

Preparing for Asynchronous DDR

This chapter describes the preparations to be completed before you configure an asynchronous interface for DDR.

Complete the preparations in the following sections:

- Configure the Line
- Create Chat Scripts
- Specify Chat Scripts to Use on a Line

For a complete description of the global and line configuration commands in this chapter, refer to the “Asynchronous DDR Preparation Commands” chapter of 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.

Configure the Line

Before you configure an asynchronous interface to support DDR, configure the line as follows:

- Specify line speed.
- Set flow control on the line, if any.
- Specify the type of modem used on the line.

For detailed information about these tasks, see the “Configuring Modem Support and Asynchronous Devices” chapter in this manual.

You may also need to configure the asynchronous interface for remote node access using PPP. See the “Configuring Asynchronous PPP and SLIP” chapter for information.

Create Chat Scripts

You must create one or more chat scripts before you configure the asynchronous interface.

A chat script is a one-line command that is used on an asynchronous interface to send commands for modem dialing and for logging on to remote systems. Chat scripts indicate the possible responses to expect and the information to send in each case. You can create a different chat script for each type of modem in use on the router and for each system the router might need to log in to.

Chat scripts are required for dialing out on the asynchronous interface on the router’s auxiliary port, but are also used on other asynchronous interfaces on access servers.

To create a chat script, perform the following task in global configuration mode:

Command	Purpose
chat-script <i>script-name expect send</i>	Create a script that will place a call on a modem or log on to a remote system or both.

We recommend that you write one chat script (a “dialer” chat script) for placing a call and another one (a “login” chat script) to log on to remote systems, where required.

For an example of how to use chat scripts, see “Using Chat Scripts Example” later in this chapter and refer to the “Configuring Modem Support and Asynchronous Devices” chapter in this manual.

For information about the recommended chat script naming convention, see the **chat-script** command in the *Dial Solutions Command Reference*.

Adhering to the recommended naming convention allows you to specify a range of chat scripts by using partial names in UNIX-style regular expressions. The regular expressions are used to match patterns and select chat scripts to use. This method is particularly useful for dialer rotary groups on an interface that dials multiple destinations. Regular expressions are described in the “Regular Expressions” appendix in the *Dial Solutions Command Reference*.

Specify Chat Scripts to Use on a Line

After a chat script has been defined, it must be applied to a line or an interface before it can be used. To specify a chat script for a line, perform the following task in line configuration mode:

Command	Purpose
script dialer <i>regexp</i>	Specify a modem script for a line.

A maximum of one **script dialer** command can be configured per line. The chat script naming convention described in the “Modem Support and Chat Scripts” appendix allows you to specify a chat script by the type of the modem attached to that line as follows:

script dialer *modem-type**

We recommend that one chat script (a “dialer” chat script) be written for placing a call and another chat script (a “system” or “login” chat script) be written to log in to remote systems, where required.

You can also assign chat scripts to asynchronous interfaces for purposes other than DDR. For more information, refer to the chapter “Configuring Modem Support and Asynchronous Devices” in this manual.

Legacy DDR Configuration Examples

The examples provided in this section show various DDR configurations as follows:

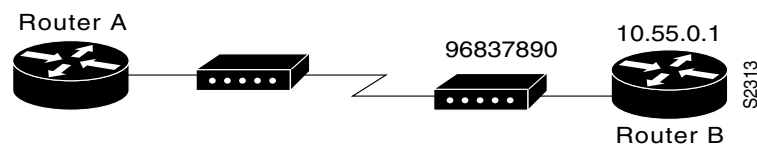
- Using Chat Scripts Example
- Writing and Implementing Chat Scripts Example
- Chat Scripts and Dialer Mapping Example
- System Login Scripts and Modem Scripts Example
- Set Up Point-to-Point DDR without Authentication Example
- Set Up Point-to-Point DDR with Authentication Example

Using Chat Scripts Example

Figure 106 shows the following configuration:

- The configuration is on Router A.
- The modem chat script *dial* is used to dial out to Router B's modem.
- The system login chat script *login* is used to log in to Router B.
- The phone number is the number of the modem attached to Router B.
- The IP address in the **dialer map** command is the address of Router B.

Figure 106 Chat Script Configuration and Function



```
chat-script dial ABORT ERROR "" "AT Z" OK "ATDT \T" TIMEOUT 30 CONNECT \c
chat-script login ABORT invalid TIMEOUT 15 name: billw word: wewpass ">"
"slip default"
interface async 10
dialer in-band
dialer map ip 10.55.0.1 modem-script dial system-script login 96837890
```

Writing and Implementing Chat Scripts Example

In the following example chat script, a pair of empty quotation marks (“”) means *expect anything* and \r means *send a return*:

```
" " \r "name:" "myname" "ord:." "mypassword" ">" "slip default"
```

The following example shows a configuration in which, when there is traffic, a random line will be used. The dialer code will try to find a script that matches both the modem script `.*-v32` and the system script `cisco`. If there is no match for both the modem script and the system script, you will see a “no matching chat script found” message.

```
interface dialer 1
! v.32 rotaries are in rotary 1
dialer rotary-group 1
! Use v.32 generic script
dialer map ip 1.0.0.1 modem-script .*-v32 system-script cisco 1234
```

The following example shows line chat scripts being specified for lines connected to Telebit and US Robotics modems:

```
! Some lines have telebit modems
line 1 6
script dialer telebit.*
! Some lines have US robotics modems
line 7 12
script dialer usr.*
```

Chat Scripts and Dialer Mapping Example

The following example shows a modem chat script called *dial* and a system login chat script called *login*. The **dialer in-band** command enables DDR on asynchronous interface 10, and the **dialer map** command dials 96837890 after finding the specified dialing and the system login scripts.

```
chat-script dial ABORT ERROR "" "AT Z" OK "ATDT \T" TIMEOUT 30 CONNECT \c
chat-script login ABORT invalid TIMEOUT 15 name: myname word: mypassword ">"
                    "slip default"
interface async 10
dialer in-band
dialer map ip 10.55.0.1 modem-script dial system-script login 96837890
```

When a packet is received for 10.55.0.1, the first thing that happens is that the modem script is implemented. Table 26 shows the functions that are implemented with each *expect-send* pair in the modem script called *dial*.

Table 26 Example Modem Script Execution

Expect and Send Pair	Implementation
ABORT ERROR	End the script execution if the text “ERROR” is found. (You can have as many active abort entries as you like.)
“ ” “AT Z”	Without expecting anything, send an “AT Z” command to the modem. (Note the use of quotation marks to allow a space in the send string.)
OK “ATDT \T	Wait to see “OK.” Send “ATDT 96837890.”
TIMEOUT 30	Wait up to 30 seconds for next expect string.
CONNECT \c	Expect “connect,” but do not send anything. (Note that \c is effectively nothing; “ ” would have indicated nothing followed by a carriage return.)

After the modem script is successfully executed, the system login script is executed. Table 27 shows the functions that are executed with each *expect-send* pair in the system script called *login*.

Table 27 Example System Script Execution

Expect and Send Pair	Implementation
ABORT invalid	End the script execution if the message “invalid username or password” is displayed.
TIMEOUT 15	Wait up to 15 seconds.
name: myname	Look for “name:” and send “billw.” (Using just “name:” will help avoid any capitalization issues.)
word: mypassword	Wait for “word:” and send the password.
“>” “slip default”	Wait for the <i>ts</i> prompt and put the line into SLIP mode with its default address.

System Login Scripts and Modem Scripts Example

The following example shows the use of chat scripts implemented with the **system-script** and **modem-script** options of the **dialer map** command.

If there is traffic for IP address 1.2.3.4, the router will dial the 91800 number using the *usrobotics-v32* script, matching the regular expression in the modem chat script. Then the router will run the *unix-slip* chat script as the system script to log in.

If there is traffic for 4.3.2.1, the router will dial 8899 using *usrobotics-v32*, matching both the modem script and modem chat script regular expressions. The router will then log in using the *cisco-compressed* script.

```

! Script for dialing a usr v.32 modem:
chat-script usrobotics-v32 ABORT ERROR "" "AT Z" OK "ATDT \T" TIMEOUT 30 CONNECT \c
!
! Script for logging into a unix system and starting up slip:
chat-script unix-slip ABORT invalid TIMEOUT 15 name: billw word: wewpass ">"
        "slip default"
!
! Script for logging into a cisco comm server and starting up TCP header
! compression
chat-script cisco-compressed...
!
line 15
    script dialer usrobotics-*
!
interface async 15
    dialer map ip 1.2.3.4 system-script unix-slip 918005551212
    dialer map ip 4.3.2.1 modem-script *-v32 system-script cisco-compressed 8899

```

Set Up Point-to-Point DDR without Authentication Example

This example shows chat scripts used in a configuration that sets up two-way reciprocal dial-on-demand routing (DDR) without authentication; the client and server have dial-in access to each other.

Remote Configuration

The following example configuration is performed on the remote side of the connection:

```

interface ethernet 0
    ip address 172.30.44.1 255.255.255.0
!
interface async 7
    ip address 172.30.45.2 255.255.255.0
    async mode dedicated
    peer default ip address 172.30.45.1
    encapsulation ppp
    dialer in-band
    dialer string 1234
    dialer-group 1
!
ip route 172.30.43.0 255.255.255.0 async 7
ip default-network 172.30.0.0
chat-script generic ABORT BUSY ABORT NO ## AT OK ATDT\T TIMEOUT 30 CONNECT
dialer-list 1 protocol ip permit
!
line 7
    no exec
    modem InOut
    speed 38400
    flowcontrol hardware
    script dialer generic

```

Local Configuration

The following example configuration is performed on the local side of the connection:

```
interface ethernet 0
 ip address 172.30.43.1 255.255.255.0
!
interface async 7
 async mode dedicated
 peer default ip address 172.30.45.2
 encapsulation ppp
 dialer in-band
 dialer string 1235
 dialer rotary-group 1
!
interface async 8
 async mode dedicated
 peer default ip address 172.30.45.2
 dialer rotary-group 1
!
ip route 172.30.44.0 255.255.255.0 async 7
 ip address 172.30.45.2 255.255.255.0
 encapsulation ppp
 ppp authentication chap
 dialer in-band
 dialer map ip 172.30.45.2 name remote 4321
 dialer load-threshold 80
!
ip route 172.30.44.0 255.255.255.0 128.150.45.2
 chat-script generic ABORT BUSY ABORT NO ## AT OK ATDT\T TIMEOUT 30 CONNECT
 dialer-list 1 protocol ip permit
!
route igrp 109
 network 172.30.0.0
 redistribute static
 passive-interface async 7
!
line 7
 modem InOut
 speed 38400
 flowcontrol hardware
 script dialer generic
```

Set Up Point-to-Point DDR with Authentication Example

This example shows chat scripts used in a configuration that sets up two-way dial-on-demand routing (DDR) with authentication; the client and server have dial-in access to each other. This configuration is demonstrated in the following two subsections.

Remote Configuration

The following example configuration is performed on the remote side of the connection. It provides authentication by identifying a password that must be provided on each end of the connection.

```
username local password secret1
username remote password secret2
interface ethernet 0
 ip address 172.30.44.1 255.255.255.0
!
interface async 7
 ip address 172.30.45.2 255.255.255.0
 async mode dedicated
 peer default ip address 172.30.45.1
```

```
encapsulation ppp
dialer in-band
dialer string 1234
dialer-group 1
!
ip route 172.30.43.0 255.255.255.0 async 7
ip default-network 172.30.0.0
chat-script generic ABORT BUSY ABORT NO ## AT OK ATDT\T TIMEOUT 30 CONNECT
dialer-list 1 protocol ip permit
!
line 7
no exec
modem InOut
speed 38400
flowcontrol hardware
script dialer generic
```

Local Configuration

The following example configuration is performed on the local side of the connection. As with the remote side configuration, it provides authentication by identifying a password for each end of the connection.

```
username remote password secret1
username local password secret2
!
interface ethernet 0
ip address 172.30.43.1 255.255.255.0
!
interface async 7
async mode dedicated
peer default ip address 172.30.45.2
dialer rotary-group 1
!
interface async 8
async mode dedicated
peer default ip address 172.30.45.2
dialer rotary-group 1
!
interface dialer 1
ip address 172.30.45.2 255.255.255.0
encapsulation ppp
ppp authentication chap
dialer in-band
dialer map ip 172.30.45.2 name remote 4321
dialer load-threshold 80
!
ip route 172.30.44.0 255.255.255.0 172.30.45.2
chat-script generic ABORT BUSY ABORT NO ## AT OK ATDT\T TIMEOUT 30 CONNECT
!
route igrp 109
network 172.30.0.0
redistribute static
passive-interface async 7
!
line 7
modem InOut
speed 38400
flowcontrol hardware
script dialer generic
```

