Configuring Asynchronous Callback

You can configure the Cisco IOS software to call back an asynchronous device that dials in and requests a callback from the router, then disconnects.

For a complete description of the commands in this chapter, refer to the *Dial Solutions Command Reference*. To locate documentation of other commands that appear in this chapter, use the command reference master index or search online.

Refer to the following sections to configure asynchronous callback:

- Cisco's Implementation
- Call Back PPP Clients
- Call Back Clients Dialing In and Connecting to the EXEC Prompt
- Call Back ARA Clients
- Examples

Cisco's Implementation

Asynchronous callback is supported for the following protocols:

- Point-to-Point Protocol (PPP)
- Any device calling in and connecting to the router at the EXEC level
- AppleTalk Remote Access (ARA)

Callback is also supported on other interface types for PPP, including Integrated Services Digital Network (ISDN).

All callback sessions are returned on physical terminal (TTY) lines. ARA is supported on VTY lines, but also is supported on TTY lines if the **vty-arap** command is used. PPP, however, is supported on interfaces. Therefore, to enable PPP callback, you must issue the **autoselect ppp** command on the callback lines.

All current security mechanisms supported in the Cisco IOS software are supported by the callback facility, including the following:

- Terminal Access Controller Access Control System Plus (TACACS+)
- Challenge Handshake Authentication Protocol (CHAP) and Password Authentication Protocol (PAP) for PPP
- Per-user authentication for EXEC callback and ARA callback

The call originator must have the appropriate permissions set on the router before it can initiate a callback session.

Callback is useful for two purposes:

- Cost savings on toll calls

  For example, suppose it costs more to call from clients in Zone A to devices in Zone D than to call from Zone D to Zone A—costs are lower when devices in Zone D call back clients in Zone A.

- Consolidation and centralization of phone billing

  For example, if a corporation has 64 dial-in clients, enabling the corporation's routers to call back these clients consolidates billing. Instead of 64 phone bills, the corporation receives one bill.

Call Back PPP Clients

You can call back PPP clients that dial in to asynchronous interfaces. You can enable callback to the following two types of PPP clients:

- Clients that implement PPP callback per RFC 1570 (as an LCP negotiated extension).
- Clients that do not negotiate callback but can put themselves in answer-mode, whereby a callback from the router is accepted.

This section describes how to enable callback to each of these types of PPP clients.

Accept Callback Requests from RFC-Compliant PPP Clients

To accept a callback request from a RFC 1570-PPP compliant client, perform the following task, in interface (asynchronous) configuration mode:

Command	Purpose
ppp callback accept	Enable callback requests from RFC1570-compliant PPP clients on an asynchronous interface.

To configure the Cisco IOS software to call back the originating PPP client, refer to the section "Enable PPP Callback on Outgoing Lines" later in this chapter.

Accept Callback Requests from Non-RFC-Compliant PPP Clients Placing Themselves in Answer Mode

A PPP client can put itself in answer-mode and can still be called back by the router, even though it cannot specifically request callback. To enable callback on the router to this type of client, perform the following task in interface (asynchronous) configuration mode:

Command	Purpose
ppp callback initiate	Initiate callback requests from non-RFC 1570-compliant PPP clients on an asynchronous interface.

To configure the Cisco IOS software to call back the originating PPP client, refer to the next section, "Enable PPP Callback on Outgoing Lines."

Enable PPP Callback on Outgoing Lines

After enabling PPP clients to connect to an asynchronous interface and wait for a callback, you must place one or more TTY lines in PPP mode. Although calls from PPP clients enter through an asynchronous interface, the calls exit the client on a line placed in PPP mode.

To enable PPP client callback on outgoing TTY lines, perform the following steps beginning in global configuration mode:

Step	Command	Purpose
1	**chat-script** *script-name expect-send*	Define a chat script to be applied when a PPP client requests callback.
2	**username** *name* [**callback-dialstring** *telephone-number*]	Specify a per-username callback dial string.
3	**username** *name* [**callback-rotary** *rotary-group-number*]	Specify a per-username rotary group for callback.
4	**username** *name* [**callback-line** [**tty**] *line-number* [*ending-line-number*]]	Specify a per-username line or set of lines for callback.
5	**line** [**tty**] *line-number* [*ending-line-number*]	Enter line configuration mode.
6	**autoselect ppp**	Configure automatic PPP startup on a line or set of lines.
7	**login** {**authentication** | **local**}	Enable authentication on the line.
8	**script callback** *regexp*	Apply a chat script to a line or set of lines.
9	**callback forced-wait** *number-of-seconds*	Delay the callback for client modems that require a rest period before receiving a callback.

A client can issue a callback dial string; that dial string is used *only* if the dial string on the router is specified as NULL, or is not defined.

The recommended PPP chat script follows:

```
chat-script name ABORT ERROR ABORT BUSY "" "ATZ" OK "ATDT \T" TIMEOUT 30 CONNECT \c
```

Note Normally a router avoids line and modem noise by clearing the initial data received within the first one or two seconds. However, when the autoselect PPP feature is configured, the router flushes characters initially received and then waits for more traffic. This flush causes time out problems with applications that send only one carriage return. To ensure that the input data sent by a modem or other asynchronous device is not lost after line activation, enter the **no flush-at-activation** line configuration command.

Call Back Clients Dialing In and Connecting to the EXEC Prompt

You can call back clients that dial in to a TTY line and connect to the EXEC prompt. To enable callback, perform the following tasks, beginning in global configuration mode:

Step	Command	Purpose
1	**service exec-callback**	Enable EXEC callback.
2	**chat-script** *script-name expect-send*	Define a chat script to be applied when clients dial in to the EXEC prompt.

Step	Command	Purpose
3	**username** *name* [**callback-dialstring** *telephone-number*]	Specify a per-username callback dial string.
4	**username** *name* [**callback-rotary** *rotary-group-number*]	Specify a per-username rotary group for callback.
5	**username** *name* [**callback-line** [**aux** | **tty**] *line-number* [*ending-line-number*]]	Specify a per-username line or set of lines for callback.
6	**username** *name* [**nocallback-verify**]	Do not require authentication on EXEC callback.
7	**line** [**tty**] *line-number* [*ending-line-number*]	Enter line configuration mode.
8	**script callback** *regexp*	Apply a chat script to the line or a set of lines.
9	**callback forced-wait** *number-of-seconds*	Delay the callback for client modems that require a rest period before receiving a callback.

The recommended EXEC chat script is as follows:

```
chat-script name ABORT ERROR ABORT BUSY "" "ATZ" OK "ATDT \T" TIMEOUT 30 CONNECT \c
```

For an example of calling back clients connecting to the EXEC facility, see the "Call Back Clients Connecting to the EXEC Prompt Example" section later in this chapter.

Call Back ARA Clients

You can call back ARA clients. Perform the following steps, starting in global configuration mode. These steps assume you have already enabled AppleTalk routing and enabled ARA.

Step	Command	Purpose
1	**arap callback**	Enable callback to an ARA client.
2	**chat-script** *script-name expect-send*	Define a chat script to be applied when an ARA client connects to a TTY line and requests callback.
3	**line** [**tty**] *line-number* [*ending-line-number*]	Enter line configuration mode.
4	**arap enable**	Enable ARA on the line.
5	**autoselect arap**	Configure automatic protocol startup on the line.
6	**login** {**authentication** | **local**}	Enable authentication on the line.
7	**script arap-callback** *regexp*	Apply an ARA-specific chat script to a line or set of lines.
8	**callback forced-wait** *number-of-seconds*	Delay the callback for client modems that require a rest period before receiving a callback.
9	**exit**	Exit to global configuration mode.
10	**username** *name* [**callback-dialstring** *telephone-number*]	Specify a per-username callback dial string.
11	**username** *name* [**callback-rotary** *rotary-group-number*]	Specify a per-username rotary group for callback.
12	**username** *name* [**callback-line** [**tty**] *line-number* [*ending-line-number*]]	Specify a per-username line or set of lines for callback.

The recommended ARA chat script follows. The parts of the string that are **bolded** are vendor-specific extensions on the Telebit 3000 modem to disable error control. Refer to the manual for your modem for the specific commands to disable error correction for ARA.

```
chat-script name ABORT ERROR ABORT BUSY "" "ATZ" OK "ATS180=0" OK "ATS181=1" OK "ATDT
\T" TIMEOUT 60 CONNECT \c
```

For an example of calling back a PPP client, see the "Call Back a PPP Client Example" section at the end of this chapter.

Examples

The following sections provide examples for callback:

- Call Back Clients Connecting to the EXEC Prompt Example
- Call Back an ARA Client Example
- Call Back a PPP Client Example

Call Back Clients Connecting to the EXEC Prompt Example

The following example shows the process to configure an outgoing callback on the same line as the incoming request. The **login local** command enables local username authentication on lines 4 and 7. Re-authentication is required upon reconnection.

```
service exec-callback
username milarepa callback-dialstring "" password letmein
line 4
 login local
line 7
 login local
```

Call Back an ARA Client Example

The following example shows the process of configuring callback to an ARA client on line 7. The **login local** command enables local username authentication on lines 4 and 7. Line 7 will always be used for ARA callback, whether the incoming call enters line 4, 7, or 8.

```
appletalk routing
arap callback
arap network 422 router test
username excalibur callback-dialstring "123456" callback-line 7 password guenivere
line 4
 login local
 modem InOut
 autoselect arap
 arap enable
line 7
 login local
 modem InOut
 autoselect arap
 arap enable
line 8
 login local
 modem InOut
 autoselect arap
 arap enable
```

Call Back a PPP Client Example

The following example shows the process of configuring callback to a PPP client on rotary 77. PAP authentication is enabled for PPP on the asynchronous interfaces. The **login local** command enables local username authentication on lines 7, 8, and 9. The remote PPP client's host name is Ted, and the callback number is fixed at 1234567.

```
username Ted callback-dialstring "1234567" callback-rotary 77
            password Rhoda
interface async7
 ip unnumbered ethernet0
 encapsulation ppp
 no keepalive
 async default ip address 1.1.1.1
 async mode interactive
 ppp callback accept
 ppp authentication pap

interface async8
 ip unnumbered ethernet0
 encapsulation ppp
 no keepalive
 async default ip address 1.1.1.2
 async mode interactive
 ppp callback accept
 ppp authentication pap

interface async9
 ip unnumbered ethernet0
 encapsulation ppp
 no keepalive
 async default ip address 1.1.1.3
 async mode interactive
 ppp callback accept
 ppp authentication pap

line 7
 login local
 modem InOut
 rotary 77
 autoselect ppp

line 8
 login local
 modem InOut
 rotary 77
 autoselect ppp

line 9
 login local
 modem InOut
 rotary 77
 autoselect ppp
```