CHAPTER 7

Dial Troubleshooting

The troubleshooting approach in this chapter demonstrates the layer-by-layer and phase-by-phase approach to dial issues. It is important to be systematic and go step-by-step, starting with the physical layer from both ends. Unlike service provider-based remote access solutions, both ends of the connection are available for troubleshooting in an enterprise remote access scenario. This chapter focuses on the following main topics:

- Detailed troubleshooting network access server (NAS) wide-area network (WAN) links, including T1 circuits and PRIs
- Troubleshooting dial-in services
- Troubleshooting dial-out services
- Specific commands and debugs for NAS 5x00 routers, including Cisco 5200, 5300, and 5400

With this chapter, you can enhance your knowledge about T1s and PRIs. This information is relevant to all parts of this book and should be reviewed as necessary. The dial-in and dial-out troubleshooting techniques are explained in detail, from the viewpoint of the NAS, which gives you the best understanding about the nature of the processes. Troubleshooting the modem operational status is demonstrated in great detail. The specifics of the different platforms provided in this chapter should help you to improve your skill set in dial technology.

Troubleshooting NAS WAN Links

A NAS is the key piece of equipment that accepts incoming calls, authenticates the caller, and routes traffic for each call. The NAS is connected to the telephone company cloud by using a variety of leased lines that range from Basic Rate Interfaces (BRIs) to digital service 3s (DS3s). Typical design solutions include T1s and Primary Rate Interfaces (PRIs), which is why more detailed explanations about troubleshooting them are included in this section.

Troubleshooting T1 Circuits

When you notice a problem with a T1 line, it's recommended that you start troubleshooting with the core end to determine where the problem exists. Start with the physical layer first.

Controller and Line Status

Start with the physical condition and integrity of the T1 line. Always check the status of the T1 controllers and verify that you do not receive any errors. The command that displays the status is **show controllers t1**, as shown in Example 7-1.

Example 7-1 *Output of the Command* **show controllers t1**

```
5300-dial#show controllers t1 0
! The circuit is up according to the next line:
T1 0 is up.
  Applique type is Channelized T1
! Cable length is shown next. If the circuit is not long, it needs to
! be configured differently:
  Cablelength is long gain36 0db
  Description: T1 with Pacific Bell.
! Alarms are shown in the next line.
! If an alarm exists, there is a major problem in the circuit:
  No alarms detected.
  Version info of slot 0:  HW: 1, PLD Rev: 11
  Framer Version: 0x8

Manufacture Cookie Info:
 EEPROM Type 0x0001, EEPROM Version 0x01, Board ID 0x48,
 Board Hardware Version 1.0, Item Number 800-3883-01,
 Board Revision A0, Serial Number 11692119,
 PLD/ISP Version 0.1,  Manufacture Date 6-May-1999.

  Framing is ESF, Line Code is B8ZS, Clock Source is Line Secondary.
! The following 4 lines show that no errors occurred in the last 10 seconds:
  Data in current interval (10 seconds elapsed):
     0 Line Code Violations, 0 Path Code Violations
     0 Slip Secs, 0 Fr Loss Secs, 0 Line Err Secs, 0 Degraded Mins
     0 Errored Secs, 0 Bursty Err Secs, 0 Severely Err Secs, 0 Unavail Secs
! The following 4 lines show a mostly clean line, but there
! were 9 Slip Seconds and 9 Errored Seconds in the last 24 hours:
  Total Data (last 24 hours)
     0 Line Code Violations, 0 Path Code Violations,
     9 Slip Secs, 0 Fr Loss Secs, 0 Line Err Secs, 0 Degraded Mins,
     9 Errored Secs, 0 Bursty Err Secs, 0 Severely Err Secs, 0 Unavail Secs
```

The first line shows line status and tells you if the T1 is either up, down, or administratively down. The first section of the output also alerts you of any alarms that are on the circuit. On the next line, you find your configured cable length. Make sure that this is configured correctly. If the observed output is not correct, the following options are available under the controller configuration mode:

```
5300-dial(config-controller)#cablelength {long [gain26 | gain36]
    [0db | -7.5db | -15db | -22.5db] | short [133 | 266 | 399 | 533 | 655]}
```

Cable length is defined as either long or short with long being anything over 655 feet in length. Length is the distance of the entire circuit from the closest repeater or switch to the NAS. In most cases, the closest repeater is not local, so you use the default **long** with a gain or a boost of 36 decibels (**gain36**). For shorter long cable lengths, the signal is stronger and you might only need **gain26**. The second part of defining a long cable length is transmitting attenuation. The default is **0db**, which is the strongest transmit signal. If the circuit provider tells you that your signal is too strong, you can lower the transmit signal by using **–7.5db**, **-15db**, or **–22db**.

A short cable length is defined in distance by feet from the NAS and closest repeater. You use the value of 133 for distances that range from 1 foot to 133 feet. The value of 266 is for distances that range from 134 to 266 feet. For distances of 267 to 399 feet, use 399. Use 533 for distances from 400 to 533. Finally, use 655 for anything between 534 and 655 feet in length.

NOTE In modular routers (7200, 1600, and 1700 series), the command **show service-module serial** *slot/port* provides detailed information about the condition of the line. Examples of this command are included in Chapter 17, "Frame Relay Troubleshooting."

Loopback Features

If you have difficulty with any of Cisco's DS1 adapters or network modules with an internal CSU, you can troubleshoot by using the **loopback** command. The three main loopback modes that are configurable are diagnostic, local, and remote. Local loopback can be configured as either line or payload. Remote loopback can be configured as in-band bit-oriented code (IBOC) or Extended Superframe (ESF). Specify the loopback format using one of the following controller configuration commands:

```
5300-dial(config-controller)#loopback [diagnostic | local | remote]
5300-dial(config-controller)#loopback [local {payload | line}]
5300-dial(config-controller)#loopback [remote {esf line | iboc | esf paylaod}]
```

Check Bit Errors with a Bit Error Rate Tester

A bit error rate tester (BERT) alerts you of any issues on the line. Although a circuit can be operational and passing data, some data might be flawed, which can be detected by using a BERT.

A BERT is not an available option on every piece of Cisco hardware. If the option is not available, the proper way to perform this test is to put a tester directly on the circuit. To use a BERT to check for bit errors, if the router supports it, use the following controller configuration command:

```
5300-dial(config-controller)# bert pattern test pattern interval minutes
```

The available options for *test pattern* are the following:

- **0s**—All 0s test pattern
- **1s**—All 1s test pattern
- **2^11**—2^11-1 test pattern
- **2^15**—2^15-1 O.151 test pattern
- **2^20-O153**—2^20-1 O.153 test pattern
- **2^20-QRSS**—2^20-1 QRSS O.151 test pattern
- **2^23**—2^23-1 O.151 test pattern
- **alt-0-1**—Alternating 0s and 1s test pattern

A further explanation of the most commonly used BERT test patterns include the following:

- 2^15 is an exponential number that represents a pseudo-random repeating pattern that is 32,767 bits long.
- 2^20 is a pseudo-random repeating pattern that is 1,048,575 bits long.
- 2^23 is a pseudo-random repeating pattern that is 8,388,607 bits long.
- 0s is a pattern of all 0s.
- 1s is a pattern of all 1s.

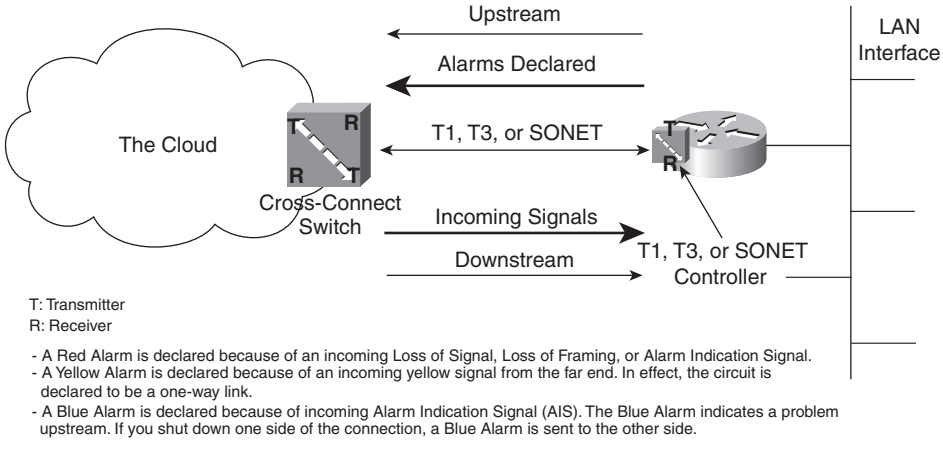
The *minutes* argument can be 1-14400, which designates the time that the BERT runs.

BERT testing can only be done over a framed T1 signal. The test cannot run if the T1 is in an alarm state where "Receiver has loss of frame (or signal)." Additionally, it can only be run on one port at a time.

Physical Layer Alarms

The alarm section of the output from **show controllers t1** is important as it tells you what type of problem might be present on the line. The presence of any alarm indicates a serious problem on the line (see Figure 7-1 and Table 7-1). In Figure 7-1, each WAN link is represented by a pair of cross-connected receivers and transmitters.

Figure 7-1 *A Functional Model of Physical Layer Alarm Messages*



When you have a T1 in an alarm state, verify that the framing and linecoding parameters are configured correctly. A common message in the alarm field is "receiver has loss of frame." Some routers also report a loss of frame (LOF) even when it should be a loss of signal (LOS). Therefore, ensure whenever you receive these errors that the T1 signal is present and the framing is correct. There are three types of alarms:

- **Blue alarm**—Another message you might receive is that the receiver is getting an Alarm Indication Signal (AIS), which means that a blue alarm indication signal is received. This generally indicates that there is a problem upstream. This is a framed or unframed all-ones signal, in both SF and ESF formats, which is transmitted to maintain transmission continuity. It typically occurs when the far-end channel service unit (CSU) has lost its terminal side equipment. For example, if you shut down your side of the connection, a blue alarm is sent to the remote side.
- **Yellow alarm**—The receiver has a remote alarm that indicates the presence of a yellow alarm. This means that the downstream CSU is in a LOF or LOS state. It is also a remote site alarm indication (RAI). When the receiver experiences LOS, the transmitter sends a yellow alarm. For SF-formats, a remote alarm is declared when bit 6 on all channels is set to 0 for at least 35 seconds. The alarm is cleared if the same bit is non-zero for 5 seconds or less (usually 1 second). When the format is ESF, a remote alarm indicates if the yellow alarm pattern exists in at least seven out of ten continuous 16-bit intervals. The alarm is cleared if this condition no longer exists for the same time intervals.
- **Red alarm**—Another typical failure is called a red alarm. A red alarm is usually indicated on the opposite end of the yellow alarm. The red alarm means that the receiver experiences LOS, LOF, or an AIS. A LOS failure is defined in RFC 1406 as "is declared upon observing 175 +/- 75 contiguous pulse positions with no pulses of either positive or negative polarity. The LOS failure is cleared upon observing an

average pulse density of at least 12.5% over a period of 175 +/- 75 contiguous pulse positions starting with the receipt of a pulse. For E1 links, the LOS failure is declared when greater than 10 consecutive zeroes are detected." After a red alarm is declared, the device sends a yellow signal to the far end. When the far end receives the yellow signal, it declares a yellow alarm. This message is accompanied by a "receiver has loss of frame" message.

TIP Always verify the framing and T1 signal when troubleshooting.

NOTE Regardless of the leased line connection that you are troubleshooting, whether a T1, T3, or Synchronous Optical Network (SONET), the alarm signals and their interpretations remain the same. The alarm states are always based on the presence or lack of signals or certain patterns. This is why the alarm states in this section apply to all circuits in the book.

Linecode Violations

These violations occur when either a bipolar violation (BPV) or excessive zero error event is present. BPVs are inserted as a means of synchronizing circuits with bipolar 8-zero substitution (B8ZS) linecoding. Linecode errors occur when BPVs that are not used for synchronization are received. Excessive zero errors occur when eight or more 0s in a row are received on a circuit where alternate mark inversion (AMI) linecoding is being used. The errors might occur because of an AMI/B8ZS configuration problem or there might be points along the transmission path that do not have all the linecoding parameters set correctly.

Pathcode Violations

Two examples for pathcode violations are frame synchronization errors for SF and cyclic redundancy check (CRC) errors for ESF. Typically, both pathcode violations and linecode violations are present simultaneously, so always that verify the linecoding is correct. Some smartjacks (and mux equipment) might need to be specifically configured for AMI/B8ZS because of problems with automatic linecode detection. Be aware that some amount of errors on your T1 can occur because of impulse noise; therefore, the errors might appear only a few times a day and the effects might be miniscule.

Slip Seconds

The presence of slips on a T1 line indicates a clocking problem. The network provides the clocking with which the customer premises equipment (CPE) must synchronize. If you see slips on the line, verify that you are deriving your clocking from the telephone company (telco) (clock source line). It is possible that only one side of the T1 is experiencing errors, so contact the provider to ensure that they are not seeing errors on their side of the circuit.

T1 Errors

RFC 1232 defines the managed objects for the DS1 interface type and standardizes the DS1 terminology and descriptions of error conditions on a T1 or E1 circuit. Table 7-1 shows the RFC 1232 categories, errors, and their descriptions.

Table 7-1 *RFC 1232 Categories, Errors, and Their Descriptions*

Object	Description
Out of Frame event	Declared when the receiver detects two or more framing-bit errors within a three millisecond period, or two or more errors out of five or less consecutive framing-bits. At this time, the framer enters the Out of Frame state, and starts searching for a correct framing pattern. The Out of Frame state ends when reframe occurs.
Loss of Signal	This event is declared upon observing 175 +/- 75 contiguous pulse positions with no pulses of either positive or negative polarity (also called keepalive).
Code Violation Error Event	The occurrence of a received CRC code that is not identical to the corresponding locally calculated code.
Bipolar Violation	A BPV for B8ZS-coded signals is the occurrence of a received BPV that is not part of a zero-substitution code. It also includes other error patterns such as eight or more consecutive 0s, and incorrect parity.
Errored Seconds	A second with one or more Code Violation Error events or one or more Out of Frame events. The presence of BPVs also triggers an Errored Second.
Severely Errored Seconds	A second with 320 or more Code Violation Error events or one or more Out of Frame events.
Severely Errored Framing Second	A Severely Errored Framing Second is a second with one or more Out of Frame events.
Unavailable Signal State	Declared at the onset of 10 consecutive Severely Errored Seconds. It is cleared at the onset of ten consecutive seconds with no Severely Errored Seconds.
Unavailable Seconds	Calculated by counting the number of seconds that the CSU is in the Unavailable Signal State, including the initial ten seconds to enter the state, but not including the ten seconds to exit the state.
Yellow Alarm	Declared because of an incoming Yellow signal from the far end. In effect, the circuit is declared to be a one-way link.
Red Alarm	Declared because of an incoming LOS, LOF, or an AIS. After a Red Alarm is declared, the device sends a yellow signal to the far end. When the far end receives the yellow signal, it declares a Yellow Alarm.
Circuit Identifier	This is a character string that is specified by the circuit vendor, and is useful when communicating with the vendor during the troubleshooting process.

Troubleshooting PRI Circuits

A PRI is actually an ISDN connection that uses a type of signaling to channelize a T1 without robbing signaling bits from each channel; instead, the signaling is done on the last (24^{th}) channel. The first step in troubleshooting problems with a PRI is to use the **show isdn status** command as shown in Example 7-2.

Example 7-2 *Output of* **show isdn status**

```
5300-dialin#show isdn status
Global ISDN Switchtype = primary-5ess
ISDN Serial0:23 interface
        dsl 0, interface ISDN Switchtype = primary-5ess
    Layer 1 Status:
        ACTIVE
    Layer 2 Status:
        TEI = 0, Ces = 1, SAPI = 0, State = MULTIPLE_FRAME_ESTABLISHED
    Layer 3 Status:
        0 Active Layer 3 Call(s)
<output omitted>
```

The following sections explain this output.

Layer 1 Status

The Layer 1 status portion of the output shows whether the T1 access circuit that the PRI signaling rides on is up or not. If it is not in an active state, go to the beginning of this chapter and troubleshoot the T1 portion of the circuit first.

Layer 2 Status

If the Layer 2 state does not show MULTIPLE_FRAME_ESTABLISHED, check the T1 circuit for incrementing errors and treat this situation as any T1 problem. More information about troubleshooting measures is covered in the section, "Troubleshooting T1 Circuits." If the T1 checks out okay, verify that the ISDN switch type and PRI group time slots were set up the same as the circuit was provisioned. Then check the serial interface associated with the PRI by using the command **show interface serial** *x***:23**, where *X* is the associated T1 port number. A sample output is shown in Example 7-3.

Example 7-3 *Output of* **show interface serial 0:23**

```
S5200-dialin>show interface serial 0:23
 Serial2:23 is up, line protocol is up (spoofing)
  Hardware is DSX1
  MTU 1500 bytes, BW 64 Kbit, DLY 20000 usec,
     reliability 128/255, txload 1/255, rxload 1/255
```

Example 7-3 *Output of* **show interface serial 0:23** *(Continued)*

```
  Encapsulation HDLC, loopback not set
  DTR is pulsed for 1 seconds on reset
  Last input 00:00:20, output never, output hang never
  Last clearing of "show interface" counters never
  Input queue: 0/75/0/0 (size/max/drops/flushes); Total output drops: 0
<output omitted>e
```

Verify that the D channel is up and not in a loopback state. If the state of Layer 2 is still not MULTIPLE_FRAME_ESTABLISHED, you need to call your service provider. However, if you receive this message but you are still experiencing problems, turn on ISDN Q921 debugs. After using **debug isdn q921** and **show debug**, ensure that the output is similar to that in Example 7-4.

Example 7-4 *Output of* **show debug** *Indicates That* **debug isdn q921** *Is On*

```
5300-dialin#show debug
ISDN:
  ISDN Q921 packets debugging is on
  ISDN Q921 packets debug DSLs. (On/Off/No DSL:1/0/-)
  DSL  0 --> 7

  1 1 1 1 1 1 1 1
```

If you are not directly connected to the console port, you have to enable **terminal monitor** to see the output from debug commands.

Also, verify that the only activity you see is similar to that in Example 7-5.

Example 7-5 *Debug Output Showing ISDN q921 Activity*

```
5300-dialin#terminal monitor
Jan  6 03:44:01.653: ISDN Se0:23: RX <-  RRf sapi = 0  tei = 0  nr = 4
Jan  6 03:44:05.669: ISDN Se1:23: TX ->  RRp sapi = 0  tei = 0 nr = 113
Jan  6 03:44:05.677: ISDN Se1:23: RX <-  RRf sapi = 0  tei = 0  nr = 1
Jan  6 03:44:14.981: ISDN Se2:23: TX ->  RRp sapi = 0  tei = 0 nr = 0
Jan  6 03:44:14.989: ISDN Se2:23: RX <-  RRf sapi = 0  tei = 0  nr = 0
Jan  6 03:44:16.169: ISDN Se3:23: TX ->  RRp sapi = 0  tei = 0 nr = 79
Jan  6 03:44:16.185: ISDN Se3:23: RX <-  RRf sapi = 0  tei = 0  nr = 79
```

A service access point identifier (SAPI) has a value assigned to it. This value determines the data type coming from the device at the other end. The data types are as follows:

- **0**—Q931 (signaling information)
- **1**—Telemetry
- **16**—X.25 on the D channel
- **63**—Data-link management

A terminal endpoint identifier (TEI) is an address used at Layer 2 that manages individual devices that are connecting to the ISDN network. The TEI is typically dynamically negotiated with the ISDN switch. The range is from 0 to 127. The following shows what each TEI number means:

- **0**—Point-to-point service (as it is for PRI)
- **1-63**—Fixed assigned
- **64-126**—Dynamically assigned by telco switch
- **127**—Broadcast (send frame to all attached devices)

NOTE The correct TEI for PRI is always 0.

If Set Asynchronous Balanced Mode Extended (SABME) messages appear, the switch-type or PRI time slots are set incorrectly. SABME messages in the debug appear as follows:

```
Jan  6 03:45:16.185: ISDN Se0:23: TX -> SABMEp sapi = 0 tei = 0
Jan  6 03:45:16.662: ISDN Se0:23: RX <- BAD FRAME(0x00017F)Line may be looped!
```

See Chapter 12, "ISDN BRI Troubleshooting," for additional information about the Q921 protocol.

Troubleshooting Dial-In Service

In this section, detail is provided on what you can do to determine the cause of the dial-in connection problems. The troubleshooting techniques include commands for use on a modem, outputs from debug commands, and some common recommendations. This section does not include debugs from Cisco AS5x00 series routers because this is covered in a subsequent section later in this chapter.

When troubleshooting a connection problem, try to piece together everything that is expected to occur throughout the process. Use a step-by-step approach to outline all activities from start to finish, and in the correct sequence.

You might be required to make some assumptions as part of the process. First, assume that the analog lines connected to the modems are working, their phone numbers are correct, and a call placed from one end to the other occurs successfully (the switched services work correctly). Obviously, if a problem exists with an analog line, it's a switched service problem, and your telephone provider must correct the issue. This section covers the following steps that are used to troubleshoot a problem with the NAS:

- Step One: Verify that the modem is ready to accept incoming calls.
- Step Two: Verify the type of connection.
- Step Three: Verify Point-to-Point Protocol (PPP) negotiation.

Step One: Verify that the Modem Is Ready to Accept Calls

The first event to occur is that the modem must pick up the call. If it is an external modem, increase the volume so that you can hear it. Place a call from a telephone to the modem to make sure it answers. If it does not, check the cabling between the modem and the router. The type of cable can differ between modems, so the easiest way to check is through the **show line** *line-number* command. The line you must check is the following:

```
Modem state: Idle
```

If the modem is ready to take an incoming call, but there is no call on the line, the state should be idle. If the state is anything other than idle, the modem will not answer the call. Several different signals (modem states) have to interact properly for the modem to be ready to accept a call. The Modem Hardware State are

- **CTS (Clear To Send)**—Provided by the data communications equipment (DCE). The DCE signals to the data terminal equipment (DTE) that the DCE accepts data.
- **DSR (Data Set Ready)—**Provided by the DCE.
- **DTR (Data Terminal Ready)**—Provided by the DTE. The DTE indicates to the DCE that it accept calls.
- **RTS (Request To Send)**—Provided by the DTE. The DTE signals that the DTE accepts data.

Modem Hardware State: CTS noDSR DTR RTS

In this example, the Modem Hardware State is correct, but the modem is in a ready state instead of an idle state, as indicated by noDSR.

If an active session is on the line, the modem cannot answer a new call. It displays a ready state. The **show users** command shows you if there is an active session. You can then clear the active session with the privileged command **clear line**.

The second reason for noDSR can be because modem control is not configured on the line. Configure the line with either **modem Dialin** or **modem InOut**.

Lastly, DSR might be high, which results from a cabling problem or if the modem is configured where Data Carrier Detect-Provided (DCD) is always high. Fix the cabling problem or reconfigure the modem so that DCD is only high when carrier detection (CD) is successful, which should clear the problem.

Modem Hardware State: noCTS noDSR DTR RTS

In cases where noCTS replaces CTS in the correct state, three different possibilities exist: the modem is turned off, a cabling problem exists, or hardware flow control on the modem is turned off. The line configuration command **no flowcontrol hardware** fixes the last of these three problems.

Modem Hardware State: CTS DSR DTR RTS

In cases where DSR replaces noDSR in the correct state, there can be a cable problem. Also, DCD can always be configured on the modem as high. Reconfigure the modem to correct this and ensure that either the line configuration command **modem Dialin** or **modem InOut** exists.

Now that the cable connection from router to modem is operational, you must also ensure that the modem is set to auto answer. Consult the modems owner's manual to set this up. This concludes the troubleshooting required if a modem does not answer.

Step Two: Verify Type of Incoming Connection

After step one, where the connection is a fact, perform the second step from the core router (NAS). The Cisco IOS features give you much more insight about the type of the connection than any other approach. Start the incoming type verification by using the **debug modem** command. The first debug reflects the configuration from a basic PPP dial-in service in Chapter 6, "Dial Design and Configuration Solutions."

When dialing into an external modem, one that does not have out-of-band signaling, the first line should read as follows:

```
17:11:38: TTY1: DSR came up
```

This signifies that the modem has trained up successfully. If this does not take place, there was a problem with the modem train-up. Line issues or modem incompatibilities can cause a train-up failure. To cure any modem incompatibility problem, ensure that the modems on both ends have the latest firmware and drivers.

The following line indicates a change in modem state:

```
17:11:38: tty1: Modem: IDLE->(unknown)
```

The modem in this example happens to be external, so the router does not know which state the modem changed to. If it had been an internal modem with out-of-band functionality, you would have seen the following:

```
17:11:38: tty1: Modem: IDLE->READY
```

At this point, if you have the interface configured with **async mode dedicated**, the connection immediately jumps into PPP. The PPP debugs are covered later in this section. If the line was configured for text authentication and no PPP, such as a modem in an AUX port, the debug output in Example 7-6 would be displayed.

Example 7-6 *Debug Output for Exec Creation*

```
! The router starts an exec session:
17:11:43: TTY1: EXEC creation
! The following two lines are used when prompting for authentication:
17:11:43: TTY1: set timer type 10, 30 seconds
```

Example 7-6 *Debug Output for Exec Creation (Continued)*

```
17:11:56: TTY1: set timer type 10, 30 seconds
17:11:67: TTY1: create timer type 1, 600 seconds
```

The **type 10 timer** is used for username and password prompts. The fourth line in the output sets a 30-second timer for the username prompt and the fifth line places another 30-second timer for the password prompt. The last line in the output is an exec timer. The default **exec-timeout** set on a line is 10 minutes, or 600 seconds.

In the preceding example, the modem changed state from idle to unknown. If the async interface was configured with **async mode interactive** and the line was configured with **autoselect ppp** and **autoselect during-login**, the debug is different, as shown in Example 7-7.

Example 7-7 *Debug Output of Modem Autoselect for PPP*

```
22:52:59: TTY1: EXEC creation
22:52:59: TTY1: set timer type 10, 30 seconds
22:53:01: TTY1: Autoselect(2) sample 7E
22:53:01: TTY1: Autoselect(2) sample 7EFF
22:53:01: TTY1: Autoselect(2) sample 7EFF7D
22:53:01: TTY1: Autoselect(2) sample 7EFF7D23
22:53:01: TTY1 Autoselect cmd:  ppp negotiate
```

During the username prompt, the router checks incoming characters to see if they are PPP or if they are part of a username. The autoselect samples shown in Example 7-7 are in hexadecimal format. If translated to ASCII, they show the text received over a line. There is an autoselect sample for each character that arrives. The router displays up to four characters in each new autoselect sample line and includes the three previous characters, followed by the last character entered as shown.

The four autoselect characters in Example 7-7, if translated into ASCII, are ~ÿ}#, which is the typical representation of a PPP link control protocol (LCP) packet. For this reason, the last line in the debug shows that autoselect executed the command **ppp negotiate**, which instructs the router to negotiate PPP.

The sample output in Example 7-8 uses the same configuration as before, except that the login was done through text. This can either be from typing it manually or from running a login script.

Example 7-8 *Debug Output of Modem Autoselect for Text*

```
23:15:22: TTY1: EXEC creation
23:15:22: TTY1: set timer type 10, 30 seconds
! Username entry of six characters, followed by carriage return
23:15:23: TTY1: Autoselect(2) sample 6A
23:15:24: TTY1: Autoselect(2) sample 6A68
23:15:24: TTY1: Autoselect(2) sample 6A6875
```

continues

Example 7-8 *Debug Output of Modem Autoselect for Text (Continued)*

```
23:15:24: TTY1: Autoselect(2) sample 6A687565
23:15:24: TTY1: Autoselect(2) sample 68756567
23:15:24: TTY1: Autoselect(2) sample 75656765
23:15:24: TTY1: Autoselect(2) sample 6567656E
23:15:25: TTY1: Autoselect(2) sample 67656E0D
23:15:25: TTY1: set timer type 10, 30 seconds
! The following lines state [suppressed--line is not echoing], because
! the password prompt never echoed the characters entered.
23:15:26: TTY1: Autoselect(2) sample [suppressed--line is not echoing]
23:15:26: TTY1: Autoselect(2) sample [suppressed--line is not echoing]
23:15:27: TTY1: Autoselect(2) sample [suppressed--line is not echoing]
23:15:27: TTY1: Autoselect(2) sample [suppressed--line is not echoing]
23:15:27: TTY1: Autoselect(2) sample [suppressed--line is not echoing]
```

In this case, the username entered is six characters followed by a carriage return. You can decode the incoming text for usernames by converting the hexadecimal characters in the sample to ASCII. The hexadecimal string can be pieced together to form 6A 68 75 65 67 65 6E 0D. When changed to ASCII, it spells jhuegen followed by 0D, which is a carriage return. The following lines states "[suppressed--line is not echoing]" because the password prompt never echoes the characters entered; however, you can make the assumption that the password entered was four characters followed by a carriage return (to make up the five lines).

Step Three: Verify PPP Negotiation

After the call connects, PPP negotiation starts. The following output is from **debug PPP negotiation**. It is split into the major steps, which are explained:

```
Mar  2 13:32:45.354: %LINK-3-UPDOWN: Interface Async1, changed state to up
Mar  2 13:32:45.354: As1 PPP: Treating connection as a dedicated line
Mar  2 13:32:45.354: As1 PPP: Phase is ESTABLISHING, Active Open
```

NOTE The Link Dead (physical layer not ready) transition state changes to the Link Establishment phase only if an external event, such as a carrier detect (CD), is up.

LCP Phase of PPP

The following explanations are based on the output you see if you type the **debug ppp negotiation** command to troubleshoot LCP issues. The first part of the output identifies how PPP treats the connection. For every dial case, it is treated as a dedicated line. The first step in LCP takes place when the dial server sends an outgoing configuration request (O CONFREQ), as shown in Example 7-9.

Example 7-9 *Debug Output of PPP Outgoing Configuration Request*

```
Mar  2 13:32:45.354: As1 LCP: O CONFREQ [Closed] id 33 len 24
Mar  2 13:32:45.354: As1 LCP:    ACCM 0x000A0000 (0x0206000A0000)
Mar  2 13:32:45.358: As1 LCP:    AuthProto PAP (0x0304C023)
Mar  2 13:32:45.358: As1 LCP:    MagicNumber 0xE82CFF9C (0x0506E82CFF9C)
Mar  2 13:32:45.358: As1 LCP:    PFC (0x0702)
Mar  2 13:32:45.358: As1 LCP:    ACFC (0x0802)
```

The server expects a reply and the reply should be similar to the Example 7-10, which is an incoming configuration acknowledgment (I CONFACK).

Example 7-10 *Debug Output of PPP Incoming Configuration Acknowledgment*

```
Mar  2 13:32:45.546: As1 LCP: I CONFACK [REQsent] id 33 len 24
Mar  2 13:32:45.546: As1 LCP:    ACCM 0x000A0000 (0x0206000A0000)
Mar  2 13:32:45.546: As1 LCP:    AuthProto PAP (0x0304C023)
Mar  2 13:32:45.546: As1 LCP:    MagicNumber 0xE82CFF9C (0x0506E82CFF9C)
Mar  2 13:32:45.546: As1 LCP:    PFC (0x0702)
Mar  2 13:32:45.546: As1 LCP:    ACFC (0x0802)
```

Next, the incoming configuration request (I CONFREQ) from the connecting client is received. At this point, the client tries to negotiate the callback protocol, as shown in Example 7-11.

Example 7-11 *Debug Output of PPP Request for Callback*

```
Mar  2 13:32:46.402: As1 LCP: I CONFREQ [ACKrcvd] id 2 len 50
Mar  2 13:32:46.402: As1 LCP:    ACCM 0x00000000 (0x020600000000)
Mar  2 13:32:46.406: As1 LCP:    MagicNumber 0x588D5503 (0x0506588D5503)
Mar  2 13:32:46.406: As1 LCP:    PFC (0x0702)
Mar  2 13:32:46.406: As1 LCP:    ACFC (0x0802)
Mar  2 13:32:46.406: As1 LCP:    Callback 6  (0x0D0306)
Mar  2 13:32:46.406: As1 LCP:    MRRU 1614 (0x1104064E)
Mar  2 13:32:46.406: As1 LCP:    EndpointDisc 1 Local
Mar  2 13:32:46.406: As1 LCP:     (0x131701F9358C5F03D643118C9B7AC7F0)
Mar  2 13:32:46.406: As1 LCP:     (0x70629C00000000)
```

The dial server then sends an outgoing rejection (O CONFREJ) because callback authentication is not turned on, as shown in Example 7-12.

Example 7-12 *Debug Output of PPP Rejection for Callback*

```
Mar  2 13:32:46.406: As1 LCP: O CONFREJ [ACKrcvd] id 2 len 11
Mar  2 13:32:46.406: As1 LCP:    Callback 6  (0x0D0306)
Mar  2 13:32:46.406: As1 LCP:    MRRU 1614 (0x1104064E)
```

The client then requests a new set of options. This is an incoming configuration request (I CONFREQ), as shown in Example 7-13.

Example 7-13 *Debug Output of PPP Incoming Configuration Request*

```
Mar  2 13:32:46.594: As1 LCP: I CONFREQ [ACKrcvd] id 3 len 43
Mar  2 13:32:46.594: As1 LCP:    ACCM 0x00000000 (0x020600000000)
Mar  2 13:32:46.594: As1 LCP:    MagicNumber 0x588D5503 (0x0506588D5503)
Mar  2 13:32:46.594: As1 LCP:    PFC (0x0702)
Mar  2 13:32:46.594: As1 LCP:    ACFC (0x0802)
Mar  2 13:32:46.594: As1 LCP:    EndpointDisc 1 Local
Mar  2 13:32:46.594: As1 LCP:     (0x131701F9358C5F03D643118C9B7AC7F0)
Mar  2 13:32:46.594: As1 LCP:     (0x70629C00000000)
```

The server then acknowledges this by sending an outgoing configuration acknowledgement (O CONFACK), as shown in Example 7-14.

Example 7-14 *Debug of PPP Outgoing Configuration Acknowledgment*

```
Mar  2 13:32:46.598: As1 LCP: O CONFACK [ACKrcvd] id 3 len 43
Mar  2 13:32:46.598: As1 LCP:    ACCM 0x00000000 (0x020600000000)
Mar  2 13:32:46.598: As1 LCP:    MagicNumber 0x588D5503 (0x0506588D5503)
Mar  2 13:32:46.598: As1 LCP:    PFC (0x0702)
Mar  2 13:32:46.598: As1 LCP:    ACFC (0x0802)
Mar  2 13:32:46.598: As1 LCP:    EndpointDisc 1 Local
Mar  2 13:32:46.598: As1 LCP:     (0x131701F9358C5F03D643118C9B7AC7F0)
Mar  2 13:32:46.598: As1 LCP:     (0x70629C00000000)
```

Next, LCP changes state to open. At this point, both sides agree that the server will provide a configuration for the client. LCP is then complete.

```
! The LCP is Open:
Mar  2 13:32:46.598: As1 LCP: State is Open
```

IF LCP never completes and does not change state to open, a few problems can possibly exist:

- First, LCP timeouts can be caused by a speed problem between the router and modem. A symptom of this problem is that either one or both of the peers do not see any incoming LCP packets. This occurs only if there is a speed issue between the router and an external modem.
- The second type of LCP problem is caused when both peers are not able to agree on authentication. The client and server must agree on authentication type, which is Password Authentication Protocol (PAP), Challenge Handshake Authentication Protocol (CHAP), or Microsoft CHAP (MS CHAP). For example, if the server side is set to authenticate through CHAP and the client is configured for PAP authentication, LCP times out while trying to negotiate.
- LCP can also fail because of a maximum transmission unit (MTU) mismatch. Make sure that MTU is defined as the same on both peers. If necessary, reduce the MTU on both sides until LCP succeeds.

The Authentication Phase of PPP

After LCP finishes and its state is open, the next step in the process is authentication. During this phase, you can see the authenticating party and whether or not authentication has passed. Example 7-15 shows the output of a successful PPP PAP authentication. This output is a result of the **debug ppp authentication** command.

Example 7-15 *Debug Output of a Successful PPP PAP Authentication*

```
Mar  2 13:32:46.598: As1 PPP: Phase is AUTHENTICATING, by this end
Mar  2 13:32:46.842: As1 PAP: I AUTH-REQ id 7 len 17 from "jhuegen"
Mar  2 13:32:46.846: As1 PAP: Authenticating peer jhuegen
Mar  2 13:32:46.846: As1 PPP: Phase is FORWARDING, Attempting Forward
Mar  2 13:32:46.846: As1 PPP: Phase is AUTHENTICATING, Unauthenticated User
Mar  2 13:32:46.850: As1 PPP: Phase is FORWARDING, Attempting Forward
Mar  2 13:32:46.850: As1 PPP: Phase is AUTHENTICATING, Authenticated User
Mar  2 13:32:46.850: As1 PAP: O AUTH-ACK id 7 len 5
Mar  2 13:32:46.850: As1 PPP: Phase is UP
```

Network Control Protocol Phase of PPP

When the PPP phase is up, authentication has completed successfully. Then, Network Control Protocol (NCP) negotiates Layer 3 protocols, including IP Control Protocol (IPCP). In the case of dial, IPCP negotiates IP addresses for the peer IP address, the Domain Name System (DNS) servers, and the Windows Internet Naming Service (WINS) servers. The following explanations are based on the output from the command **debug ppp negotiation**. Remember that the output can be long and you see these lines only if the previous (authentication) phase is successful. First, an outgoing configuration request is sent to the peer that contains its own IP address:

```
Mar  2 13:32:46.854: As1 IPCP: O CONFREQ [Closed] id 7 len 10
Mar  2 13:32:46.854: As1 IPCP:    Address 192.168.0.249 (0x0306C0A800F9)
```

The peer then sends a request to the NAS to do Compression Control Protocol (CCP):

```
Mar  2 13:32:47.010: As1 CCP: I CONFREQ [Not negotiated] id 6 len 10
Mar  2 13:32:47.014: As1 CCP:    MS-PPC supported bits 0x00000001 (0x120600000001)
```

CCP is then rejected by an outgoing protocol rejection (O PROTREJ) packet. The peer should not attempt to renegotiate CCP:

```
Mar  2 13:32:47.014: As1 LCP: O PROTREJ [Open] id 34 len 16 protocol CCP
 (0x80FD0106000A120600000001)
```

An incoming configuration request is received, and the peer requests VJ 15 header compression and IP addresses for the peer, including the Primary DNS, Primary WINS, Secondary DNS, and Secondary WINS servers, as shown in Example 7-16.

Example 7-16 *Debug Output of IPCP Incoming Configuration Request*

```
Mar  2 13:32:47.058: As1 IPCP: I CONFREQ [REQsent] id 7 len 40
Mar  2 13:32:47.058: As1 IPCP:    CompressType VJ 15 slots CompressSlotID
 (0x0206002D0F01)
Mar  2 13:32:47.058: As1 IPCP:    Address 0.0.0.0 (0x030600000000)
Mar  2 13:32:47.058: As1 IPCP:    PrimaryDNS 0.0.0.0 (0x810600000000)
Mar  2 13:32:47.058: As1 IPCP:    PrimaryWINS 0.0.0.0 (0x820600000000)
Mar  2 13:32:47.058: As1 IPCP:    SecondaryDNS 0.0.0.0 (0x830600000000)
Mar  2 13:32:47.058: As1 IPCP:    SecondaryWINS 0.0.0.0 (0x840600000000)
```

NOTE VJ compression is Van Jacobsen TCP header compression, which is a widely accepted compression method. See Part IV, "Frame Relay" for more information.

An outgoing configuration reject (O CONFREJ) is sent to reject VJ 15 header compression:

```
Mar  2 13:32:47.058: As1 IPCP: O CONFREJ [REQsent] id 7 len 10
Mar  2 13:32:47.058: As1 IPCP:    CompressType VJ 15 slots CompressSlotID
(0x0206002D0F01)
```

An incoming configuration acknowledgment (I CONFACK) is received and the peer acknowledges the IP address of the NAS:

```
Mar  2 13:32:47.062: As1 IPCP: I CONFACK [REQsent] id 7 len 10
Mar  2 13:32:47.062: As1 IPCP:    Address 192.168.0.249 (0x0306C0A800F9)
```

Because VJ 15 header compression was rejected, so was the request for IP addresses for the peer and those of the DNS and WINS servers. The peer then sends another configuration request packet because the first was rejected, only this time, it asks for addressing without asking for header compression, as shown in Example 7-17.

Example 7-17 *Debug Output of IPCP Incoming Configuration Request Without Compression*

```
Mar  2 13:32:47.246: As1 IPCP: I CONFREQ [ACKrcvd] id 8 len 34
Mar  2 13:32:47.246: As1 IPCP:    Address 0.0.0.0 (0x030600000000)
Mar  2 13:32:47.246: As1 IPCP:    PrimaryDNS 0.0.0.0 (0x810600000000)
Mar  2 13:32:47.246: As1 IPCP:    PrimaryWINS 0.0.0.0 (0x820600000000)
Mar  2 13:32:47.246: As1 IPCP:    SecondaryDNS 0.0.0.0 (0x830600000000)
Mar  2 13:32:47.246: As1 IPCP:    SecondaryWINS 0.0.0.0 (0x840600000000)
```

Confusion occurs when the router sends a configuration non-acknowledgment, which actually refuses the request for the peer to use 0.0.0.0 as every address. Along with this non-acknowledgment (NAK), the NAS sends the peer the IP address and those of the DNS and WINS servers that it wants the peer to use, as shown in Example 7-18.

Example 7-18 *Debug Output of Outgoing Non-Acknowledgment with Server Assigned Addressing*

```
Mar  2 13:32:47.250: As1 IPCP: O CONFNAK [ACKrcvd] id 8 len 34
Mar  2 13:32:47.250: As1 IPCP:    Address 192.168.0.251 (0x0306C0A800FB)
Mar  2 13:32:47.250: As1 IPCP:    PrimaryDNS 192.168.100.3 (0x8106C0A86403)
Mar  2 13:32:47.250: As1 IPCP:    PrimaryWINS 192.168.100.5 (0x8206C0A86405)
Mar  2 13:32:47.250: As1 IPCP:    SecondaryDNS 192.168.100.4 (0x8306C0A86404)
Mar  2 13:32:47.250: As1 IPCP:    SecondaryWINS 192.168.100.6 (0x8406C0A86406)
```

The peer responds to the NAK with yet another configuration request to use the IP addresses provided with the previous NAK. Example 7-19 shows the expected output for this exchange.

Example 7-19 *Debug Output of Incoming Request with Server Assigned Addressing*

```
Mar  2 13:32:47.446: As1 IPCP: I CONFREQ [ACKrcvd] id 9 len 34
Mar  2 13:32:47.446: As1 IPCP:    Address 192.168.0.251 (0x0306C0A800FB)
Mar  2 13:32:47.446: As1 IPCP:    PrimaryDNS 192.168.100.3 (0x8106C0A86403)
Mar  2 13:32:47.446: As1 IPCP:    PrimaryWINS 192.168.100.5 (0x8206C0A86405)
Mar  2 13:32:47.446: As1 IPCP:    SecondaryDNS 192.168.100.4 (0x8306C0A86404)
Mar  2 13:32:47.450: As1 IPCP:    SecondaryWINS 192.168.100.6 (0x8406C0A86406)
```

The NAS then acknowledges the peer as configured correctly by sending an outgoing configuration acknowledgment packet, as shown in Example 7-20.

Example 7-20 *Debug Output of Outgoing Acknowledgment with Server Assigned Addressing*

```
Mar  2 13:32:47.450: As1 IPCP: O CONFACK [ACKrcvd] id 9 len 34
Mar  2 13:32:47.450: As1 IPCP:    Address 192.168.0.251 (0x0306C0A800FB)
Mar  2 13:32:47.450: As1 IPCP:    PrimaryDNS 192.168.100.3 (0x8106C0A86403)
Mar  2 13:32:47.450: As1 IPCP:    PrimaryWINS 192.168.100.5 (0x8206C0A86405)
Mar  2 13:32:47.450: As1 IPCP:    SecondaryDNS 192.168.100.4 (0x8306C0A86404)
Mar  2 13:32:47.450: As1 IPCP:    SecondaryWINS 192.168.100.6 (0x8406C0A86406)
```

When both sides agree on the addressing, IPCP changes state to open and installs the directly connected route to the dialup peer:

```
Mar  2 13:32:47.450: As1 IPCP: State is Open
Mar  2 13:32:47.454: As1 IPCP: Install route to 192.168.0.251
Mar  2 13:32:47.454: %LINEPROTO-5-UPDOWN: Line protocol on Interface Async1, changed
  state to up
```

NOTE To troubleshoot a problem during NCP negotiation, ensure that all required IP addresses and protocols are configured.

Two of the most common issues that occur during the NCP stage of PPP negotiation are the following:

- The IP address is not configured on the group-async interface on the NAS. In most cases, you are not able to configure an IP address directly on the interface; therefore, you need to configure a loopback interface with an IP address. Use the command **ip unnumbered** *interface* on your group-async interface, where *interface* refers to your loopback interface. This instructs the group-async interface to use the IP address of the loopback.
- Verify the availability of pool IP addressing for the client. If all addresses in the pool are already allocated, NCP fails by not providing the peer with an IP address.

NOTE The typical dial oversubscription ratio for Internet service providers (ISPs) is about ten users to one DS0, and scaling beyond this number is not recommended. In the case of an enterprise environment, remote users tend to dial up more often, and this ratio can be as low as 5 to 1. Oversubscription ratios for enterprises that provide a wide variety of remote access services might find it easier to base their oversubscription rate on percentages. For example, an enterprise might try to keep an average of 40 percent of DS0s available throughout a typical day. This 40 percent is there to handle the spikes in daily usage and holidays, when dial usage is normally much higher than on average.

- Verify that DNS and WINS server IP addresses are configured to respond to BOOTP requests. Use the global configuration commands **async-bootp dns-server** *address(es)* and **async-bootp nbns-server** *address(es*) to configure this feature.

After PPP negotiation is complete, the dial connection is complete and traffic is able to pass, unless of course routing problems exist in the network.

Troubleshooting Dial-Out Service

When an outbound call is placed, the output in Example 7-21 is generated from dial-on-demand events and PPP protocol negotiation debugging. The commands that enable this debugging include **#debug dialer** and **#debug ppp negotiation**.

Example 7-21 *Debug Output of Dialer and PPP Negotiation for an Outgoing Call*

```
Mar  1 00:11:15.975: As65 DDR: place call
Mar  1 00:11:15.975: As65 DDR: Dialing cause ip (s=192.168.0.2, d=216.115.102.82)
Mar  1 00:11:15.975: As65 DDR: Attempting to dial 6222230
Mar  1 00:11:15.975: CHAT65: Attempting async line dialer script
Mar  1 00:11:15.983: CHAT65: no matching chat script found for 6222230
Mar  1 00:11:15.983: CHAT65: Dialing using Modem script: d0efault-d0ials0cript
    & System script: none
Mar  1 00:11:15.987: CHAT65: process started
Mar  1 00:11:15.987: CHAT65: Asserting DTR
Mar  1 00:11:15.987: CHAT65: Chat script d0efault-d0ials0cript started
```

To place a call, some interesting traffic must be routed out the async65 interface. In the output shown in Example 7-21, the source address 192.168.0.2 (a host on the Ethernet segment) sent a packet to 216.115.102.82, which caused the router to attempt to dial.

In most cases, you do not need a dial chat script and the default works. The default dial script simply performs the modem AT commands to cause the modem connect and then hands the connection over to the async interface. If the modem does not attempt to dial, check for cabling problems or the correct line configuration. The line must be set for **modem InOut** and the async interface requires a dialer string. Also, ensure that the phone line is active because without a dial tone the modem cannot dial.

At this point, the modem should train up. Once this is successful, the script ends and the interface changes state to up, as shown in the output in Example 7-22.

Example 7-22 *Continuation of Debug from Example 7-21*

```
Mar  1 00:11:39.731: CHAT65: Chat script d0efault-d0ials0cript finished,
    status = success
00:11:41: %LINK-3-UPDOWN: Interface Async65, changed state to up
Mar  1 00:11:41.735: As65 DDR: Dialer state change to up
Mar  1 00:11:41.735: As65 DDR: Dialer call has been placed
Mar  1 00:11:41.735: As65 PPP: Treating connection as a callout
Mar  1 00:11:41.735: As65 PPP: Phase is ESTABLISHING, Active Open
Mar  1 00:11:41.735: As65 PPP: No remote authentication for call-out
```

After the interface is up, PPP negotiation takes place. By default with PAP and CHAP, the router attempts to authenticate the remote side, even on outbound calls. However, this router is configured to only authenticate inbound calls and does not authenticate the server that it is dialing into. To configure this, the word **callin** must be added to the **ppp authentication** interface configuration command.

From this point, the PPP negotiation is exactly the opposite from the previous section because the router acts as a client, and not as the server. The dial-out router sends an outgoing configuration request to the NAS to advise it has connected, as shown in Example 7-23.

Example 7-23 *Debug Output of PPP Outgoing Configuration Request*

```
Mar  1 00:11:41.739: As65 LCP: O CONFREQ [Closed] id 12 len 20
Mar  1 00:11:41.739: As65 LCP:    ACCM 0x000A0000 (0x0206000A0000)
Mar  1 00:11:41.739: As65 LCP:    MagicNumber 0xE0293F01 (0x0506E0293F01)
Mar  1 00:11:41.739: As65 LCP:    PFC (0x0702)
Mar  1 00:11:41.739: As65 LCP:    ACFC (0x0802)
```

Then, the NAS sends the dial-out router a configuration request asking to authenticate through PAP, as shown in Example 7-24.

Example 7-24 *Debug Output of Incoming Request for PAP*

```
Mar  1 00:11:41.915: As65 LCP: I CONFREQ [REQsent] id 253 len 28
Mar  1 00:11:41.915: As65 LCP:    MRU 1500 (0x010405DC)
Mar  1 00:11:41.915: As65 LCP:    ACCM 0x000A0000 (0x0206000A0000)
Mar  1 00:11:41.919: As65 LCP:    MagicNumber 0x01000000 (0x050601000000)
Mar  1 00:11:41.919: As65 LCP:    PFC (0x0702)
Mar  1 00:11:41.919: As65 LCP:    ACFC (0x0802)
Mar  1 00:11:41.919: As65 LCP:    AuthProto PAP (0x0304C023)
```

The router responds to the configuration request for PAP with a configuration acknowledgment, as shown in Example 7-25.

Example 7-25 *Debug Output of Outgoing Acknowledgment for PAP*

```
Mar  1 00:11:41.919: As65 LCP: O CONFACK [REQsent] id 253 len 28
Mar  1 00:11:41.919: As65 LCP:    MRU 1500 (0x010405DC)
Mar  1 00:11:41.919: As65 LCP:    ACCM 0x000A0000 (0x0206000A0000)
Mar  1 00:11:41.919: As65 LCP:    MagicNumber 0x01000000 (0x050601000000)
Mar  1 00:11:41.919: As65 LCP:    PFC (0x0702)
Mar  1 00:11:41.919: As65 LCP:    ACFC (0x0802)
Mar  1 00:11:41.919: As65 LCP:    AuthProto PAP (0x0304C023)
```

The NAS then responds with a configuration acknowledgment and LCP is complete, as shown in Example 7-26.

Example 7-26 *Debug Output of Incoming Acknowledgment for PAP*

```
Mar  1 00:11:41.923: As65 LCP: I CONFACK [ACKsent] id 12 len 20
Mar  1 00:11:41.923: As65 LCP:    ACCM 0x000A0000 (0x0206000A0000)
Mar  1 00:11:41.927: As65 LCP:    MagicNumber 0xE0293F01 (0x0506E0293F01)
Mar  1 00:11:41.927: As65 LCP:    PFC (0x0702)
Mar  1 00:11:41.927: As65 LCP:    ACFC (0x0802)
Mar  1 00:11:41.927: As65 LCP: State is Open
```

After the LCP phase is completed, the router and NAS start authentication. First, the router sends an outgoing authentication request with the username and password. The following shows the username in the debug output:

```
Mar  1 00:11:41.927: As65 PPP: Phase is AUTHENTICATING, by the peer
Mar  1 00:11:41.927: As65 PAP: O AUTH-REQ id 1 len 37 from "jhuegen"
```

Next, the router receives an incoming authentication acknowledgment, which indicates that authentication is successful. If authentication had failed, a NAK would have been received along with a reason, which is usually authentication failure:

```
Mar  1 00:11:43.883: As65 PAP: I AUTH-ACK id 1 len 5
```

The following output shows that authentication is now complete, and the router moves on to the NCP stage of PPP negotiation:

```
Mar  1 00:11:43.883: As65 PPP: Phase is FORWARDING, Attempting Forward
Mar  1 00:11:43.887: As65 PPP: Phase is ESTABLISHING, Finish LCP
Mar  1 00:11:43.887: As65 PPP: Phase is UP
```

The router now sends an outgoing configuration request stating its intent to use IP address 0.0.0.0. The router expects this request will be refused and a real address will be assigned to it:

```
Mar  1 00:11:43.887: As65 IPCP: O CONFREQ [Closed] id 1 len 10
Mar  1 00:11:43.887: As65 IPCP:    Address 0.0.0.0 (0x030600000000)
```

Next, the dial-out router receives an incoming configuration request to use VJ 15 header compression. Also in this configuration request, the NAS sends its IP address:

```
Mar  1 00:11:43.891: As65 IPCP: I CONFREQ [REQsent] id 254 len 16
Mar  1 00:11:43.895: As65 IPCP:    CompressType VJ 15 slots CompressSlotID
  (0x0206002D0F01)
Mar  1 00:11:43.895: As65 IPCP:    Address 216.192.135.254 (0x0306D8C087FE)
```

The router then rejects the header compression, and by doing so it rejects the IP address of the NAS. This is sent back to the dial-out router another time without the compression request:

```
Mar  1 00:11:43.895: As65 IPCP: O CONFREJ [REQsent] id 254 len 10
Mar  1 00:11:43.895: As65 IPCP:    CompressType VJ 15 slots CompressSlotID
  (0x0206002D0F01)
```

The NAS rejects the requested IP address of 0.0.0.0 and offers 216.192.135.145 to the router for use:

```
Mar  1 00:11:44.007: As65 IPCP: I CONFNAK [REQsent] id 1 len 10
Mar  1 00:11:44.007: As65 IPCP:    Address 216.192.135.145 (0x0306D8C08791)
```

In return, the router requests that the NAS allow it to use the address it offered in the previous configuration request:

```
Mar  1 00:11:44.007: As65 IPCP: O CONFREQ [REQsent] id 2 len 10
Mar  1 00:11:44.007: As65 IPCP:    Address 216.192.135.145 (0x0306D8C08791)
```

The NAS then resends its IP address to the router; this time without requesting header compression:

```
Mar  1 00:11:44.011: As65 IPCP: I CONFREQ [REQsent] id 255 len 10
Mar  1 00:11:44.011: As65 IPCP:    Address 216.192.135.254 (0x0306D8C087FE)
```

The dial-out router replies and acknowledges the IP address of the NAS as follows:

```
Mar  1 00:11:44.015: As65 IPCP: O CONFACK [REQsent] id 255 len 10
Mar  1 00:11:44.015: As65 IPCP:    Address 216.192.135.254 (0x0306D8C087FE)
```

Next, the NAS acknowledges the requested IP address of the dial-out router, which it offered to the router in the first place:

```
Mar  1 00:11:44.123: As65 IPCP: I CONFACK [ACKsent] id 2 len 10
Mar  1 00:11:44.123: As65 IPCP:    Address 216.192.135.145 (0x0306D8C08791)
```

Finally, the NCP state of PPP negotiation is completed, as signaled by the State is Open statement:

```
Mar  1 00:11:44.123: As65 IPCP: State is Open
Mar  1 00:11:44.123: As65 IPCP: Install negotiated IP interface address
  216.192.135.145
```

NOTE The router did not request DNS and WINS server information. It is configured this way because there is no way to automatically provide negotiated information to PCs upon connection. Therefore, this information was obtained ahead of time, and statically configured into the Dynamic Host Configuration Protocol (DHCP) server in the router.

When PPP is up, the router forwards the packets it was holding in the dialer hold-queue during the train-up, authentication, and PPP negotiation phases to their destination. The router also installs a route to the peer on the other end of the connection, as shown in the third line of the output:

```
Mar  1 00:11:44.127: As65 DDR: dialer protocol up
Mar  1 00:11:44.127: As65 DDR: Call connected, 5 packets unqueued, 5 transmitted,
  0 discarded
Mar  1 00:11:44.131: As65 IPCP: Install route to 216.192.135.254
00:11:44: %LINEPROTO-5-UPDOWN: Line protocol on Interface Async65, changed state
  to up
```

The command **show ip interface brief** shows the async interface with its negotiated IP address, and the method by which this IP address was received. Because in this case the IP address is a result of successful PPP negotiation, it shows IPCP as the method. This command and output is demonstrated in Example 7-27.

Example 7-27 *Output of* **show ip interface brief**

```
Router#show ip interface brief
Interface                  IP-Address      OK? Method Status          Protocol
Async65                    216.192.135.145 YES IPCP   up              up
Ethernet0/0                192.168.0.1     YES NVRAM  up              up
```

If PPP does not fully negotiate, it is most likely not caused by a LCP timeout issue on the NAS site. This is because a line speed issue prevents the modem from dialing in the first place. If you receive LCP timeouts as covered in the section, "Troubleshooting Dial-In Service," the service provider is most likely the source of the issue.

Another issue is an authentication problem caused when the username and password combination fails. If this occurs, reconfigure the username and password and, if necessary, verify it with the provider.

The last issue that can cause problems is the router trying to authenticate the NAS. Because the router is the connecting party, and no provider allows a connecting party to authenticate them (unless this is arranged in advance), check the configuration for **ppp authentication** *type*. This command should not be included in the async interface configuration because it will try to authenticate the NAS. If this is the case, your debug should display the output, as shown in Example 7-28.

Example 7-28 *Authentication Type Mismatch*

```
Mar  1 00:09:29.507: As65 LCP: O CONFREQ [ACKsent] id 2 len 24
Mar  1 00:09:29.511: As65 LCP:    ACCM 0x000A0000 (0x0206000A0000)
Mar  1 00:09:29.511: As65 LCP:    AuthProto PAP (0x0304C023)
Mar  1 00:09:29.511: As65 LCP:    MagicNumber 0xE0273903 (0x0506E0273903)
Mar  1 00:09:29.511: As65 LCP:    PFC (0x0702)
Mar  1 00:09:29.511: As65 LCP:    ACFC (0x0802)
Mar  1 00:09:29.827: As65 LCP: I CONFREJ [ACKsent] id 2 len 8
Mar  1 00:09:29.831: As65 LCP:    AuthProto PAP (0x0304C023)
Mar  1 00:09:29.831: As65 LCP: O CONFREQ [ACKsent] id 3 len 24
Mar  1 00:09:29.831: As65 LCP:    ACCM 0x000A0000 (0x0206000A0000)
Mar  1 00:09:29.831: As65 LCP:    AuthProto PAP (0x0304C023)
Mar  1 00:09:29.831: As65 LCP:    MagicNumber 0xE0273903 (0x0506E0273903)
Mar  1 00:09:29.831: As65 LCP:    PFC (0x0702)
Mar  1 00:09:29.831: As65 LCP:    ACFC (0x0802)
Mar  1 00:09:30.083: As65 LCP: I CONFREJ [ACKsent] id 3 len 8
Mar  1 00:09:30.083: As65 LCP:    AuthProto PAP (0x0304C023)
```

In the debug in Example 7-28, the router requests PAP authentication from the NAS. The NAS rejects the request, and because authentication is required, the router requests it again. This process continues in a loop until the router is disconnected for exceeding the connect timer.

AS5x00 Specific Commands and Debugs

High-density Cisco dial routers include the AS5100, AS5200, AS5300, AS5400, and AS5800 series routers. These devices were all developed for an ISP to provide dial-in services for a wide variety of customers. All these devices are modular so that if a modem or a group of modems is faulty, a feature card can be replaced instead of the entire router.

The AS5100 series router is no longer in production, but was solely a router with a lot of external modems on line cards connected to it, all in one chassis; not modular. In fact, phone lines were connected to each of the individual modem line cards on the router. Because the AS5100 is no longer in production and was essentially the same as the router used for the examples on Basic PPP Dial-In Service, it is not covered in detail in this chapter.

The AS5200 series was the first Cisco product in this line to introduce 56 k modems. The router used Microcom modems that were software upgradeable using a digital signal processor (DSP) and firmware files available from the Cisco Systems web site. This server was also the first Cisco dial router to use channelized T1s or PRIs to handle incoming calls. This router has since reached the end of the production cycle.

The AS5300 opened the door to out-of-band signaling. Although the AS5200 had limited out-of-band signaling, the AS5300 (with the MICA modems) allows for polling all kinds of data from the modem that was not previously available. This includes real-time bandwidth measurements, line shape measurements, serial-to-noise ratio, as well as many other

parameters that are key to troubleshooting real-time. Most, if not all competitor dial servers do not provide for this real-time data, and it must be extracted from the modems via the AT command **ati11** after the call is completed. The AS5300 can handle a total of 8 PRIs worth of calls in its 2U chassis.

The AS5400 provides the same out of band signaling, but it has taken port density to a new level. At the time this book was written, the AS5400 can receive a CT3 worth of calls (552 calls if split into 24 PRIs with 23 available channels each) in a router that only consumes 2U worth of rack space. This is possible because of the new NextPort modems included in the AS5400. The AS5350 shares these modems in a 1U chassis that can handle 16 PRIs worth of calls.

The AS5800 was developed with the AS5300 and provided the same functionality as the AS5300, but in a higher density package. It is a large 48VDC unit that was designed for the service provider environment. As well as the use of the MICA modems, all configuration commands are the same as the AS5300.

The AS5850 shares the new NextPort modems with the AS5400 and AS5350, but in a larger chassis developed for sizeable deployments. The AS5850 is a 14U rack mount unit that can handle up to 3360 calls (5xCT3s), 96 T1s or 86 E1s. While not as dense as the AS5400 in terms of ports per unit of rack space, the AS5850 was developed to handle six times its current density.

One of the most commonly used maintenance procedures for all earlier platforms is to upgrade the modem firmware. Although it is relatively straightforward, the procedure requires some steps that are included in the following example.

To upgrade modem software on the routers, you must first know what version of IOS is running on the router. There are two different scenarios for upgrading modem firmware, depending on the IOS version. Each step is covered here.

The first step in upgrading modem firmware is to get the latest version from Cisco.com and put it on your TFTP server. Next, copy it to the flash on the router that requires the update. Be sure not to erase the original contents of the flash memory when copying the upgrade to flash. Example 7-29 shows how to copy an image file to flash.

Example 7-29 *Output from the* **copy tftp flash** *Command*

```
5300-dialin#copy tftp flash
Address or name of remote host []? 192.168.1.17
! Enter the image file name you want to copy:
Source filename []? mica-modem-pw.2.7.3.0.bin
Destination filename [mica-modem-pw.2.7.3.0.bin]?
Accessing tftp://www-emp/ mica-modem-pw.2.7.3.0.bin...
Erase flash: before copying? [confirm]n
Loading mica-modem-pw.2.7.3.0.bin from 192.168.1.17 (via FastEthernet0):
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
    !!!!!!!!!!!!!!
[OK - 474939/949248 bytes]

Verifying checksum...  OK (0xCB52)
```

The image must reside in flash because the modems load this software each time the router is restarted. For IOS versions earlier than 12.0(5), the modem upgrade is done through the **copy flash modem** command, at an enabled prompt, but not in configuration mode. The following is an illustration of that process:

```
5300-dialin#copy flash modem
Modem Numbers (/[-/] | group | all)? all
Name of file to copy? mica-modem-pw.2.7.3.0.bin
Type of service [busyout/reboot] reboot
```

NOTE Using **reboot** as the type of service does not upgrade the modems until the next time the router is reloaded. This is handy if there are prescheduled maintenance windows. You can then use the **reload at** command to schedule the reload for a specific time, in advance.

As a precaution, only use **busyout** if call volume on the router is low. The router upgrades the modems by busying out all modems and upgrading the groups that have no calls on them. These groups contain either six or 12 modems, depending on type. If calls currently connected prevent modems from upgrading, inbound calls can be rejected with a fast-busy signal because no modems are available.

To confirm the upgrade, use the command **show modem version** to verify that the modems have the new version.

For IOS releases later than 12.0(5), the upgrade is done in Software Port Entry (SPE) configuration mode. Use the command **show spe version** to see what firmware versions are stored in flash on the router and determine which ones require an upgrade. The following lines show the configuration commands required to upgrade spe software:

```
5300-dialin#configure terminal
5300-dialin(config)#spe 1/0 2/9
5300-dialin(config-spe)#firmware upgrade reboot
5300-dialin(config-spe)#firmware location flash:mica-modem-pw.2.7.3.0.bin
```

If you want to use **reboot** as the type of upgrade service, specify this before specifying the location of the firmware. After the location of the firmware is specified, the router begins the upgrade. Because the default is **busyout**, you can easily busy out every modem on a busy access server if you do not specify this beforehand.

To confirm the upgrade, use the command **show spe modem** to verify that the modems have the new version.

AS5200 Specific Commands and Debugs

There are numerous additional commands for modems in the AS5200 series. Although this product has reached the end of life (EOL), it is still in widespread use so these commands are still worth covering. Only the most useful and most commonly used commands are covered.

The first of these commands is **show modem**, which shows a list of all modems along with the number of successful and failed incoming call attempts, outgoing call attempts, and a percentage of successful calls. In addition, it displays the current status of every modem: active, busied out, bad, downloading firmware, or pending download.

If you prefer to only see the summary of this information, use the command **show modem summary**. You can almost immediately tell if there is a problem by looking at the success percentage. Most dial-in pools have a 90 percent or higher success rate.

Using the same **show modem** *slot/port* command (including a specific modem) shows you much more detail about the specified modem, along with all the connection speeds reached with that modem.

The **show modem call-stats** command displays the statistics for all modems and includes reasons for disconnect, compression, number of retrains, and a summary of all the information.

Another useful command is **show modem connect-speeds**, which shows the total number of connections at each speed throughout the range for both receive and transmit. A summary along with percentages is also provided later in the section. This is useful to get a quick glimpse of what your clients are experiencing.

A command that you use often when reporting problems to the Technical Assistance Center (TAC) is **show modem csm** *slot/port*, which provides the TAC a view of exactly what is occurring with a call switching module (CSM) at that precise point in time.

Another command that the TAC often uses to help debug problems is **show modem log** *slot/port*. It provides all information about what the modem has done previously. The data is purged on a first in first out basis, so use this show command immediately after the problem occurs. The command shows all RS232 events, modem state events, modem analog signal events, and connection events. It also contains information on train-up speeds, modulation, and serial-to-noise ratio.

The last of the **show** commands created for the AS5200 is **show modem version**, which shows what version of software is on the modems. Always keep this software up to date to obtain the best possible connection for remote users. A separate procedure exists for each kind of modem (see www.cisco.com for further details).

The AS5200s also have several added debugging modes. The first is **debug modem oob**, which shows information about a modem's out-of-band port that polls modem events. The **debug modem csm** command displays information about the CSM that connects calls. It shows all information about calls coming in from the PRI and what modem it lands on.

Perhaps one of the most useful debug commands in dial routers is **debug modem trace**. With this debug, you can select normal, abnormal, or all reasons for call termination. To gather all abnormal call terminations on the entire router, use the command **debug modem trace abnormal**. If you want to gather all terminations from a specific modem, in the event that you think a particular modem is malfunctioning, use the command **debug modem**

trace all *slot/port*. When disconnecting a call that matched the debug, a trace is sent to the screen including everything that took place with the call, down to the finest detail.

Finally, there is one other command that is neither a show command nor a debug command. This command is **test modem back-to-back** *slot/port slot/port*. It performs a test from one modem to the other to ensure that it can train up and pass data. This test is only done at V.34 speeds.

AS5300 Specific Commands and Debugs

The commands for the AS5200 also work with the AS5300. The AS5300 added MICA modems along with some unique out-of-band functionality. This allowed for a few more commands to provide real-time information that was not possible before. Most of these commands provide little information, but assist the TAC in quickly determining a problem.

The first of these new commands is **show modem configuration**, which shows the setup of the modem. Any changes you make to the modem through a modemcap, appears in the output of this command.

Another command for the AS5300 is **show modem mica** *slot/port*. This is another command that the TAC reviews to help determine a problem.

The **show modem operational-status** *slot/port* command assists you in determining some problems, without the help of TAC engineers. The following is sample output from this command with additional comments provided to assist with troubleshooting:

```
5300-dial#show modem operational-status 1/13
Modem(1/13) Operational-Status:

 Parameter #0  Disconnect Reason Info:  (0x0)
       Type (=0 ):  <unknown>
      Class (=0 ):  Other
     Reason (=0 ):  no disconnect has yet occurred
```

The parameter #0 of the output shows a disconnect reason that identifies the type of disconnect, the class of disconnect, and the reason that the disconnect took place. If there is an issue with a user frequently getting disconnected, determine what modem the user was last connected on, and perform this command to obtain information about the disconnection.

Parameters #1 and #2 cover the connection protocol and compression:

```
Parameter #1  Connect Protocol:  LAP-M
Parameter #2  Compression:  V.42bis both
```

Parameter #3 displays the error correction (EC) retransmission count, or the number of times that the modem has gone into error recovery in the transmit direction for a particular

connection. Compare this parameter against the count produced by Parameter #36 (EC packets transmitted, received) to determine if a problem really exists.

```
Parameter #3  EC Retransmission Count:  193
```

Parameter #4 shows the error count received during a back-to-back modem test. During any normal active call, this number is 0:

```
Parameter #4  Self Test Error Count:  0
```

Parameter #5 shows how long the call is connected in seconds. This client is connected for just over 3 hours and 44 minutes:

```
Parameter #5  Call Timer:  13491 secs
```

Parameter #6 shows the number of retrains done by the modems. A high number of retrains identifies that the connection made by the client is not stable. This can happen from a dirty phone line, low signal-to-noise ratio (SNR), or even from someone picking up the phone and setting it back down while a modem call is in place over that line:

```
Parameter #6  Total Retrains:  4
```

Parameter #7 displays the measure of the receive signal quality (SQ) bit error rate for the current modulation, as estimated by the DSP. A value of 0 has the highest number of errors and 7 has the lowest. This value is used in conjunction with some S values configured on the modem to determine when to speed shift or retrain. If the SQ value reported in this parameter drops to the value of S32 (SQ Threshold) for longer than the value of S35 in seconds, the DSP attempts a downward speed shift or retrain. Similarly, if the SQ value goes above the threshold for longer than the value of S34 in seconds, an upward speed shift or retrain occurs.

Parameters #8 and #9 show the connection standard and current connection speed of the modem. This real-time data and might not be the same for the connecting client. The client uses a modem that does not have out-of-band signaling and only knows the initial connection speed:

```
Parameter #7  Sq Value:  3
Parameter #8  Connected Standard:  V.90
Parameter #9  TX,RX Bit Rate:  50666, 28800
```

Parameter #11 displays the transmit symbol rate that transmits samples to the line and the receive symbol rate that receives samples from the line:

```
Parameter #11 TX,RX Symbol Rate:  8000, 3200
```

Parameter #13 displays the transmit carrier frequency in Hertz that the local DCE uses and the receive carrier frequency that the remote DCE uses:

```
Parameter #13 TX,RX Carrier Frequency:  0, 1829
```

Parameter #15 shows trellis coding. Trellis coding adds dependency between symbols to make the detection in noise more robust (forward error correction [FEC]). Trellis coding is displayed in values. The value of 0 correlates to a connection standard V.22, V.22*bis*, V.21,

Bell212, Bell103, V.29, or V.27. The value of 8 correlates to a connection standard of V.32, V.32*bis*, or V.17. The value of 16, 32, or 64 correlates to a connection standard of V.34, V.34+, V.90, or K56Flex:

```
Parameter #15 TX,RX Trellis Coding:  0, 16
```

Parameter #16 shows the preemphasis index, which involves shaping the raw transmit spectrum to deal with spectrum roll-offs. A zero denotes no reshaping. This index is used only with V.34 and V.34+ connection standards:

```
Parameter #16 TX,RX Preemphasis Index:  22, 0
```

Parameter #17 shows if constellation shaping is used. Constellation shaping is a technique for improving noise immunity by using a probability distribution for transmitted signal points. The signal states predict the sensitivity to certain transmission impairments. Constellation shaping is used only with the V.34 and V.34+ connection standards. Values displayed by this parameter are either Off or On:

```
Parameter #17 TX,RX Constellation Shaping:  Off, Off
```

Parameter #18 shows if nonlinear encoding is used. Nonlinear encoding occurs during the training phase and moves the outer points of the constellation away to deal with nonlinear distortion. Nonlinear distortion tends to affect the higher-powered signals. Moving the outer constellation points out reduces the chance of error. Nonlinear encoding is used only with the V.34 and V.34+ connection standards. Values displayed by this parameter are either Off or On:

```
Parameter #18 TX,RX Nonlinear Encoding:  Off, Off
```

Parameter #19 shows if precoding is used. Precoding serves the same purpose as the preemphasis index, but instead manages the bits and not the raw transmit signals. This management is done only when asked for and therefore occurs only in the receive mode. Precoding is used only with the V.34 and V.34+ connection standards. Values displayed by this parameter are either Off or On:

```
Parameter #19 TX,RX Precoding:  Off, Off
```

Parameter #20 shows the transmit level reduction. The transmit level affects the transmit signal by reducing it in a range of 0 to 15 dBm. If nonlinear distortion is detected on either end, the modem detecting this distortion requests a lower-powered transmit signal. Transmit level reduction is used with the V.34 and V.34+ connection standards:

```
Parameter #20 TX,RX Xmit Level Reduction:  0, 0 dBm
```

Parameter #21 shows the SNR. This is the ratio on the server side and should not change. The SNR on the client side is determined by issuing the command **ati11** on the modem after disconnect. The higher the number, the better:

```
Parameter #21 Signal Noise Ratio:  36 dB
```

Parameter #22 shows that the power of the received signal ranges from 0 to -128. The optimum range for the receive level displayed by this parameter is from -12 dBm to -24 dBm:

```
Parameter #22 Receive Level:  -23 dBm
```

Parameter #23 shows the frequency offset, which is a shift in the receive spectrum between the expected carrier frequency and the actual carrier frequency. The typical value is 0 Hz:

```
Parameter #23 Frequency Offset:  0 Hz
```

Parameters #24 and #25 deal with phase jitter. This is found only in analog trunk circuits. Typical frequencies are power generation frequencies and their harmonics (that is, 60 and 120 Hz within the U.S; 50 and 100 Hz international). MICA modems cancel all frequencies of phase jitter:

```
Parameter #24 Phase Jitter Frequency:  0 Hz
Parameter #25 Phase Jitter Level:  0 degrees
```

Parameter #26 displays the far-end echo level (the portion of the transmitted analog signal that has bounced off of the analog front end of the remote modem), which can range from 0 to -90 dBm. A MICA modem cannot handle near-end echo if far-end echo is present and the round-trip delay is greater than ten microseconds. This constraint comes from the number of taps in the echo canceller of MICA modems. The reported far-end echo level must be less than -55 dBm to achieve a V.34+ connection. A greater echo level indicates a digital-to-analog conversion in the path between the MICA modem and the switch:

```
Parameter #26 Far End Echo Level:  -47 dBm
```

A form of this signal/constellation pattern echoes off equipment at the central office and is sent back to the MICA modem. However, the constellation shape might be rotated from its original position. This rotation is called the phase roll. It is shown in degrees of rotation in Parameter #27. The echoed signal consists of a frequency component and a phase component. If the frequency component changes at all, a correction is needed for echo cancellation to work correctly. The typical value is 0 or close to 0:

```
Parameter #27 Phase Roll:  0 degrees
```

Parameter #28 shows round-trip delay, which is the total round-trip from modem to modem in microseconds. This delay is important for proper echo cancellation:

```
Parameter #28 Round Trip Delay:  7 msecs
```

Parameter #30 displays the total count of characters sent and received before any modem compression takes place:

```
Parameter #30 Characters transmitted, received:  11116387, 2693386
```

Parameter #32 is included but not used. Other parameters, such as 10, 12, 14, 34, and 37 do not even show up in the output:

```
Parameter #32 General Portware Information:  22
```

Parameters #33 and #35 show the number of packets transmitted and received by the dial server from the client modem. This is useful in determining whether or not the modems are passing traffic to each other:

```
Parameter #33 PPP/SLIP packets transmitted, received:  24080, 23583
Parameter #35 PPP/SLIP packets received (BAD/ABORTED):  0
```

Parameter #36 displays the number of EC packets transmitted and the number of EC packets received:

```
Parameter #36 EC packets transmitted, received OK:  53258, 48144
```

Parameter #38 shows the moving average of EC packets. One way to determine if the connection has gotten better or worse with speedshifts and retrains is to check the moving average several times over the duration of a call. If the value decreases, the connection has become more stable:

```
Parameter #38 Moving Average of EC packets (Received BAD/ABORTED):  88
```

Parameter #39 shows what robbed bit signaling pattern is used. If robbed bit signaling is detected, the dial server must run a pattern so that the missing bit does not affect data transfer. As long as this is a 0, there is no RBS line between the server and the client:

```
Parameter #39 Robbed Bit Signalling (RBS) pattern:  0
```

Chapter 5, "Dial Technology Background," provided an explanation of the digital pad. Cisco dial servers try to avoid the problems incurred by the use of digital pads on a link. Parameter #40 shows if the dial server detected a digital pad and the compensation status:

```
Parameter #40 Digital Pad:  6.0   dB,  Digital Pad Compensation: Enabled
```

Parameters 41 through 44 deal with V.110 calls. These parameters show frames received bad, frames received good, frames transmitted, and number of times synchronization has been lost. Because this is a V.90 call, the values are always 0:

```
Parameter #41 V110/PIAFS frames received bad:  0
Parameter #42 V110/PIAFS frames received good:  0
Parameter #43 V110/PIAFS frames transmitted:  0
Parameter #44 V110/PIAFS sync lost:  0
```

The last piece of information to specifically review from the **show modem operational-status** output is the line shape. The MICA modems use the V.90 Digital Impairment Learning (DIL) sequence to determine the quality of the line and how much of the line is usable for data. Even if the DIL shows that V.90 is not feasible for the client modem connecting, it does store this information and trains up in V.34 mode.

When reviewing the line shape, you want the vertical line created by the asterisks to be as straight as possible. As the line is more curved at the beginning and the end, the maximum attainable connection speed drops. Example 7-30 shows the line shape output.

Example 7-30 *Line Shape*

```
.........................*
................................*
.................................*
.................................*
................................*
................................*
................................*
................................*
................................*
................................*
................................*
................................*
................................*
................................*
................................*
................................*
................................*
................................*
................................*
................................*
................................*
................................*
...............................*
.............................*
.........................*
```

In this case, the line is mostly straight. Also, if you noticed the speed from parameter 9, the connection speed attained in the example is 50,666 baud, which is almost as good as you can get with a 56 k modem.

The only debugs added for the AS5300s are MICA modem debugs. The command **debug modem mica** displays the status of software downloads to the modems.

AS5400 Specific Commands and Debugs

With the introduction of the AS5400, most of the modem commands changed. The following is a list of the commands from the AS5200 and AS5300 that exist in the AS5400 under a different command:

- **show modem** changed to **show spe modem**.
- **show modem version** changed to **show spe version**.
- **show modem log** changed to **show spe log**.
- **show modem summary** changed to **show spe modem csr summary**.
- **show modem connect-speeds** changed to two separate commands:
  - **show spe modem high** shows high-speed modulation connects.
  - **show spe modem low** shows low-speed modulation connects.

The first of the two newly added show commands is **show spe modem active** *slot/spe*. It is not really a new command, but because modem types changed from MICA to NextPort modems, this command replaces the **show modem operational-status** from the AS5300s. Unfortunately, this command only works when the modem is active. To view the disconnect reason for a specific disconnected modem, refer to the command **show spe log** *slot*/*spe*.

The other helpful command is **show spe modem disconnect-reason** [*slot*/*port* | **summary**]. This command reveals the allocation of disconnect reasons that the particular slot and/or port on that specific modem has encountered. When reviewing the summary, it displays the total number of disconnects for each disconnect reason. This provides a quick indication of how many modem train-up failures took place.

Show commands are not the only thing that changed between the AS5300 and the AS5400. Most of the debug commands changed or are no longer available. The **debug modem csm** changed to **debug csm modem** and the **debug modem trace** command changed to **debug spe modem trace**.

Many specialized debugs added for the AS5400 deal with the NextPort engine and modules. The details of each are more than this book can cover. If you are experiencing a problem where these debugs are required for troubleshooting, the TAC can instruct you on which debugs to run and provide the required output.

Summary

This chapter follows the troubleshooting methodology suggested in Chapter 4, "Troubleshooting Approaches, Models, and Tools," with a layer-by-layer, phase-by-phase approach to remote access dial issues. Some of the topics of the chapter, such as detailed troubleshooting NAS WAN Links and T1s PRIs, are relevant to other parts of the book. This chapter includes an extensive amount of information about troubleshooting dial-in and dial-out services, where the modem's operational status is covered in greatest detail. The specifics of the different NAS platforms provided in this chapter can serve as a practical troubleshooting manual when dealing with all the different platforms of 5x00 NASs.

Review Questions

Answers to the review questions can be found in Appendix A, "Answers to Review Questions."

1. What SQ value shows up in the output of **show modem operational-status**, indicating the most stable connection possible?
2. True or False: Upgraded modem firmware does not have to be stored in flash after it is copied to the modem.

3. What is the controller configuration command required to correctly configure the distance of a circuit that is 639 feet?

```
Router(config-controller)#cablelength short 655
```

4. If you are receiving a blue alarm, what bit pattern is the equipment receiving?
5. What color is an alarm in which the controller status shows "Receiver has loss of Frame"?
6. What happens if one end of a T1 receives a yellow signal?
7. What information is negotiated in the IPCP stage of NCP of a modem dial call?
8. What modem AT command shows the connection information for a call that was previously disconnected?
9. What does BERT stand for?
10. What is the primary advantage that the AS5300 MICA modems have over the Microcom modems in the AS5200 series?
11. What does the call timer show?
12. What command on an AS 5200 and AS5300 shows a summary of all disconnect reasons and the number of disconnects for every reason?
13. What does CSM stand for and what is the command to show the processes of this device?