Configuring and Using Solstice™ PPP 3.0.1 Clients

2550 Garcia Avenue
Mountain View, CA 94043
U.S.A.

Part No: 802-5353-10
Revision A, March 1996

SunSoft
A Sun Microsystems, Inc. Business

Contents

Preface

Configuring and Using Solstice PPP Clients tells you how to use the Solstice PPP product to connect your machine to remote servers across a public telephone network.

Solstice PPP is a standard implementation of the Point-to-Point Protocol (PPP) which defines a standard method for transporting multiprotocol datagrams across serial connections. This means that it can interoperate with all other standard implementations of PPP.

Who Should Use This Book

This book is intended for users with no experience connecting PPP clients, and with limited experience of the Solaris operating environment. It assumes that your system administrator or Internet provider has configured the remote server to accept your calls, and has given you all the information you need to configure your client machine.

Before You Read This Book

This book assumes that you have already installed the product software, as described in *Installing and Licensing Solstice PPP*. It also assumes that you have the information you need to configure your client machine. For example, the telephone number of the server to which you are connecting your machine.

How This Book Is Organized

Chapter 1, "Introducing Solstice PPP," describes how to use Solstice PPP to connect your client machine to remote servers across the public telephone network..

Chapter 2, "Configuring Your Solstice PPP Client," tells you how to use the configuration script `pppinit(1m)` to configure your Solstice PPP client.

Chapter 3, "Connecting your Solstice PPP Client to a Server," tells you how to use `ppptool(1m)` to connect your Solstice PPP client to a remote server, and how to run common network applications such as `mailtool(1)` and the Netscape™ web browser over a PPP link.

Chapter 4, "Solving Problems with Your Solstice PPP Client," describes how to detect and resolve common problems with Solstice PPP installed on a client machine.

Appendix A, "Configuration Files and CHAT Scripts," describes the syntax of the CHAT scripts used by Solstice PPP, and includes examples of both interactive and non-interactive CHAT scripts.

Related Books

The following books are part of the Solstice PPP document set:

- *Installing and Licensing Solstice PPP*
  Part No. 804-5278-10 (CD-insert Format)
- *Configuring and Using Solstice PPP Clients*
  Part No. 802-5353-10
- *Configuring and Using Solstice PPP Servers and Routers*
  Part No. 802-5354-10

What Typographic Changes Mean

The following table describes the typographic changes used in this book.

Table P-1 Typographic Conventions

Typeface or Symbol	Meaning	Example
`AaBbCc123`	The names of commands, files, and directories; on-screen computer output	Edit your `.login` file. Use `ls -a` to list all files. `machine_name% You have mail.`
`AaBbCc123`	What you type, contrasted with on-screen computer output	`machine_name%` **`su`** `Password:`
AaBbCc123	Command-line placeholder: replace with a real name or value	To delete a file, type `rm` *filename*.
AaBbCc123	Book titles, new words or terms, or words to be emphasized	Read Chapter 6 in *User's Guide*. These are called *class* options. You *must* be root to do this.

Shell Prompts in Command Examples

The following table shows the default system prompt and superuser prompt for the C shell, Bourne shell, and Korn shell.

Table P-2 Shell Prompts

Shell	Prompt
C shell prompt	`machine_name%`
C shell superuser prompt	`machine_name#`
Bourne shell and Korn shell prompt	`$`
Bourne shell and Korn shell superuser prompt	`#`

Introducing Solstice PPP 1

This chapter describes how to use Solstice PPP to connect your client machine to remote servers across private and public telephone networks.

Overview

Solstice PPP is an implementation of the Point-to-Point Protocol. It is used to connect a client machine to a public or private telephone network through a modem, and to initiate calls to one server at a time. You can then run common network applications, such as `mailtool(1)` or the Netscape™ web browser, to access the resources of your remote office or Internet provider.

Note – If you *have not* installed the software license system for Solstice PPP, you are limited to a single modem connection. You can still initiate calls to multiple remote servers; however, you can only communicate with one remote server at a time.

If you *have* installed the software license system for Solstice PPP, you can make multiple modem connections; therefore, you can initiate several calls to remote servers simultaneously. See *Installing and Licensing Solstice PPP* for instructions on how to obtain and install a license for this product.

Figure 1-1 shows a typical configuration, with a Solstice PPP client connected to an office server. The server may also be running Solstice PPP, or it may be using some other implementation of the Point-to-Point Protocol (PPP) that allows it to route IP traffic.

Your system administrator or Internet provider must configure the server to accept calls from the client, and must provide you with the information you need to configure your client machine.

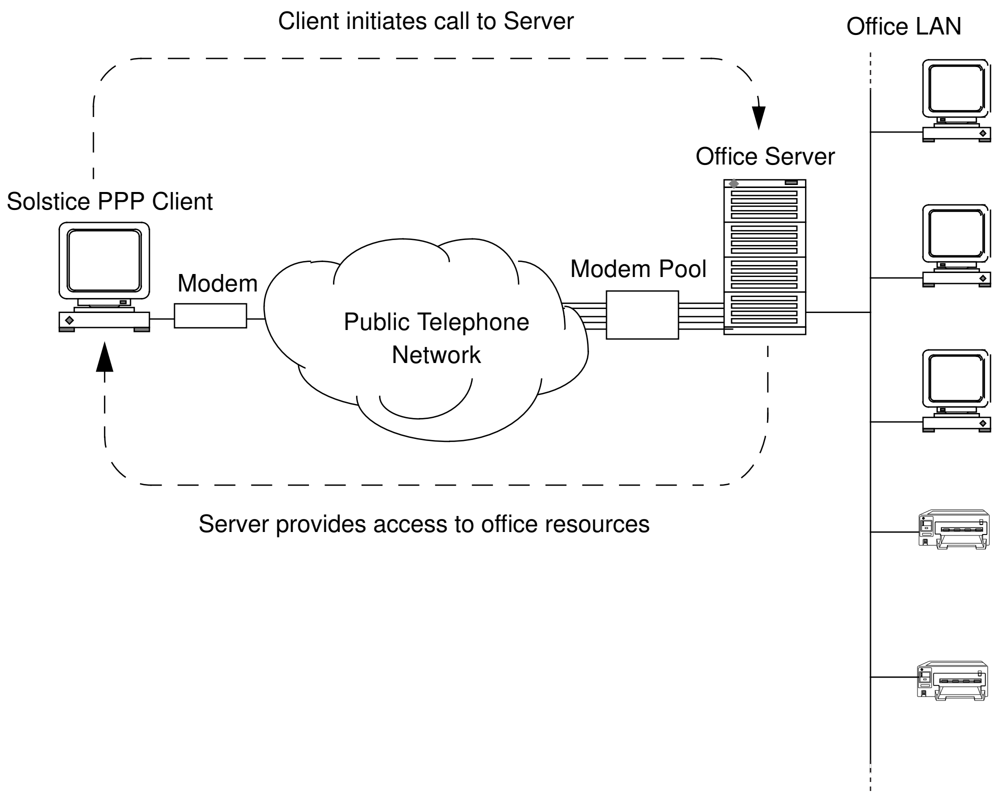

Figure 1-1 Solstice PPP Client Accessing Remote Resources

Configuring Your Solstice PPP Client

The initialization script `pppinit(1m)` is used to configure Solstice PPP on your client for the first time. It will prompt you for the information that describes your particular configuration, and create the relevant configuration files.

The configuration files for Solstice PPP are:

- `/etc/opt/SUNWconn/ppp/ppp.conf`
- `/etc/opt/SUNWconn/ppp/ppp.link`

Normally, you should not need to modify these files manually to configure your client; however, a list of commonly used keywords and their appropriate values is contained in Appendix A, "Configuration Files and CHAT Scripts".

Connecting to a Remote Server

Once you have configured Solstice PPP on your client, you can use `ppptool` to display a view of the servers to which you are able to connect, and to initiate connections to these servers. See Chapter 3, "Connecting your Solstice PPP Client to a Server" for a detailed description of how to use `ppptool`.

It may take up to one minute to connect your client to a server, during which the following events occur:

1. **The client dials the server, and the two modems communicate to set up the physical connection across the telephone network.**
   If the speaker is enabled on your modem, you will hear the tones generated when the telephone number is dialed and the carrier signal is detected.

2. **The client logs in to the server.**
   Most implementations of PPP require a login phase. For Solstice PPP running on Solaris systems, this is equivalent to the UNIX login sequence. The user id and password sent to the server are contained in its CHAT script. See "CHAT Scripts" on page 5 for more information.

3. **The client and the server negotiate a common configuration for the PPP link between them.**
   The policy is to converge the negotiation, if at all possible; however, if certain mandatory parameters, such as authentication parameters, do not match on both sides of the link, the connection will be closed automatically.

Security and Authentication

You need to take care when transmitting potentially confidential information over public telephone networks, and particularly when communicating on the Internet. The communication is not encrypted by the transmission media; therefore, anyone can tap into the network and read the information you send. However, you can use Sun security products, such as Solstice SunScreen or Solstice Firewall-1, to protect your network and encrypt the network traffic.

Solstice PPP supports three levels of authentication that are used by the server to prevent unauthorized clients from connecting to it:

- Standard UNIX login sequence
- Password Authentication Protocol (PAP)
- Challenge Handshake Authentication Protocol (CHAP)

Optionally, Solstice PPP can also be used in conjunction with third-party, dynamic challenge-response authentication products. Your system administrator or Internet provider will tell you which of these authentication methods are implemented by your server.

Standard UNIX Login Sequence

This is the simplest form of authentication, and is required by most implementations of PPP running on UNIX servers. Your system administrator or Internet provider will give you a PPP user id and password to use for the UNIX login.

Password Authentication Protocol (PAP)

The Password Authentication Protocol (PAP) is an optional authentication method that identifies the client based on its PAP id and PAP password. Your system administrator or Internet provider will give you a PAP id and password, if this feature is enabled on the remote server.

Challenge Handshake Authentication Protocol (CHAP)

The Challenge Handshake Authentication Protocol (CHAP) is an optional authentication method, which is more secure than PAP. CHAP identifies the client based on its CHAP id and a challenge value that is calculated from the CHAP secret. Your system administrator or Internet provider will give you a CHAP id and secret, if this feature is enabled on the remote server.

Dynamic Challenge-Response Authentication

Solstice PPP can be used in conjunction with third-party, dynamic challenge-response authentication products. These products issue a challenge value that requires a response from the user. This response is calculated dynamically based on the challenge value. Note that this is the only authentication method that controls the identity of the user rather than the identity of the client machine.

Solstice PPP uses interactive CHAT scripts, which are discussed in the next section, to support challenge-response authentication.

CHAT Scripts

A CHAT script (sometimes called a *connect* script, by other implementations of PPP) is executed automatically each time the client initiates a call. It is used to exchange information with the server during the connection phase. Solstice PPP supports two types of CHAT script:

- A *non-interactive* CHAT script defines all the information the client needs to exchange with the server.
- An *interactive* CHAT script requires some input from the user. For example, the response to dynamic challenge-response authentication.

The exact contents and syntax of the CHAT script is dependent on the configuration of the server and the authentication method it uses, if any. Therefore you need a different CHAT script to connect to each server.

Your system administrator or Internet provider should provide you with a CHAT script that enables your client to connect to their server. Alternatively, you may be given a CHAT script template that you must modify for your client. See Appendix A, “Configuration Files and CHAT Scripts” for more information.

Configuring Your Solstice PPP Client 2

This chapter tells you how to use the initialization script `pppinit(1m)` to configure your Solstice PPP client for the first time.

Configuration-Specific Information

When you run `pppinit`, you will need the following information, which you must obtain from your system administrator or Internet provider:

- The telephone number used to make calls to the remote server.
- Your PPP login id and password.

Depending on the configuration of the remote server, your system administrator or Internet provider may also give you some of the following optional information:

- The name used to identify the remote server.
- An IP address for your client machine.
- A CHAT script, or CHAT script template.
- A PAP identifier and password.
- A CHAP name and secret.

Running pppinit

Caution – Running pppinit will overwrite any previous configuration of Solstice PPP that you have on your machine. You should use pppinit to initialize your client the first time you run Solstice PPP.

1. **Login as root, or become superuser**
2. **Start the initialization script pppinit, by typing:**

```
prompt# /usr/bin/pppinit

Welcome to the Solstice PPP 3.0 configuration script
```

3. **Select the appropriate modem from the list displayed.**
   Up to ten modems are displayed at one time. Use the + and - keys to scroll up and down the list.

```
Modem configuration
-------------------

    Select one modem from your database
    There are 20 modems available:
     [0]   - Null Modem
     [1]   - BocaModem V.34 DataFax
     [2]   - AT&T DataPort Express
     [3]   - AT&T 14400 bps Data-Fax PCMCIA Modem
     [4]   - Cardinal V.34/V.FC 28.8 data/fax
     [5]   - Cardinal MVP288I 28.8 Kbps V.34 Fax Modem
     [6]   - SupraFaxModem 288
     [7]   - Hayes Accura 144B
     [8]   - Hayes Accura 288V.FC
     [9]   - Practical 14400 V32bis
     [10]  - USRobotics Sporter 14400

Modem type (+/- to scroll the list):
```

4. **Enter the name of the serial port to which your modem is connected.**
   The script displays a list of the devices it detects in your system. The onboard serial interfaces in a SPARC workstation or server have device names of the form `/dev/tty`*n*.

```
List of unix devices available:
[ ttya ttyb ]
Unix device used for this modem [ttya]: ttyb
```

5. **Enter the name of the remote server.**
   This name is only used to identify the server as it appears in `ppptool` and your local configuration files. If your system administrator or Internet provider did not give you a name for your server, you can enter any character string in response to this question.

```
Name of the remote server: server1
```

6. **Enter the phone number used to call this remote server.**
   Solstice PPP will accept both digits and characters, including special characters such as `#` and `*`. Remember to include any digits or characters required to pass outside your local private branch exchange, if necessary. Commas are used to insert a pause. The letter P is used to invoke pulse dialing. For example, `P0,,123456789`.

```
Phone number for this server: P0,,123456789
```

7. **Enter the name of the CHAT script to be used for this remote server.**
   This script will be executed automatically each time you connect to this server.

```
Filename of the chat script [miles-ppp.scr]:
```

   If the remote server is running a Solaris operating environment, `pppinit` can create a simple CHAT script to handle the UNIX login phase for your client.

Enter your PPP login id and password.

```
This script can create a template file, with default
parameters for connecting to Unix systems.
Do you want to do this now? [y] y

Login id sent to miles-ppp: ppp2
Password sent to miles-ppp as ppp2:
    Password:
    Re-enter Password:
```

8. **Enter the IP addresses for your client and for this server.**
   If your system administrator or Internet provider did not give you any IP addresses, press Return to accept the default value `none`. The server will assign IP addresses for the PPP link when you make the connection.

   You can enter IP addresses in dot notation, or represented by a host name. If you enter a host name, it must be associated with an IP address in the local file `/etc/hosts`.

```
Enter your IP address [none]? 129.xxx.xxx.117
Enter the Server IP address: 129.xxx.xxx.101
```

9. **Enter the IP netmask:**
   If your system administrator or Internet provider did not give you an IP netmask, press Return to accept the default value.

```
IP netmask for this interface [255.255.255.0]:
```

10. **Enter the PAP id and password for this server.**
    If your system administrator or Internet provider did not give you any PAP parameters, press Return to accept the default value `none`. This feature is not enabled on the remote server.

```
Enter your PAP Id [none]? angel505
    Enter the PAP Password:
    Re-enter Password:
```

11. **Enter the CHAP name and secret for this server.**
    If your system administrator or Internet provider did not give you any CHAP parameters, press Return to accept the default value `none`. This feature is not enabled on the remote server

```
Enter your CHAP Name [none]? papyrus*ok
   Enter the CHAP Secret:
   Re-enter Secret:
```

12. **Enter the inactivity timeout.**
    This sets the amount of time that an idle connection will remain open before it is closed automatically. The default value is sufficient under most circumstances.

```
Inactivity timeout [120]:
```

You have now configured your Solstice PPP client so that it can connect to a remote server. To specify another remote server, repeat steps 5 through 12.

```
Do you want to add an access to another server? [n] y
```

When you have finished, you can either save your configuration to file, or exit the script without saving.

```
[1] - Asynchronous client (done).
[W] - Exit Without saving
[E] - Exit and Save
```

The information you save to file is used to configure Solstice PPP each time it is started on your machine, and to initialize your modem each time you initiate a call to a server.

Starting and Stopping Solstice PPP

Your new configuration is invoked the next time you start Solstice PPP on your machine. Solstice PPP is started automatically each time your machine boots.

To start Solstice PPP manually, become root and type:

```
prompt# /etc/init.d/ppp start
```

To stop Solstice PPP manually, become root and type:

```
prompt# /etc/init.d/ppp stop
```

Connecting your Solstice PPP Client to a Server 3

This chapter tells you how to use `ppptool(1m)` to connect your Solstice PPP client to a remote server, and how to run common network applications such as `mailtool(1)` and the Netscape web browser over a PPP link.

About `ppptool`

The graphical user interface for Solstice PPP is called `ppptool`. Using `ppptool` you can connect your Solstice PPP client to any of the remote servers that you defined when you ran `pppinit`.

Note – If you *have not* installed the software license system for Solstice PPP, you are limited to a single modem connection. You can still initiate calls to multiple remote servers; however, you can only communicate with one remote server at a time.

If you *have* installed the software license system for Solstice PPP, you can make multiple modem connections; therefore, you can initiate several calls to remote servers simultaneously. See *Installing and Licensing Solstice PPP* for instructions on how to obtain and install a license for this product.

To Start ppptool

You can either start ppptool from the command line, or add the command to your workspace menu.

♦ **To start ppptool from the command line, type:**

```
prompt% /usr/bin/ppptool
```

When you start ppptool, the Hosts map is displayed, as shown in Figure 3-1. The Hosts map contains one icon for each of the servers you defined when you ran the initialization script pppinit.

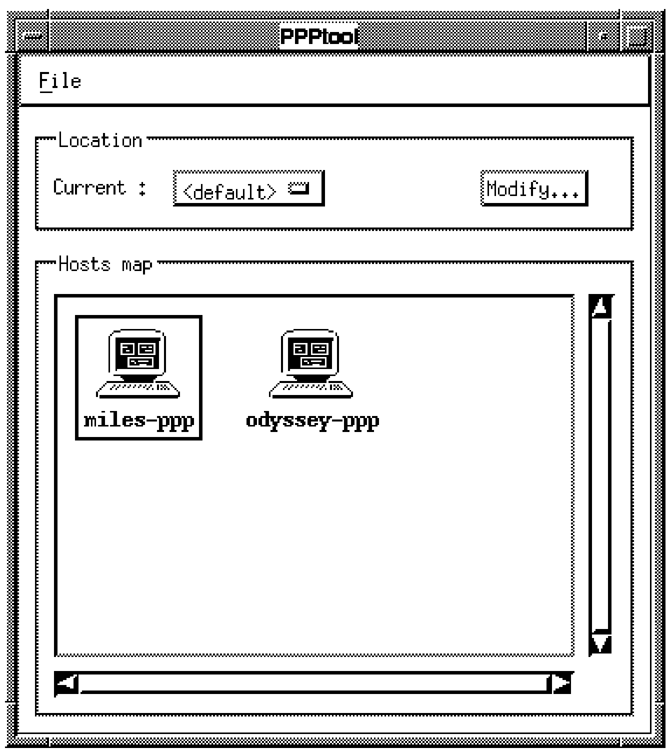

Figure 3-1 The ppptool Hosts map

When you double-click SELECT on one of the server icons in the hosts map, the Connection window for that server appears, as shown in Figure 3-2.

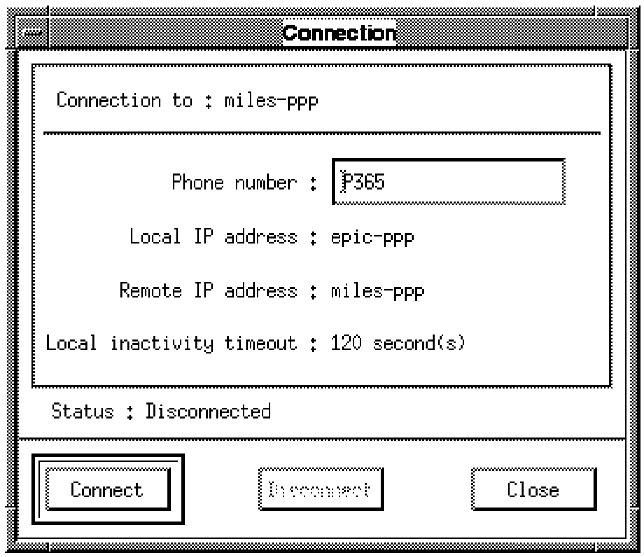

Figure 3-2 The Connection Window

The Connection window displays the following information:

Connection to:
The name used to identify the remote server for this connection.

Phone number
The telephone number used to initiate calls to the server. This is the telephone number you assigned when you ran `pppinit`; however, it is an editable field so you can change the number if you want. Your changes are not saved when you exit `ppptool`.

Local IP address
The IP address assigned to your Solstice PPP client, expressed either as a host name or in dot notation. This field will be set to `<unassigned>` if the remote server is going to supply the IP address. It will change to show the IP address or hostname received from the server when the connection is established.

Remote IP address
The IP address assigned to the remote server, expressed either as a host name or in dot notation. This field will be set to `<unassigned>` if the remote server is going to supply the IP address. It will change to show the remote IP address or hostname received from the server when the connection is established.

Local inactivity timeout
This is the time that the connection can remain unused before it is closed automatically.

Status:
The current status of the connection.

Using `ppptool`

▼ To Connect to a Server

When you initiate a connection to a server, you use the telephone number that you assigned using `pppinit`, preceded by the prefix associated with your current location. There is no prefix if the current location is `<default>`. To connect your Solstice PPP client to a server:

1. **Double-click SELECT on the server's icon to display its Connection window.**
2. **Click SELECT on the Connect button to initiate the connection.**
   If your client uses an interactive CHAT script for this connection, a dialog box will appear. Type in the responses requested by the remote server. For example, user id and challenge passwords.

▼ To Disconnect from a Server

The connection to the server is usually terminated automatically when the connection remains unused for the period of time defined by the local inactivity timeout. To disconnect your Solstice PPP client from a server manually:

1. **Double-click SELECT on the server's icon to display its Connection window.**
2. **Click SELECT on the Disconnect button to terminate the connection.**

▼ To Change the Current Location

The Current Location is used to add a prefix to all the telephone numbers associated with the servers defined in the hosts map. This feature is used for mobile communications when you cross exchange and international boundaries. To change the Current Location:

♦ **Choose a new location from the Current Location pull-down menu.**

▼ To Add a new Location to the Current Location List

You can add a new location and prefix to the Current Location list at any time. To add a new location to the Current Location list:

1. **Click SELECT on the Modify... button to display the Location window.**
2. **Click SELECT on the New Location button and enter the Location name and associated Prefix.**
3. **Click SELECT on the Add button to enter the new location in the list.**

▼ To Modify the Current Location List

You can modify the name and prefix of an existing location at any time. The change will take effect the next time you initiate a call from that location. To modify an existing location:

1. **Click SELECT on the Modify... button to display the Location window.**
2. **Choose one of the locations from the list and modify its prefix.**
3. **Click SELECT on the Modify button to change the prefix.**

▼ To Delete a Location from Current Location List

You can delete an existing location at any time. To delete an existing location:

1. **Click SELECT on the Modify... button to display the Location window.**
2. **Choose one of the locations from the list.**
3. **Click SELECT on the Delete button to remove it from the list.**

Running Applications over your PPP Link

Once you have connected your Solstice PPP client to a remote server, you can run network applications transparently over the PPP link in the same way as you would over your local area network connection.

Running Applications without using a Name Service

Name services, such as the Domain Name Service (DNS), are used to resolve hostnames into their corresponding IP addresses. You can run simple IP applications over your PPP link without enabling a name service on your client; however, you need to know the IP address or hostname of your server. This is the remote IP address displayed in the connection window by `ppptool`, as shown in Figure 3-3.

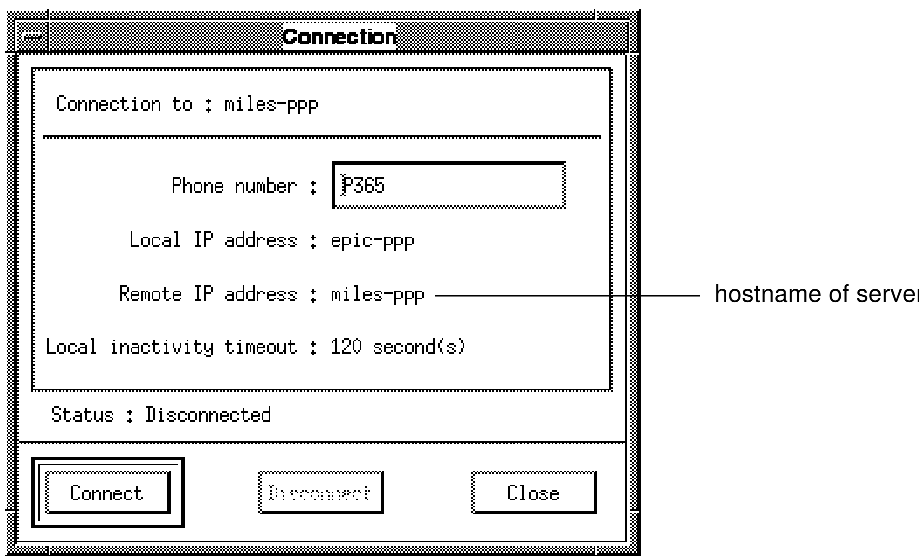

Figure 3-3 Remote IP Address of Server

Once you are connected to the server, you can use Unix commands such as `rlogin(1)`, `rsh(1)`, `telnet(1)`, and `ftp(1)` to access the resources of the remote network.

Running Applications with DNS Enabled

The Domain Name Service (DNS) is the most commonly used mechanism for resolving IP addresses and hostnames. You need to use a system such as DNS if you want to be able to browse a network or to reach further than your server. The server must be configured to route IP traffic, and your system administrator or Internet provider must give you the following information:

- The domain name for the subnetwork
- The IP address(es) of your name server(s)

The name server, and the server to which your client is connected, may be the same machine; but they do not have to be.

To enable DNS on your client:

1. **Edit the file `/etc/nsswitch.conf`, and modify the `hosts` entry as follows:**

```
hosts:   files dns
```

2. **Create the file `/etc/resolv.conf`, and enter the domain name and the IP addresses of all the name servers as follows:**

```
domain        xyz.Company.COM
nameserver    179.xxx.aaa.11
nameserver    ...
nameserver    ...
```

3. **Optionally, edit the file `/etc/hosts` to add the IP address and hostname of your server:**

```
127.0.0.1       localhost
179.xxx.aaa.117 papyrus loghost
179.xxx.aaa.100 server
```

All of these changes take place automatically; there is no need to reboot your machine.

Solving Problems with Your Solstice PPP Client 4

This chapter tells you how to detect and resolve problems with Solstice PPP installed on a client machine.

First Steps in Troubleshooting

1. Check the cable between your machine and the modem, and the cable between the modem and the telephone socket.
2. Check that the modem is switched on, and is configured correctly.
3. Check that you ran `pppinit` correctly, and that you entered the configuration information exactly as it was given to you.
4. Check that Solstice PPP is configured and running on your machine, by typing:

```
prompt% ps -ef | grep ppp
root pid timestamp 0:00 /usr/sbin/pppd -d 1
root pid timestamp 0:00 /usr/sbin/pppd -d 1
```

5. Check for error and status messages in the Solstice PPP log file (`/var/opt/SUNWconn/ppp.log`), by typing:

Outgoing call detected

Login sequence failed

```
prompt# tail -f /var/opt/SUNWconn/ppp.log
05/31/95 22:52:48 - Link manager (17302) has started 05/31/95
05/31/95 22:52:48 - Successful configuration
05/31/95 22:55:02 - Connection requested to remote_server
05/31/95 22:55:03 - Dialing number 389 ...
05/31/95 22:55:21 - Got modem connection
05/31/95 22:56:02 - fail at line 12 in chat script chat_script
```

Understanding the Log File

Error and status messages for Solstice PPP are written to the log file `/var/opt/SUNWconn/ppp.log`. These messages show the progress of successful connection attempts, and can help identify which part of your configuration may be causing a problem if the connection attempt fails.

To display the latest messages written to the log file, type:

```
prompt# tail -f /var/opt/SUNWconn/ppp.log
```

The following log file extract shows Solstice PPP started and configured successfully on your machine. A connection to a remote server is opened and then closed.

Solstice PPP started successfully

Modem connection complete

PPP negotiation complete and connection open

Disconnect request received and connection closed

```
11/13/95 18:00:48 - Link manager (359) has started 11/13/95
11/13/95 18:00:48 - Successful configuration
11/13/95 18:00:58 - Connection requested to miles
11/13/95 18:00:59 - Dialing number P365 ...
11/13/95 18:01:32 - Got modem connection
11/13/95 18:01:35 - LCP up on ipdptp0
11/13/95 18:01:35 - IP_NCP up on ipdptp0
11/13/95 18:01:35 - IP up on interface ipdptp0, with timeout set
to 240 seconds
11/13/95 18:01:54 - Disconnect indication on ipdptp0
11/13/95 18:01:54 - IP_NCP down on ipdptp0
11/13/95 18:01:54 - LCP down on ipdptp0
```

Problems Configuring Solstice PPP

When Solstice PPP is started on your machine, it reads the configuration files that you created using the configuration script `pppinit`. If these files do not exist on your machine, Solstice PPP displays the following message:

```
starting ppp (not configured)
```

The following log file extract shows the error messages that are generated when one of the configuration files contains an error:

```
11/13/95 18:53:22 - Link manager (460) has started 11/13/95
11/13/95 18:53:22 - parse_config_file: unrecognized symbol
inactivity_timeout
11/13/95 18:53:22 - parse_config_file: unrecognized symbol 180
11/13/95 18:53:22 parse_config_file: Errors in configuration file
/etc/opt/SUNWconn/ppp/ppp.conf
```

In this example, there is an unrecognized keyword contained in the file `ppp.conf`, which is rejected when the file is parsed. If you have a problem in one of your configuration files, run the configuration script `pppinit` to regenerate it. Contact your system administrator or Internet provider for help, if this does not solve your problem.

Problems Getting the Modem Connection

The first step in the connection phase is the modem connection. The client dials the telephone number of the modem connected to the server, and the two modems communicate to set up the connection.

The following log file extract shows a failed attempt to establish the modem connection:

```
11/13/95 19:57:44 - Connection requested to miles
11/13/95 19:57:45 - Dialing number P365 ...
11/13/95 19:58:09 - remote host is busy
```

The message “remote host is busy” is displayed whenever the client receives a busy signal from the server. This may mean that the server modem is already in use by another client, or it may mean that there is a problem with the equipment. In this example, the error was provoked by using tone dialing with an office exchange that expected the client to use pulse dialing.

Try telephoning the number directly to make sure that you can reach the modem. If you cannot reach the modem, check the telephone number you are dialing. If you are sure that you are dialing the correct number, but you receive a busy signal systematically, contact your system administrator or Internet provider for help.

Problems with PPP Negotiation

Once you have a modem connection, the next step in the connection phase is the PPP negotiation. The client and the server communicate to negotiate a common configuration for the PPP link. The policy is to converge if at all possible; however, failure to agree on certain mandatory parameters will cause the negotiation to fail.

The following log file extract was generated when the PPP negotiation failed:

```
11/13/95 20:34:42 - Connection requested to miles
11/13/95 20:34:43 - Dialing number P365 ...
11/13/95 20:35:16 - Got modem connection
11/13/95 20:35:19 - LCP up on ipdptp0
11/13/95 20:35:19 - PPP error on ipdptp0: Negotiation of
mandatory options failed
```

In this example, the error was provoked by an IP address that did not match the IP address expected by the server. If the PPP negotiation fails, check your configuration against the information that your system administrator or Internet provider gave you. If it is correct, contact your system administrator or Internet provider for help, as they may need to modify the server configuration.

Problems with PAP and CHAP Authentication

PAP and CHAP authentication are used to prevent unauthorized clients from connecting to the server. If either of these authentication methods is enabled on the server, your system administrator or Internet provider will provide you with the relevant passwords and identifiers.

The following log file extract shows what happens when your client fails to respond to a request for authentication:

```
11/14/95 10:22:41 - Connection requested to miles
11/14/95 10:22:42 - Dialing number P365 ...
11/14/95 10:23:16 - Got modem connection
11/14/95 10:23:47 - PPP error on ipdptp0: Maximum number of
configure requests exceeded
```

In this example, the server requested PAP authentication and the client rejected the request. After a specified number of requests, the server broke the connection without starting the PPP negotiation.

The following log file extract shows what happens when the client responds to the request for authentication with the wrong password:

```
11/13/95 20:34:42 - Connection requested to miles
11/13/95 20:34:43 - Dialing number P365 ...
11/13/95 20:35:16 - Got modem connection
11/13/95 20:35:19 - LCP up on ipdptp0
11/13/95 20:35:19 - PPP error on ipdptp0: Negotiation of
mandatory options failed
```

In this example, the PPP negotiation fails and the server closes the connection automatically.

Problems with the Inactivity Timeout

The inactivity timeout closes the connection automatically when it remains unused for a specified number of seconds. This means that you do not pay for telephone connections that are left open accidently. However, if the inactivity timeout is too short, your connection may be closed prematurely.

The following log file extract shows an inactivity timeout triggered by the client after 60 seconds:

Disconnect triggered by inactivity timeout on the client side

```
11/13/95 18:00:03 - IP up on interface ipdptp0, with timeout set
to 60 seconds
11/13/95 18:00:14 - Interface ipdptp0 has timed out
```

If the connection times out systematically, run the configuration script `pppinit` to regenerate the configuration files, and increase the inactivity timeout for calls to the server.

The following log file extract shows a connection that has been disconnected by the server for some reason. One possible cause is an inactivity timeout on the server that is shorter than the inactivity timeout on the client. Contact your system administrator or Internet provider for help if the server disconnects your client systematically.

Disconnect triggered by server for an unspecified reason

```
11/13/95 18:13:46 - IP up on interface ipdptp0, with timeout set
to 240 seconds
11/13/95 18:15:47 - interface ipdptp0 has been disconnected
```

Solving Common Problems

The following sections describe common problems you may encounter when installing or using Solstice PPP.

Problems Installing Solstice PPP

Problem: Cannot start `pkgadd(1M)`.

Solution: You must log in as `root` or become `superuser` before you can run the `pkgadd`.

Problem: Cannot find the packages for Solstice PPP.

Solution: Check that you typed the source directory correctly. If the Volume Manager (`vold`) is running on your machine, the Solstice PPP packages are located in `/cdrom/ppp_3.0`. If you are not running the Volume Manager (vold) on your machine, you need to mount the CD-ROM manually. See *Installing and Licensing Solstice PPP* for detailed instructions.

Problems Using Solstice PPP

Problem: Solstice PPP stops working after upgrading to Solaris 2.5.

Solution: Solaris 2.5 creates a default file /etc/ttydefs, which overwrites some changes made by Solstice PPP. Safeguard the configuration files ppp.conf, ppp.link, and the CHAT scripts. Reinstall Solstice PPP to correct the problem.

Problem: Cannot establish PPP link. Operation fails with the status message: "PPP error on ip_interface: Maximum number of configure requests exceeded"

Solution: PPP Configure-request frames are generated to start the link establishment phase. After a certain number of frames (defined by the keyword lcp_max_restart in the file ppp.conf) are generated without a valid response, the client assumes that the remote server is unreachable. This may indicate one of the following:

There is a problem with the physical connection between the two hosts. Check the cable to your modem.

PPP is not running on the remote host. Check that PPP is configured and started at both ends of the link.

The link establishment phase is not completed, because the configuration negotiation does not converge. Check for configuration problems.

If you are trying to establish a link over a long-delay network, such as a satellite connection, or over a congested line, the maximum number of configure requests may be exceeded before the negotiation is completed. Increase the maximum number of configure requests sent (lcp_max_restart) and the time between retries (lcp_restart_timer).

Problem: Cannot establish PPP link. Operation fails with the status message: "Authentication failed"

Solution: The peer authentication phase failed. Check that the PAP and CHAP parameters set on the two hosts are coherent. If the server requests authentication using either PAP or CHAP, the client must participate in the authentication phase, or the link is closed.

Problem: Cannot establish PPP link. Operation fails with the status message: `"Loop back detected"`

Solution: The PPP frames generated by the remote server are being reflected. The *magic numbers* contained in the PPP frames indicate a loop back condition. This may indicate one of the following:

There is a problem with the physical connection between the two hosts. Check the cable to your modem.

The UNIX login sequence is not completed successfully. Check that the login id and password set in the CHAT script are correct, and that a corresponding user account exists on the remote server. Check that the rest of the login dialog defined in the CHAT script is correct. If the remote server is not running a Solaris environment, you may need to modify the login sequence provided in the template file created by `pppinit(1M)`.

The remote host fails to respond quickly enough, and the maximum number of configure requests is exceeded before the negotiation is completed. Increase the maximum number of configure requests sent (`lcp_max_restart`) and the time between retries (`lcp_restart_timer`).

Problem: Modem dials unexpectedly, or when the machine is rebooted.

Solution: The IP interfaces associated with Solstice PPP are usually marked *down*, by default. However, if you modify the file ppp.conf to mark the interfaces *up*, the PPP link manager initiates the PPP link automatically whenever an IP datagram is passed to the interface by the IP layer.

Some applications and processes broadcast requests occasionally. For example, when searching for a license daemon, or when the machine is rebooted. The PPP link manager responds to the broadcast and tries to dial the remote host.

To prevent this behavior, mark the relevant IP interface *down*, and use `pppconn(1M)` to initiate connections as required.

Problem: The connection phase fails with the error message: "`PPP error on ipdptp1: Negotiation of mandatory options failed`"

Solution: Check that the IP interface used to initiate the connection, and the IP interface associated with the dialup path used to accept the connection have coherent IP addresses. The source address on the client must match the destination address on the server.

Problem: Cannot make `rsh(1)` or `rlogin(1)` connection to the remote host. Operation fails with the message "`Permission denied`"

Solution: Check for the hostname, or IP address, of your local host in the files `/etc/.rhosts` and `/etc/hosts.equiv` on the remote host. A + character in these files enables access for all hosts.

Configuration Files and CHAT Scripts

A

This appendix lists some of the most common keywords that appear in the configuration files for Solstice PPP. It also describes the syntax of the CHAT scripts used by Solstice PPP, and includes examples of both interactive and non-interactive CHAT scripts.

PPP Path Configuration File (`ppp.conf`)

The PPP path configuration file (`/etc/opt/SUNWconn/ppp/ppp.conf`) describes the path used to reach each remote server. Each path is identified by the keyword `dialup_path`. For example:

```
dialup_path
    ip_interface          ipdptp0
    remote_host           miles-ppp
    request_ip_addr       on
    inactivity_timeout    120
    default_route
    .
    .
```

The most common keywords that may appear in your PPP path configuration file are:

`dialup_path`
: Indicates the start of an asynchronous (or dialup) path definition.

`ip_interface` *interface*
: Mandatory parameter. Associates the path with one of the point-to-point (`ipdptp`*n*) IP interfaces defined in the `ifconfig` section of the file.

`remote_host` *name*
: Mandatory parameter. Identifies the remote server reached using this path. The value *name* can be any character string.

`default_route`
: Optional parameter. Adds the route to the routing table as the default destination.

`inactivity_timeout` *seconds*
: Optional parameter. Specifies the number of seconds of inactivity that elapse before the connection is closed automatically.

  The value *seconds* can be any integer. The default value is 120 seconds (2 minutes). If the value *seconds* is set to zero, the connection remains open until closed explicitly.

`request_ip_addr` *state*
: Optional parameter. Enables dynamic IP address allocation *at the client side only*. When the value *state* is set to on, the client requests its IP address from the server. This feature must also be configured and enabled at the server side.

  The value *state* can be `on` (enabled) or `off` (disabled). The default value is `off`.

`send_authentication` *mode*
: Optional parameter. Indicates whether the client will respond to a request for authentication by the server.

  The value *mode* can be `off` (no authentication), `pap` (authentication using PAP), `chap` (authentication using CHAP), or `pap|chap` (authentication using both PAP and CHAP). The default value is `off`.

`send_pap_id` *pap_id*
: Specifies the PAP identifier sent to the server when it requests authentication. The value *pap_id* can be any string, between zero and 255 characters in length.

`send_pap_passwd` *pap_passwd*
: Specifies the PAP password sent to the server when it requests authentication. The value *pap_passwd* can be any string, between zero and 255 characters in length.

`send_chap_name` *chap_name*
: Specifies the CHAP name sent to the server when it requests authentication. The value *chap_name* can be any string, between 1 and 255 characters in length.

`chap_own_secret` *chap_secret*
: Specifies the CHAP secret that is combined with the challenge value to generate the response sent to the server. The value *chap_secret* can be any string, between 1 and 255 characters in length.

Link Configuration File (`link.conf`)

The link configuration file (`/etc/opt/SUNWconn/ppp/link.conf`) contains a description of your modem configuration and the dialing information used to make a connection to a remote server.

Modem Configuration

Each modem configuration is identified by the keyword `dialup_device`. For example:

```
dialup_device                pppdev0
    unix_device              ttya
    line_speed               38400
    modem                    Practical 14400 V32bis
    call_setup               dial
```

The most common keywords that may appear in this part of your link configuration file are:

`dialup_device pppdev`*n*
Indicates the start of an asynchronous device definition, and assigns a name to the device.

`unix_device` *device_name*
Specifies the serial port used to connect the client to the modem.

`line_speed` *speed*
Specifies the line speed for the connection between the client and the modem. For optimum performance, the line speed should be equal to, or greater than, the baud rate of the modem.

`modem` *modem_id*
Specifies the type of modem connected to the serial port, and associates the asynchronous device with one of the modem definitions in the file `/etc/opt/SUNWconn/ppp/modems`. This parameter is set to `none` for a null modem configuration.

`call_setup` *call_type*
This parameter is always set to `dial` for client configurations.

Dialing Information

Each remote server is identified by the keyword `remote_host`. For example:

```
remote_host miles
    phone_number           123456789
    chat_script            miles.scr
```

The most common keywords that may appear in this part of your link configuration file are:

`remote_host` *name*
Indicates the start of a remote host definition, and associates it with one of the paths defined in the PPP path configuration file (`ppp.conf`).

`phone_number` *number*

Specifies the telephone number used to call the remote host. This can be an extension number if the call is within the same private branch exchange, and must include any special digits required to pass outside a private branch exchange. The telephone number can consist of digits and characters, including special characters such as # and *. A dummy telephone number is assigned for null modem configurations.

`chat_script` *filename*

Specifies the name of the file that contains the CHAT script for this connection. By default, the connect scripts for remote hosts are located in the directory `/etc/opt/SUNWconn/ppp/script`.

Editing CHAT Scripts

The CHAT script defines the dialog that occurs between the client and the server during the connection phase. A simple non-interactive CHAT script uses `send` and `expect` keywords to specify the character strings exchanged, as shown in the following example:

Start dialog

Response expected from server

Login id sent to server

Response expected from server

Login password sent to server

```
#
# Chat script for ppp login
#

#
# Set the line regarding the remote site configuration
# Due to UUCP limitations some systems only accept cs7
#
# setline cs7 parodd

send    RETURN
expect  "ogin:"  10  onerror send BREAK repeat 3

send    "ppp1"
expect  "word: " 40
#
# Set the ppp password of the remote host here
#
send    "okppp1"
```

When it initiates the call, the client waits for a response from the remote server to begin the login sequence. The length of time the client waits, and the number of times it attempts to initiate the call, are defined by the following entry:

```
expect  "ogin:"  10  onerror send BREAK repeat 3
```

The expected response is `ogin`. The first figure (`10`) defines the wait period, and the second figure (`3`) defines the number of call initiation attempts. You can modify both of these parameters.

For example, to retry the call initiation once every 5 seconds for a total of 10 attempts, change the line in the file to:

```
expect  "ogin:"  5  onerror send BREAK repeat 10
```

Interactive CHAT scripts use `echo` and `read` keywords to display prompts and to acquire user input. The user input is stored as variables, which are identified by the `$` prefix. For example, an interactive version of the previous script could be:

Prompt for input from user
Read input from user
Prompt for input from user
Read input from user

```
send   RETURN
expect  "ogin:"  10  onerror send BREAK repeat 3

echo   "Enter your PPP login id: "
read   $login
send   "$login"

expect  "word: " 40

echo   "Enter your PPP password: "
read   $password
send   "$password"
```

A more complex example shows a CHAT script used to manage the interaction between the user and a dynamic challenge-response authentication system:

Prompt for input from user

Read input from user

Wait for response from server

Prompt for input from user

Wait for response from server

```
send   RETURN
expect "ID:"  10  onerror send BREAK repeat 3

echo   "Enter your user ID number: "
read   $id
send   "$id"

expect "Challenge: ${challenge,6}" 10
echo   "Enter the response for Challenge ${challenge}: "
read   $response
send   "$response"

expect "${host}:" 20
echo   "Connected to ${host}\n"
send   "ppp"
```

In this example, the script reads a user id and sends it to the server. It waits for 10 seconds for a response from the server that starts with the string `Challenge:` and then reads the next six characters which represent the challenge value.

The script then displays the challenge value, waits for the user to enter the corresponding response, and sends this to the server. If the response is accepted, the connection is completed.

The `{}` brackets are used to delimit a variable when it appears with other characters as part of a character string.

Index

T

U

V