;login:

THE MAGAZINE OF USENIX & SAGE

July 2001 • Volume 26 • Number 4

inside:

PALM PILOTS AND THE POINT-TO-POINT PROTOCOL

by James Caple

USENIX & SAGE

The Advanced Computing Systems Association &
The System Administrators Guild

palm pilots and the point-to-point protocol

Sun Microsystems is right: "The Network Is the Computer"™. Chances are, if you use a computing machine of any sort, you have a need to make it network-aware at some point. Personal Digital Assistants (PDAs) are great candidates for both wired and wireless networking, extending the functionality of your networked desktop workstation to your pocket. This article discusses a simple set of steps for establishing a persistent connection between your Palm Pilot and your Linux workstation using Palm's Hot Sync cradle and the Point-to-Point Protocol (PPP).

by James Caple

James Caple has eight years of combined industry experience in proposal development, hardware hacking, UNIX systems administration, and most recently, Java and C programming.

jcaple@patriot.net

Note: Please refer to RFC1661 for further details regarding the PPP.

So What?

Why would you want to do this, you ask? Well, I am not sure I can answer that for you, but here are some reasons that might help:

1. You are writing a networked Palm application that you want to test on a real device, but you don't want to pay for a wireless modem just to test your application.
2. You need to telnet to a remote workstation and all you have with you is your Palm Pilot (it could happen . . .).
3. You want to set up a kiosk consisting of nothing but Palm cradles and a Linux box where you want users to be able to browse the Web using only their Palm Pilot.

These are some possible situations where the knowledge presented in this article might be of some use. If for no other reason, it is an interesting exercise nonetheless. Furthermore, it is kind of cool to have the ability to telnet or browse the Web using your Palm Pilot.

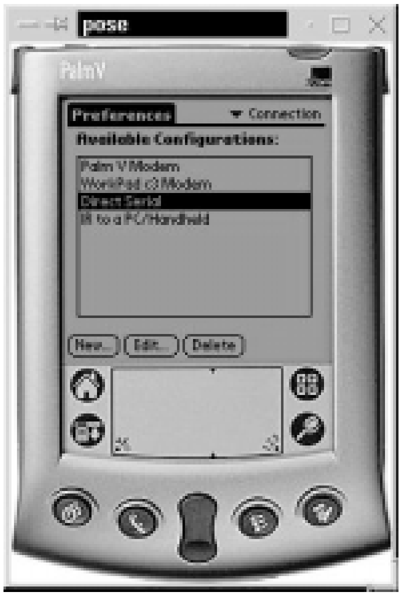
For those of you most familiar with Microsoft Windows, the subject of this article may seem less arcane since using the Microsoft RAS protocol makes PDA networking rather simple. It's not as evident how one would go about networking a Palm Pilot with one's Linux workstation, however. Recognizing that wireless Palm Pilot networking, using a wireless modem (like those offered by OmniSky) or a Palm VII is optimal for PDA networking, these alternatives are still rather pricey. PPP networking provides a cost-effective alternative for doing cool networking "things" with your Palm Pilot: application testing, for example.

Preparation

Most Linux distributions come with a number of PPP utilities. If you are unfamiliar with PPP, please refer to the PPP HOWTO articles at *http://wwww.linuxdoc.org*. Establishing a PPP connection between your Palm and Linux workstation is not that different from establishing a PPP connection with your Internet Service Provider (ISP), although configuring your Palm device correctly can be a little tricky. Also note that in the steps presented below, I am using the Beta version of RedHat Linux 7.0 (kernel version 2.4.0-0.99.11), and a Palm Vx (Palm OS v.3.5.0), so your results may vary slightly depending on your specific configuration.

OK, here we go. First, connect your Palm cradle to a free serial port at the back of your workstation. Make sure your cradle is properly connected. One way to do this is to grab

Step 1

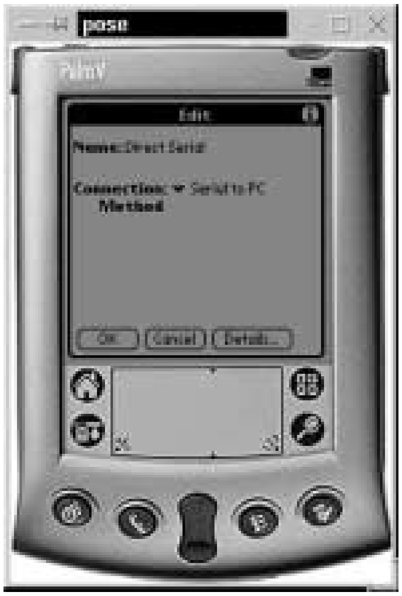


Preferences ▼ Connection

Available Configurations:

Palm V Modem
WorkPad c3 Modem
Direct Serial
IR to a PC/Handheld

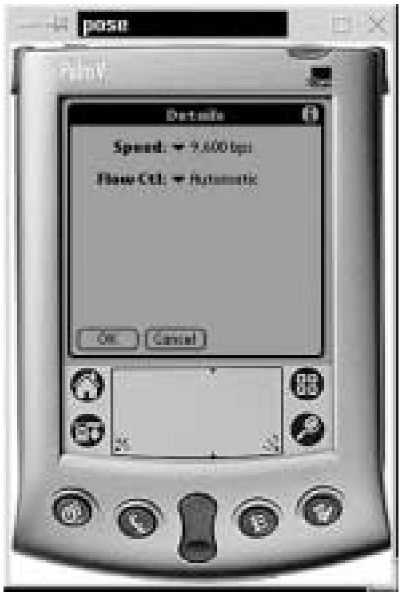
Step 2



Name: Direct Serial

Connection Method: ▼ Serial to PC

Step 3



Speed: ▼ 9,600 bps

Flow Ctl: ▼ Automatic

pilot-link from *http://pilot-link.sourceforge.net* and try to use the pilot-xfer utility to transfer a Palm Database (PDB) from your device to your workstation. For example, try to transfer MemoDB as follows:

```
[root@localhost]# pilot-xfer /dev/ttyS1 -f MemoDB
```

If you have trouble with this step, try using /dev/ttyS3. If all else fails, read the pilot-xfer documentation for further details.

PPP Configuration

Secondly, if you want to be able to reach machines beyond your own Linux workstation with your networked Palm Pilot, you will need to make sure that your kernel supports IP Masquerading. In the worst case, you will need to rebuild your kernel with support for IP Masquerading, but chances are, if you are using the latest RedHat distribution, it is already supported in your kernel. In addition, I had to remove the ipchains module in order for my ppplogin script (see below) to work. You can do this using the following command:

```
[root@localhost]# rmmod ipchains
```

You must either create or modify your existing /etc/ppp/options file to contain these parameters:

```
lock
debug
noauth
crtscts
```

Now create a simple script in /etc/ppp called ppplogin, which should contain the following lines:

```
# Load the NAT module (this pulls in all the others).
modprobe iptable_nat

# Note: Change eth0 to ppp0 if you use a modem.
# Dial-up for your Internet connection.
iptables -t nat -A POSTROUTING -o eth0 -j MASQUERADE

# Turn on IP forwarding.
echo 1 > /proc/sys/net/ipv4/ip_forward

# 10.100.166.56 is the IP of the local Linux workstation.
# 192.168.1.2 is the IP I wish to assign to the Palm device.
# Modify accordingly.
/usr/sbin/pppd 10.100.166.56:192.168.1.2 connect \
  '/usr/sbin/chat -vf /etc/ppp/chat' /dev/ttyS1
```

Finally, the last step in configuring PPP on your Linux machine is to create the /etc/ppp/chat script. This file should simply contain a line that reads TIMEOUT 30, or whatever timeout value you wish to assign to pppd.

Configure Network

Now that we have configured the PPP Server, we need to set up the network configuration on the Palm device. The accompanying screenshots, grabbed from the Palm OS Emulator for UNIX, show how to configure your network settings.

Step 1. Tap Prefs and select Connection from the pull-down menu.

Step 2. Select Direct Serial and tap Edit.

Step 3. Tap Details and configure as seen.

Step 4. Select Network from the pull-down menu and create a new Service as shown.

Step 5. Tap the Details button in step 4 and configure as shown.

Step 6. Tap the Script button in step 5 and configure as shown.

While these screenshots may prove helpful, I have made this process somewhat easier by providing Connection and Network Palm Databases containing this configuration information. If you want, you can download these files and install them on your device, which could save you some frustration (see Resources, below). Once you have obtained these Palm Databases, you can use the pilot-xfer utility on your Linux machine to install them on your Palm device like so:

```
[root@localhost]# pilot-xfer /dev/ttyS1 -i NetworkDB ConnectionDB
```

You might want to make a backup copy of your original databases before doing this, however (see the pilot-xfer man page).

Now we are ready to test our configuration. Place your Palm Pilot in the Hot Sync cradle. Tap the Prefs icon and select Network from the pull-down menu. Change your working directory on your Linux workstation to /etc/ppp, and run the ppplogin script as root. You can look at your system logfile after running ppplogin to make sure it is running properly (e.g., tail -f /var/log/messages). The following lines from /var/log/messages indicate that pppd is waiting for a ppp connection request on /dev/ttyS1:

```
Apr 30 22:07:02 velocis pppd[2715]: pppd 2.4.0 started by root, uid 0
Apr 30 22:07:03 velocis chat[2716]: timeout set to 30 seconds
Apr 30 22:07:03 velocis pppd[2715]: Serial connection established.
Apr 30 22:07:04 velocis pppd[2715]: Using interface ppp0
Apr 30 22:07:04 velocis pppd[2715]: Connect: ppp0 <—> /dev/ttyS1
```

Finally, tap the Connect button on your Palm (see step 4, above), and wait a second for the connection to get established. Provided you have IP Masquerading and your PPP settings properly configured, you should be able to access remote machines using a telnet client, or even a Web browser. You can also use this configuration to test your own custom network-aware Palm applications.

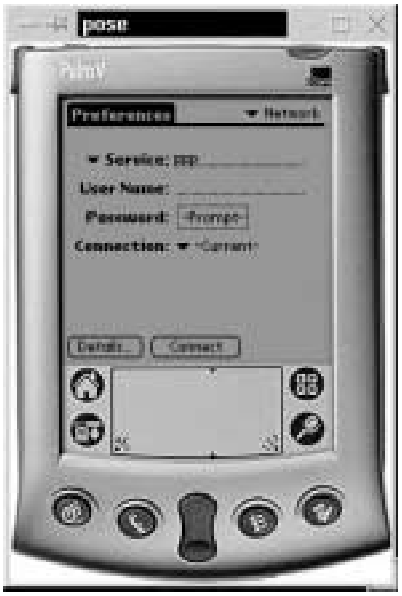
Conclusions

Establishing a PPP connection between your Palm and Linux workstation is just about as easy as dialing up your local ISP. There can be a number of benefits in having the ability to do this as well. The most notable benefit I see, however, is that this configuration, using your Palm Linux workstation and Palm Pilot, can provide a low-cost alternative to testing your network-aware Palm applications on real devices, without requiring more expensive wireless modems and service providers.

Resources

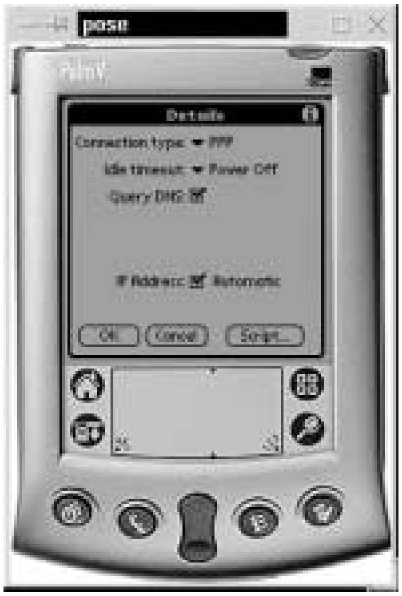
RFC1661 – *http://www.cis.ohio-state.edu/cgi-bin/rfc/rfc1661.html*
PPP-HowTo – *http://www.linuxdoc.org/HOWTO/PPP-HOWTO/index.html*
Pilot-Link – *http://pilot-link.sourceforge.net*
Palm OS Emulator (POSE) – *http://www.palm.com/devzone*
Sample Palm Network / Connection Databases –
http://www.trexlabs.com/USENIX/pdbs.tar.gz
Telnet Client for Palm – *http://www.mochasoft.dk*
Web Browser for Palm – *http://www.ilinx.co.jp/en*

Step 4



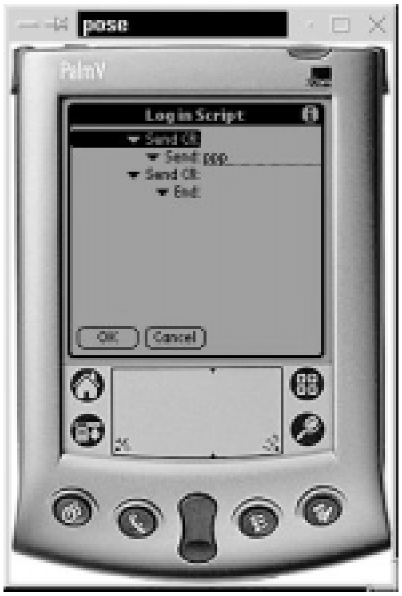
Preferences ▾ Network
▾ Service: PPP
User Name:
Password: -Prompt-
Connection: ▾ -Current-

Step 5



Details
Connection type: ▾ PPP
Idle timeout: ▾ Power Off
Query DNS: ☑
IP Address: ☑ Automatic

Step 6



Log in Script
▾ Send CR
▾ Send: ppp
▾ Send CR
▾ End: