ifconfig** to verify that you have a connection. You should see the ppp0 interface. Note the server IP address and the client IP address. **Ping** the server Gothmog from the client Gimli.
6. Configure the Ethernet interfaces on **Celebrian** and **Gothmog** as shown in the map above.
7. Set up the server Gothmog as a router by turning on IP forwarding.  
**echo 1 > /proc/sys/net/ipv4/ip\_forward**
8. Set up the routing tables so that client Gimli can ping the host Celebrian on the 192.168.2.0 network. Save the output of a successful ping as part of your lab submission. Whose routing table did you have to configure?
9. Terminate the PPP connection from the client Gimli by killing the pppd process:  
**killall pppd**

### Part 4

Now that you have successfully set up a PPP connection manually in Part III, you will configure ppp scripts to automate the whole process, including the configuration of the routing table.

1. On the server Gothmog create a special account for ppp connections:  
**useradd -c "PPP Account" -d /etc/ppp -s /usr/sbin/pppd ppp**  
Note: we are making the pppd daemon the shell for this account!

2. Set the password on this account to be **"secret"**  
**passwd ppp**
3. Options for the pppd daemon can be specified in the **/etc/ppp/options** file. Make sure the following options are specified in that file, one option per line:
  - detach**
  - crtscts**
  - lock**
  - proxyarp**
  - 10.0.0.1:10.0.0.2**
  - /dev/ttyS0**
  - 38400**
4. On the client Gimli, in root's directory, write a script called **ppp-on** that contains the following line:
  - pppd updetach crtscts defaultroute /dev/ttyS0 38400 connect \**  
**"exec chat -v TIMEOUT 3 ogin:--ogin: ppp assword: secret"**
 Note: to type one command on two lines you have to escape the <cr>
5. Make this script executable:
  - chmod 755 ppp-on**
6. Now run the script and verify the ppp connection is up:
  - ./ppp-on**
  - ifconfig**
7. Make sure your network is good by pinging from Gimli to the host Celebrian.
8. Note that the default route was set up for you. Why?
9. When you are all done collecting your data, bring down the pppd interface with the following command from the client: **killall pppd**

### To turn in

Record the following in your labx2 report:

On Gimli:

- **ifconfig** output
- **route -n** output
- **ping -c3 192.168.2.107** output
- **cat ppp-on** output

On Gothmog:

- **cat /etc/ppp/options** output
- **cat /etc/passwd | grep ppp** output

Check your work for completeness then submit as many times as you wish up until the due date deadline.

**cp labx2 /home/rsimms/turnin/labx2.\$LOGNAME**

### Grading rubric (30 points)

- 5 points for correct submittal and 100% complete report header
- 5 points for correct ifconfig and route -n output on Gimli
- 5 points for correct ping from Gimli to Celebrian
- 5 points for correct ppp-on script
- 5 points for correct /etc/ppp/options file
- 5 points for the correct ppp entry in /etc/passwd on Gothmog.

