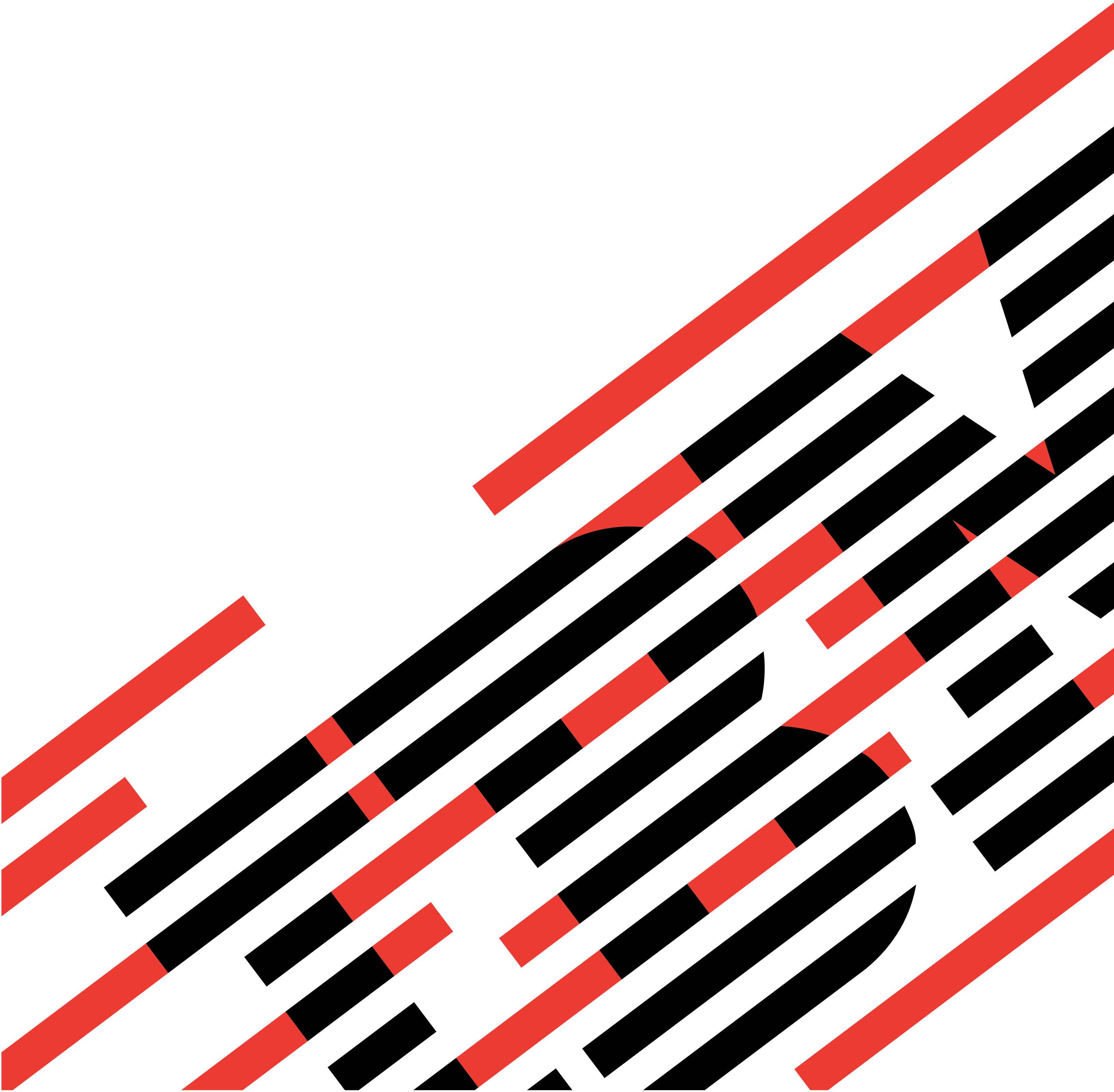
IBM

eserver

iSeries

TCP/IP troubleshooting

Version 5



iSeries

TCP/IP troubleshooting

Version 5

Contents

Chapter 1. TCP/IP troubleshooting 1
What’s new for V5R2? 1
Print this topic 2

Chapter 2. General TCP/IP problems 3
Initial TCP/IP problem analysis 3
Cause list A 3
| IPv6 solutions 5
Cause list B 6
Cause list C 7
Cause list D 8
Cause list E 9
PING command considerations 9
Concatenate the domain name to the host name 10
Common error messages 10
Work with the job log and message queues 11

Chapter 3. Specific application problems 13

| **Chapter 4. Communications trace**. 15
| Plan a communications trace 15
| Perform a communications trace 16
| Start a communications trace 16
| End a communications trace 17
| Dump a communications trace 17
| Print a communications trace 18
| View the contents of a communications trace 18
| Read a communications trace 19
| Additional communications trace function. 21

Chapter 5. TCP/IP configuration files 23

Chapter 6. Product activity log 25

Chapter 1. TCP/IP troubleshooting

What is holding up your TCP/IP functionality? You designed a solid network and followed all the directions, but you have reached an impasse. This topic will lead you to the solution.

This site is a centralized resource for finding answers to TCP/IP problems. You may have a general connectivity problem that is quickly identified or a more localized problem that requires in-depth consideration. Troubleshooting tools are provided below to help you solve the problem.

What's new for V5R2?
Use this topic to learn about new and changed methods for troubleshooting TCP/IP.

Print this topic
Use this topic to print or download a Portable Document Format (PDF) version of the TCP/IP troubleshooting documentation.

General TCP/IP problems
This topic helps you verify your TCP/IP connectivity. Use a question-and-answer format to zero in on your problem and link to potential solutions.

Specific application problems
If you know your problem lies within a particular application, such as FTP or DNS, then use this topic to link to that application for specific solutions.

Communications trace
This topic guides you through the process of collecting a communications trace. A trace can isolate errors and open the door to solving the problem. You can use the trace information yourself or provide it to IBM® specialists as they assist you in troubleshooting.

TCP/IP configuration files
This topic shows you how to copy your TCP/IP configuration files. You will need to provide these copies to IBM, if you decide to consult a specialist for assistance.

Product activity log
Use this topic to find out how the product activity log can assist you in problem analysis.

What's new for V5R2?

New items in the TCP/IP troubleshooting topic for Version 5 Release 2 include:

- **General TCP/IP problems**
  Find ways to troubleshoot problems related to Internet Protocol version 6 (IPv6).
- **Communications trace**
  Find instructions for performing a communications trace using CL commands. This troubleshooting tool traces the data on the communications line, so you can locate the source of your problem.

To find other information about what's new or changed this release, see the Memo to Users .

Print this topic

To view or download the PDF version, select TCP/IP troubleshooting (about 152KB or 26 pages).

To save a PDF on your workstation for viewing or printing:

1. Right-click the PDF in your browser (right-click the link above).
2. Click **Save Target As...**.
3. Navigate to the directory in which you would like to save the PDF.
4. Click **Save**.

Downloading Adobe Acrobat Reader

If you need Adobe Acrobat Reader to view or print these PDFs, you can download a copy from the Adobe Web site (www.adobe.com/prodindex/acrobat/readstep.html) .

Chapter 2. General TCP/IP problems

This topic guides you through several troubleshooting techniques. Use these techniques to isolate general problems and verify TCP/IP connectivity. If you have already verified TCP/IP connectivity, and you know that your problem lies within a particular application, then go to Specific application problems.

Initial TCP/IP problem analysis
This information includes a series of instructions and questions that will help you identify the cause of your problem.

PING command considerations
This information helps you better understand the PING command and make it work for you.

Work with the job log and message queues
This topic provides another option for troubleshooting your TCP/IP.

Initial TCP/IP problem analysis

These questions and answers lead you through problem analysis to help you identify problems and solutions. Link to cause lists as indicated for further troubleshooting.

1. Use the PING command to a host on the local network. Were you successful?
   a. Yes. See item 2.
   b. No. See Cause list A.
2. Use the PING command to the remote system. Were you successful?
   a. Yes. See item 3.
   b. No. See Cause list B.
3. Check the QSYSWRK subsystem for all necessary TCP/IP jobs. Are all jobs there?
   a. Yes. See item 4.
   b. No. See Cause list C.
4. Verify that the interface is active by using NETSTAT. Is interface active?
   a. Yes. See item 5.
   b. No. See Cause list D.
5. Verify the TCP/IP routes are properly configured using TELNET or FTP. Also, see if the connection is established by using NETSTAT. Is the connection there?
   a. Yes. Start Application.
   b. No. See Cause list E.

Cause list A

Be aware that the remote system may have ICMP replies disabled. If ICMP replies are disabled, you will not receive a response from the remote system even though you may have a solid connection. If you suspect that this is the problem, try verifying the connection to other systems and between those other systems to determine where the failure is most likely located.

1. Verify TCP/IP has been activated on your system.

   To ensure that your TCP/IP stack is active:

   a. Enter the `STRTCP` command. If it is active, you should receive message TCP1A04, TCP/IP currently active. If TCP/IP is not active, entering the `STRTCP` command will activate TCP/IP on your server. Verify that no errors occurred while starting TCP/IP.
   b. If you are using IPv6, see IPv6 solutions for troubleshooting techniques related specifically to IPv6. Otherwise, continue with the next item.

2. Verify your server TCP/IP software.

   On the server, the host name LOOPBACK and the interface with a line description value of *LOOPBACK, are reserved for verifying the TCP/IP software. If you specify the LOOPBACK host name, no data is sent out on any of the physical lines. This allows you to quickly determine if TCP/IP software is working correctly on your system.

   To verify your TCP/IP software:

   a. Ensure the local host table has an entry for a LOOPBACK host name and internet address of 127.0.0.1.

   b. Ensure that the interface associated with the LOOPBACK host is active. The internet address usually associated with the LOOPBACK interface is 127.0.0.1. Ensure that there is an interface with the LOOPBACK host name's IP address configured with a line description of *LOOPBACK. Use the command:

      ```
      NETSTAT OPTION(*IFC)
      ```

      to view the LOOPBACK interface's status. If it is not active, use option 9 to activate it.

   c. After verifying that the LOOPBACK host's interface is active, type:

      ```
      PING RMTSYS(LOOPBACK)
      ```

      The loopback host allows the user to:

      - Test FTP, TELNET, LPR or user-written application programs without being attached to a physical line or network.
      - Verify that the TCP/IP software is installed and operating correctly.

      A similar test can be performed by using the PING command to verify connectivity to one of your other locally defined IP addresses.

   d. To test the software and hardware (adapter and network connection), specify the internet address of an external host on your network:

      ```
      PING RMTSYS('nnn.nnn.nnn.nnn')
      ```

   e. If you cannot successfully verify your system's connection to the network by specifying your system name or its internet address, check the source service access point (SSAP) of the line description associated with the interface. X'AA' must be specified as an entry in the SSAP (source service access point) list. This occurs by default when a new line description is created if the SSAP parameter is left at its default value of *SYSGEN. If you have an existing line description, use the Change Line Description command to add these values to the list.

      Not all line description types must have an SSAP for TCP/IP so please check the source service access point (SSAP) list in the line description associated with the interface.

   f. Verify all line description items, particularly the frame size which should be greater than or equal to the maximum transmission unit (MTU) of the interface.

   g. If a remote system fails to respond, it may mean that the system, network, external host, or bridge in the network is unavailable or not working. Failure to respond can also mean that the remote system has ICMP replies disabled. This can occur if the remote system is acting as a firewall and has been configured to not respond to ICMP requests. Try verifying the connection to other systems and between those other systems to determine where the failure is most likely located.

   h. Verify that the local interface configuration is correct.

   i. Ensure the following two routing entries are configured in the QSYSWRK subsystem description if the TCP/IP interfaces, including LOOPBACK, do not activate or you cannot end or start TCP/IP. If they do not exist, or if they are not correct, then add or correct them and try the request again.

      ```
      ADDRTGE  SBSD(QSYS/QSYSWRK) +
               SEQNBR(2505) +
               CMPVAL(TCPIP) +
               PGM(QSYS/QTOCTCPIP) +
               CLS(QSYS/QSYSCLS20) +
               MAXACT(*NOMAX) +
      ```

```
               POOLID(1)

      ADDRTGE  SBSD(QSYS/QSYSWRK) +
               SEQNBR(2506) +
               CMPVAL(TCPEND) +
               PGM(QSYS/QTOCETCT) +
               CLS(QSYS/QSYSCLS20) +
               MAXACT(*NOMAX) +
               POOLID(1)
```

Return to Initial TCP/IP problem analysis to continue troubleshooting.

IPv6 solutions

If you are having problems with IPv6 communications, try these techniques to troubleshoot the network.

1. Verify that the IPv6 stack is running.
   a. Ensure that the loopback interface has been configured and is active. To check the status of the loopback interface, follow these steps:
      1) In iSeries™ Navigator, expand your **server —> Network —> TCP/IP Configuration —> IPv6 —> Interfaces**.
      2) In the right pane, find the loopback interface. The IP address for the IPv6 loopback address is ::1 and the line name is Loopback 6. If the loopback interface does not appear in the list, then you must configure the loopback interface using the **IPv6 Configuration** wizard.
   b. Ping the loopback address (::1). The server sends an IPv6 packet to itself and thereby verifies that the IPv6 stack is working. To test the stack using the ping utility, follow these steps:
      1) In iSeries Navigator, expand your **server —> Network**.
      2) Right-click **TCP/IP Configuration**, click **Utilities**, and click **Ping**.
2. After you verify that the IPv6 stack is running, ensure that your IPv6 line is configured and active. This line may be either an Ethernet line or a configured tunnel line.

   To check the status of lines that are configured on the server, follow these steps:
   a. In iSeries Navigator, expand your **server —> Network —> TCP/IP Configuration —> Lines**.
   b. In the right pane, find the line that should be configured for IPv6 and check the status column. If the line does not appear in the list, then you must configure a line for IPv6 using the **IPv6 Configuration** wizard. See Configure IPv6 for instructions on configuring a line for IPv6. If the line appears in the list and shows a status of **Not loaded**, then the line is configured but is not loaded into the IPv6 stack configuration. Use the Work with line descriptions (WRKLIND) command in the character-based interface to diagnose the problem on the line.
3. Ensure that at least two IPv6 interfaces are active: your local interface and the interface to which you are sending the ping.

   To check the status of the IPv6 interfaces, follow these steps:
   a. In iSeries Navigator, expand your **server —> Network —> TCP/IP Configuration —> IPv6 —> Interfaces**.
   b. In the right pane, find the IP address associated with the local interface and check the status of the interface.
   c. If the interface is **Inactive**, you must activate the interface. To activate the interface, right-click the IP address, and select **Start**.
   d. Repeat these steps to check the status of the remote interface.
4. If your ping to an IPv6 address was unsuccessful, verify the address state of both interfaces. Both interfaces should have an address state of **Preferred**. If either the target or source interface is not in the preferred state, then either choose other interfaces for the test or change the interfaces being used to the proper status and address state.

   To verify or change the address state of the source interface, follow these steps:

a. In iSeries Navigator, expand your **server —> Network —> TCP/IP Configuration—> IPv6 —> Interfaces**.

b. In the right pane, right-click the IP address associated with the interface, select **Properties**, and select the **Options** page. This dialog allows you to specify a preferred lifetime or valid lifetime for the interface.

c. Repeat these steps to check the state of the target interface address.

Cause list B

If your VFYTCPCNN or PING commands were successful to the local system, you should verify the possibility of connecting between your system and the system with which you want to communicate. Run the PING command as you did previously, but this time specify the internet address of the remote host. See Common error messages. Be aware that the remote system, or an intermediate firewall, may have ICMP replies disabled. If ICMP replies are disabled, you will not receive a response from the remote system even though you may have a solid connection. If you suspect that this is the problem, try verifying the connection to other systems and between those other systems to determine where the failure is most likely located.

1. If you can verify the connection using the remote internet address but not the remote system name, then the name or address is not correct in your host table, or the remote name servers may not be available.
2. If your system uses remote name servers, verify that you can reach each remote name server by using the PING command and specifying the internet address of the remote name server.
3. There are additional parameters on the PING command that allow you to specify the packet length, the number of packets to be sent, and the wait time for a response. The default wait time of 1 second allows the remote system enough time to respond in most networks. However, if the remote system is far away or if the network is busy, increasing the wait time parameter can give a successful result.

   It is recommended that the parameter values be left at their default values. Be aware that if you do change them, a combination of large packet length and short wait time may not give the network enough time to transmit and receive the response, and time-outs can occur. If the network is not given enough time to transmit and receive the response, it can appear that you do not have connectivity to a system when, in fact, you actually do.
4. If a remote system fails to respond, it may mean that the system, network, gateway, router, or bridge in the network is unavailable or not working. Failure to respond can also mean that the remote system, or an intermediate firewall, has ICMP replies disabled. Try verifying the connection to other systems and between those other systems to determine where the failure is most likely located.
5. If a remote system fails to respond when you are using the PING command to verify an interface, which is configured to a line description of Ethernet type, make sure the correct Ethernet standard or *ALL is specified in the Ethernet line description.
6. Failure to get responses from all systems in a network indicates the trouble is somewhere along the path. Verify the connection to the gateway leading into the network in question. If this fails, work back from the remote system you cannot reach until you find the point of failure.
7. Packets are sent using a low-level protocol that does not guarantee delivery. Because an echo request may be lost, do not assume that a network or gateway has failed until several commands fail to get beyond a point in the path.

If the PING command to a host on a remote network fails, use the trace route (TRACEROUTE) command to that same network. The trace route utility can perform many of the same connectivity tests that individual ping requests can do, but trace route can do them all in one step. Trace route will test each hop along the path to the remote destination and will indicate whether the problem lies within an intermediate router or within the remote network.

Type `TRACEROUTE RMTSYS('x.x.x.x')`. You may specify the remote system using an IP address or using the remote system name; for example, `('xxxx.xxxx.com')`. The trace route utility accepts both the IPv4 address format `('x.x.x.x')` and the IPv6 address format `('x:x:x:x:x:x:x:x')`.

Trace route is also available through iSeries Navigator. To start the trace route, follow these steps:

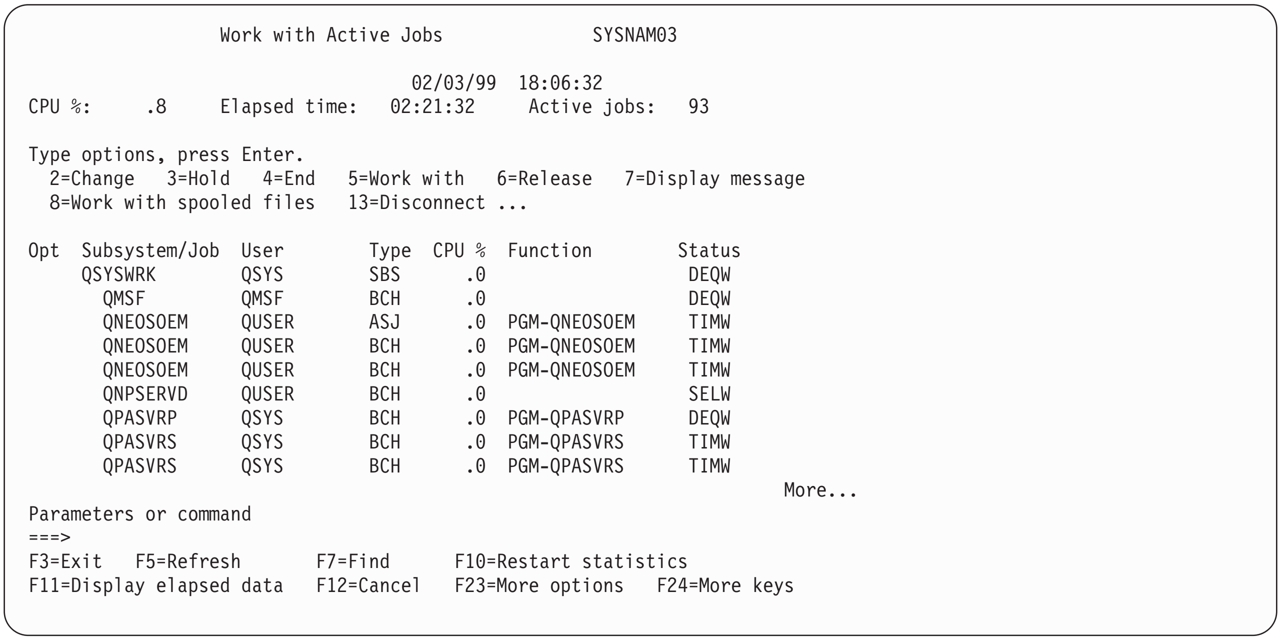
1. In iSeries Navigator, expand your server —> **Network**.
2. Right-click **TCP/IP Configuration**, select **Utilities**, and select **Trace Route**.

Return to Initial TCP/IP problem analysis to continue troubleshooting.

Cause list C

1. Check the server QSYSWRK subsystem for all necessary jobs (local or remote). There should be at least the QTCPIP job. The QTCPIP job controls starting and ending TCP/IP interfaces. There should also be at least one job for each of the applications you are attempting to use as shown in Figure 1 on page 8. It is possible that these jobs may not be named identically to your subsystem jobs for the FTP, LPD, and TELNET jobs. All FTP jobs begin with `QTFTP`. All LPD jobs begin with `QTLPD`. All TELNET jobs will be named `QTVTELNET` and `QTVDEVICE`. It is possible to have more than one FTP, LPD, or TELNET server job. All SMTP jobs begin with `QTSMTP`. SMTP has up to four jobs active in the QSYSWRK subsystem and two jobs active in the QSNADS subsystem. All SNMP jobs begin with `QTMSNMP`. SNMP can have three jobs active in the QSYSWRK subsystem, QTMSNMP, QTMSNMPRCV, and QSNMPSA.

   Use the Work with Active Jobs (WRKACTJOB) command to display these jobs. Type `WRKACTJOB SBS(QSYSWRK)`.
2. End TCP/IP processing using the `ENDTCP OPTION(*IMMED)` command if all the jobs are not there. Look for all the job logs associated with the jobs.
3. Change the job description message logging level for all the job description objects to `4 0 *SECLVL`. See Work with the job log and message queues for detailed information on the message logging levels.
4. Start TCP/IP processing again using the `STRTCP` command
5. Verify that all jobs are active.
6. Check the job logs if the appropriate jobs are not active.



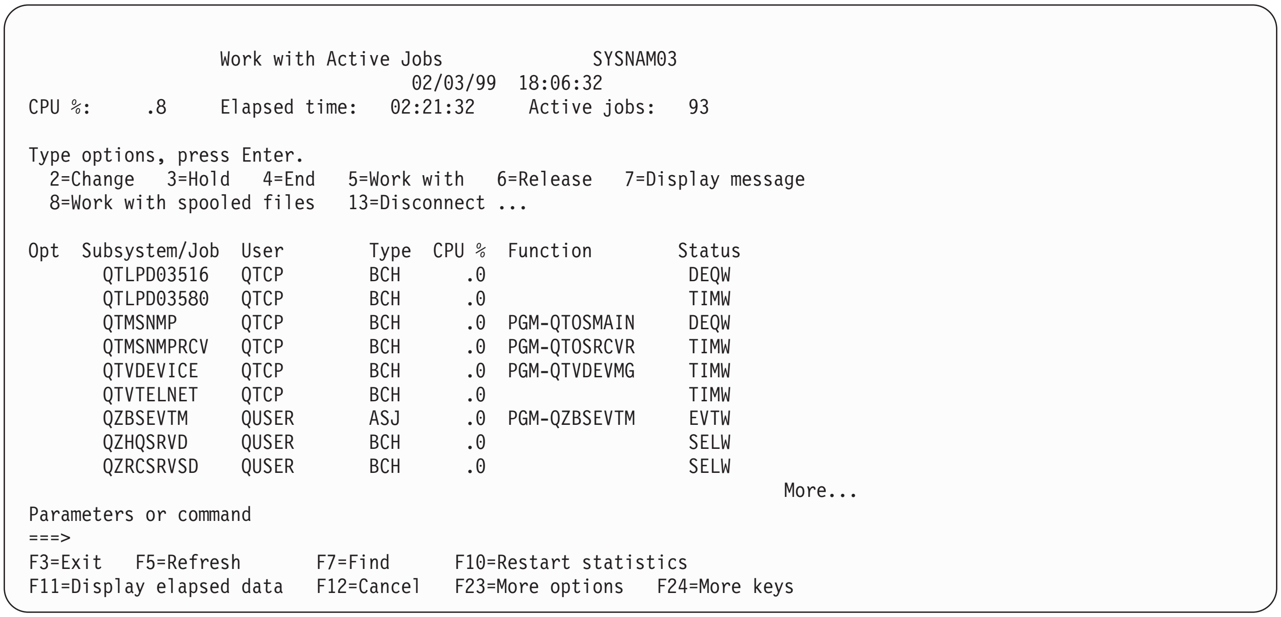
```
                  Work with Active Jobs              SYSNAM03

                                    02/03/99  18:06:32
CPU %:     .8     Elapsed time:   02:21:32     Active jobs:   93

Type options, press Enter.
  2=Change   3=Hold   4=End   5=Work with   6=Release   7=Display message
  8=Work with spooled files   13=Disconnect ...

Opt  Subsystem/Job  User        Type  CPU %  Function        Status
     QSYSWRK        QSYS        SBS      .0                   DEQW
       QMSF         QMSF        BCH      .0                   DEQW
       QNEOSOEM     QUSER       ASJ      .0  PGM-QNEOSOEM     TIMW
       QNEOSOEM     QUSER       BCH      .0  PGM-QNEOSOEM     TIMW
       QNEOSOEM     QUSER       BCH      .0  PGM-QNEOSOEM     TIMW
       QNPSERVD     QUSER       BCH      .0                   SELW
       QPASVRP      QSYS        BCH      .0  PGM-QPASVRP      DEQW
       QPASVRS      QSYS        BCH      .0  PGM-QPASVRS      TIMW
       QPASVRS      QSYS        BCH      .0  PGM-QPASVRS      TIMW
                                                                       More...
Parameters or command
===>
F3=Exit   F5=Refresh       F7=Find      F10=Restart statistics
F11=Display elapsed data   F12=Cancel   F23=More options   F24=More keys
```

Figure 1. Work with Active Jobs Display—Display 1



```
                  Work with Active Jobs              SYSNAM03
                                    02/03/99  18:06:32
CPU %:     .8     Elapsed time:   02:21:32     Active jobs:   93

Type options, press Enter.
  2=Change   3=Hold   4=End   5=Work with   6=Release   7=Display message
  8=Work with spooled files   13=Disconnect ...

Opt  Subsystem/Job  User        Type  CPU %  Function        Status
       QTLPD03516   QTCP        BCH      .0                   DEQW
       QTLPD03580   QTCP        BCH      .0                   TIMW
       QTMSNMP      QTCP        BCH      .0  PGM-QTOSMAIN     DEQW
       QTMSNMPRCV   QTCP        BCH      .0  PGM-QTOSRCVR     TIMW
       QTVDEVICE    QTCP        BCH      .0  PGM-QTVDEVMG     TIMW
       QTVTELNET    QTCP        BCH      .0                   TIMW
       QZBSEVTM     QUSER       ASJ      .0  PGM-QZBSEVTM     EVTW
       QZHQSRVD     QUSER       BCH      .0                   SELW
       QZRCSRVSD    QUSER       BCH      .0                   SELW
                                                                       More...
Parameters or command
===>
F3=Exit   F5=Refresh       F7=Find      F10=Restart statistics
F11=Display elapsed data   F12=Cancel   F23=More options   F24=More keys
```

Figure 2. Work with Active Jobs Display—Display 2

Return to Initial TCP/IP problem analysis to continue troubleshooting.

Cause list D

The network status (NETSTAT) function on the server allows you to view the TCP/IP interface status, the TCP/IP route configuration information, and the TCP/IP connection status on your local system. You can use either the WRKTCPSTS command or the NETSTAT command.

1. Start TCP/IP using the STRTCP command before using the network status function. The Work with TCP/IP Network Status menu is displayed but the options are not functional until TCP/IP has been started.
2. On the Work with TCP/IP interface status display, if you attempt to start an active interface or end an inactive interface, an appropriate error message is sent. If an inactive interface does not reach the active state after taking the start interface option, there may be a problem with the interface, the line, or the line configuration. See the job log of the QTCPIP job in the QSYSWRK subsystem to see what errors might have occurred when activating the interface. You can also look in the QSYSOPR message queue and the history log, QHT (DSPLOG) to help determine the status.
3. Type WRKCFGSTS *LIN to determine if the line description has a problem.
4. Verify that at least one passive listening connection is shown for each of the servers on the Work with TCP/IP Connection Status display, option 3 from the Work with TCP/IP Network Status display. You should verify the connection status for the servers supporting these applications and any other pertinent servers on the network:

   SNMP

   TELNET

   Version 4 Release 4 supports SSL Telnet in addition to Telnet. SSL Telnet reflects a listening port of 992 by default and traditional Telnet uses port 23. Restricting Telnet listening ports is the recommended approach to disabling the traditional Telnet server, while allowing the SSL Telnet to be enabled.

   FTP

   SMTP, if configured

   POP

   LPD

   REXEC

   HTTP, if configured

   Passive listening connections have an asterisk in the *Remote Address* and *Remote Port* fields. Ending these connections is not recommended. Remote systems cannot use SNMP, FTP, or TELNET, send SMTP mail to the local system, or send spooled files using LPR to the local system if the associated passive listening connections have been ended. They can be restarted by ending and starting the servers using the ENDTCPSVR and STRTCPSVR commands and then specifying the server you want ended and started.
5. Ensure that the ports associated with the application you are attempting to use are not restricted. Use option 4 (Work with TCP/IP port restrictions) from the Configure TCP/IP menu to view the current port restrictions.

Return to Initial TCP/IP problem analysis to continue troubleshooting.

Cause list E

Verify the configuration data. If everything checks out, go to Specific application problems and choose the particular application that you are using for further troubleshooting assistance.

PING command considerations

Read the following sections to find out more about the PING command.

Concatenate the domain name to the host name
This section discusses how the server concatenates the domain name to the host name.

Common error messages
This provides examples of some of the most common PING error conditions.

Concatenate the domain name to the host name

This example illustrates how the server uses the local domain name as a search list and concatenates domain names to the host name if a period is not used at the end of the domain name.

Your server name is SYSNAM01.A400SSC.DFW.COMPANY.COM, and you want to verify the connection to a system whose full name is SYSNAM02.DFW.COMPANY.COM. You do not have a SYSNAM02 host name in your local host table.

> If you type `PING SYSNAM02.DFW.COMPANY.COM`, the server sends SYSNAM02.DFW.COMPANY.COM to the remote name server.
>
> If you type `PING SYSNAM02`, the server first sends SYSNAM02.A400SSC.DFW.COMPANY.COM to the remote name server. Then it sends SYSNAM02.DFW.COMPANY.COM. If this wasn't found, it would finally send SYSNAM02.COMPANY.COM. In other words, iSeries TCP/IP concatenates each part of the local domain name to the host name.
>
> If you type `PING SYSNAM02.`, the remote name server reports that the host is unknown. The reason that the remote name server does not recognize SYSNAM02 is because the server sends the SYSNAM02 name to the remote name server without any part of the search list concatenated. The only difference between this name and the previous name is the use of the period at the end of the name.

Common error messages

When you use the PING command to verify the connection to another host in the network, TCP/IP could give you an error message. Use this table to identify common error messages and to determine what you should do to solve the problems.

Error message	What you should do
`No TCP/IP service available`	• TCP/IP has not been started yet or has not completed starting. Use the NETSTAT command to see if TCP/IP is active. • All jobs may not be started in the QSYSWRK subsystem. Use the Work with Active Jobs (WRKACTJOB) command to verify that the QSYSWRK subsystem and related jobs are active. If they are not active, look in the job log or system default output queue for any messages.
`Not able to establish connection with remote host system`	Check your configured interfaces, their related line descriptions and the TCP/IP routes.
`Cannot reach remote system`	TCP/IP could not find a route to the requested destination. Check NETSTAT option 2 and verify that a *DFTROUTE or equivalent network route has been configured and is active.
`Remote host did not respond to VFYTCPCNN within 10 seconds for connection verification 1.`	• Your configuration is probably correct, but you do not get an answer back from the remote system. Ensure that the remote host is able to reach your system. Call the remote system operator and ask them to verify the connection to your system. • Check the host tables or remote name server (if you are using a name server) for both systems, and the TCP/IP interfaces and routes. The remote name server may not be able to serve you for some reason. • If you are using an Ethernet line, make sure you specified the correct Ethernet standard or *ALL.
`VFYTCPCNN: Unknown host, xxxxxx` where xxxxxx is the host name.	The host name could not be resolved to an IP address, either using the host table or a name server. Check the local host table or the remote name servers (if you are using a name server) for the remote host's entry.

Work with the job log and message queues

TCP/IP is shipped with several job descriptions.

The job descriptions are stored in the QSYS or QTCP library. They are generally shipped with a message logging level of 4, a message logging severity of 0, and a message logging text value of *NOLIST. They are shipped with these values to prevent job logs from being created with only job started and job ended messages in them.

If you are having problems with the operation of TCP/IP, one of the first things to do is to change the message logging level on the job description for the application you are having problems with to a message logging text value of *SECLVL. Changing the message logging level generates a job log for that application. You must end, then restart the server for the change to take effect. If you want to change the job immediately, you must use the CHGJOB command to change the message logging level of the active job.

To change the message logging level on the job description for a particular application, see these examples:

- If the problem is with the FTP server, change the QTMFTPS job description by typing this CL command:

  ```
  CHGJOBD  JOBD(QTCP/QTMFTPS)  LOG(4 0 *SECLVL)
  ```

- If the problem is with SMTP, change the QTMSMTPS job description by typing this CL command:

  ```
  CHGJOBD  JOBD(QTCP/QTMSMTPS)  LOG(4 0 *SECLVL)
  ```

  In addition to the QTMSMTPS job description, you might consider changing the logging level of the QSNADS subsystem job description by typing this CL command:

  ```
  CHGJOBD  JOBD(QGPL/QSNADS)  LOG(4 0 *SECLVL)
  ```

Chapter 3. Specific application problems

If you have determined that your problem lies within a specific application that you're running on TCP/IP, then choose the application below for detailed troubleshooting information. Each link takes you out of the general TCP/IP troubleshooting site and into a new site for the application that you choose.

Domain Name System server (DNS)
This topic provides a flow chart for problem analysis and guides you through debugging strategies for DNS problems.

File Transfer Protocol (FTP)
This topic suggests solutions to your FTP problems and demonstrates the server job log as a troubleshooting tool.

Point-to-Point Protocol (PPP)
This topic offers solutions to common PPP connection problems.

Post Office Protocol server (POP)
See this topic for troubleshooting the POP server and other e-mail applications.

Rexec
This topic provides a flow chart to help you zero in on your Rexec problem and find potential solutions.

Simple Mail Transfer Protocol (SMTP)
This topic provides several methods for solving problems with Simple Mail Transfer Protocol (SMTP) and other e-mail applications.

Telnet
This topic assists you with general Telnet problems as well as specific problems related to emulation type and SSL server. In addition, find out what information is necessary for reporting your problem.

Virtual Private Networking (VPN)
This topic guides you through several troubleshooting strategies for VPN problems related to connection, configuration errors, filter rules, and more.

Chapter 4. Communications trace

Use communications trace to troubleshoot TCP/IP. Communications trace is a service function that allows data to be traced on a communications line, such as a local area network (LAN) or a wide area network (WAN). Once the data has been traced, the raw data may be dumped into a stream file or it may be formatted and placed in a spooled file to be displayed or printed.

Communications trace may be used for troubleshooting both IPv4 and IPv6 communications.

Use communications trace in these situations:

- Your problem analysis procedures do not give enough information about the problem.
- You suspect that a protocol violation is the problem.
- You suspect that line noise is the problem.
- You want to know if your application is transmitting information correctly across the network.
- You want to know if you have performance problems with network congestion or data throughput.

To use the CL commands to perform a communications trace, you must have *SERVICE special authority, or be authorized to the Service Trace function of Operating System/400® through iSeries Navigator. See the chapter on user profiles in iSeries Security Reference for more information on this type of authority.

Trace Connection (TRCCNN) is a command for an alternative method of getting a trace that is similar to a communications trace. If you have TCP applications that use SSL or if you use IP Security, the data that flows over the communications line is encrypted; the communications trace may not be helpful if you need to see the data. TRCCNN traces the data before encryption and after decryption and therefore, may be used when the general communications trace is not effective. It provides output similar to the general communications trace output. See TRCCNN (Trace Connection) Command Description in the Application Programming Interfaces (API) topic, for parameters and examples associated with this command.

To use the communications trace function, follow these steps:

Plan a communications trace
The preliminary steps required before you may perform a communications trace.

Perform a communications trace
The steps required to perform the communications trace.

Additional communications trace function
More functions associated with the communications trace.

Plan a communications trace

Before starting to work with a communications trace, follow these steps:

1. If you have not created the library IBMLIB or output queue IBMOUTQ, specify the following commands:

   ```
   CRTLIB LIB(IBMLIB)
   CRTOUTQ OUTQ(IBMLIB/IBMOUTQ)
   ```

2. Specify the following commands to add the library IBMLIB to your library list and to change the output queue for your job to output queue IBMOUTQ:

   ```
   ADDLIBLE IBMLIB
   CHGJOB * OUTQ(IBMLIB/IBMOUTQ)
   ```

3. If the QTCPPRT printer file does not exist on your system, then specify the following commands to create it:

```
CRTPRTF FILE(QTCP/QTCPPRT)  DEV(*JOB)
        RPLUNPRT(*YES) SCHEDULE(*FILEEND)
        FILESEP(0) LVLCHK(*NO)
        TEXT('TCP/IP printer file')
CHGOBJOWN OBJ(QTCP/QTCPPRT) OBJTYPE(*FILE)
        NEWOWN(QSYS)
```

4. Specify the following commands to send the spooled file QTCPPRT containing the communications trace to the output queue IBMOUTQ in library IBMLIB:

```
OVRPRTF FILE(QTCPPRT) OUTQ(IBMLIB/IBMOUTQ)
OVRPRTF FILE(QPCSMPRT) TOFILE(QTCP/QTCPPRT)
```

The printer file overrides are not in effect after your job ends.

5. Obtain the name of the line description associated with the TCP/IP interface with which you are having the problem or which is used by the application or network with which you are having a problem. Use NETSTAT *IFC to determine the name of the line description associated with the interface.
6. Ensure that the line is varied on and that the TCP/IP interface associated with the line has been started so that TCP/IP data can be sent and received over the interface and the line. Use NETSTAT *IFC to verify that the interface is active.

What to do next:
Perform a communications trace

Perform a communications trace

You must use CL commands in the character-based interface to perform a communications trace. Follow these steps to perform a communications trace:

1. Start a communications trace
2. End a communications trace
3. Dump a communications trace
4. Print a communications trace
5. View the contents of a communications trace
6. Read a communications trace

Start a communications trace

This action starts a communications trace for the specified line or network interface description.

Note: A communications trace may no longer be used to trace data on a network server description (*NWS). Use the communication trace function to trace data on either a specific line (*LIN) or a network interface description (*NWI).

To start a communications trace, follow these steps:

1. At the command line, specify `STRCMNTRC`.
2. At **Configuration object**, specify the name of the line, such as `TRNLINE`.
3. At **Type**, specify the type of resource, either `*LIN` or `*NWI`.
4. At **Buffer size**, specify a sufficient amount of storage for the anticipated volume of data. For most protocols, 8 MB is sufficient storage. For a 10/100 Ethernet, 16 MB through 1 GB is sufficient. If you are uncertain, specify `16 MB` for the maximum amount of storage allowed for the protocol.
5. At **Communications trace options**, specify `*RMTIPADR` if you want to limit the data collected to a trace of one remote interface. Otherwise, use the default value.

6. At **Remote IP address**, specify the IP address associated with the remote interface to which the trace data will be collected.

The communications trace continues until one of the following occurs:

- The ENDCMNTRC command is run.
- A physical line problem causes the trace to end.
- The **Trace full** parameter specifies `*STOPTRC` and the buffer becomes full.

What to do next:
End a communications trace

End a communications trace

In order to format and display the trace, you must first end the trace. This action ends the trace but saves the communications trace buffer.

To end a communications trace, follow these steps:

1. At the command line, specify `ENDCMNTRC`.
2. At **Configuration object**, specify the same line you specified when you started the trace, such as `TRNLINE`.
3. At **Type**, specify the type of resource, either `*LIN` or `*NWI`.

What to do next:
Dump a communications trace into a stream file. This is an optional step that may be useful to you. If you prefer to print the raw data without dumping it, go to Print a communications trace

Dump a communications trace

If you are using Internet Protocol version 6 (IPv6), you must dump the trace data into a stream file by following these steps; however, if you are using IPv4, this is an optional part of the communications trace process.

Dumping the data to a stream file offers several advantages. Consider these advantages when deciding whether to use this function:

- You may run new traces without losing data from the existing trace.
- You may format trace data multiple times. For example, if one of your applications is using ASCII, you may first need to format the communications trace in ASCII; if another application is using EBCDIC, you may need to format the same trace data in EBCDIC. Dumping the trace data into a stream file provides the flexibility to format this data twice.
- You may retain trace data while running an initial program load (IPL).
- You may use a custom formatter to generate ouput.

To dump a communications trace, follow these steps:

1. Create a directory, such as `mydir`. See CRTDIR (Create Directory) Command Description in the Control Language (CL) topic, to create a directory.
2. At the command line, specify `DMPCMNTRC`.
3. At **Configuration object**, specify the same line you specified when you started the trace, such as `TRNLINE`.
4. At **Type**, specify the type of resource, either `*LIN` or `*NWI`.
5. At **To stream file**, specify the pathname, such as `/mydir/mytraces/trace1`.

What to do next:
Print a communications trace

Print a communications trace

You may print the communications trace data from two different sources, depending on how you collected the trace. You may print from the raw data you collected, or you may print from a stream file in which you previously dumped the raw data.

Note: To print the communications trace data from a stream file, you must have Java™ (5722JV1) installed on the system.

This action writes the communications trace data for the specified line or network interface description to a spooled file or an output file.

Print from raw data collected:

If you collected the raw data without dumping it, follow these steps to print the data:

1. At the command line, specify `PRTCMNTRC`.
2. At **Configuration object**, specify the same line you specified when you started the trace, such as `TRNLINE`, and press Enter.
3. At **Type**, specify the type of resource, either `*LIN` or `*NWI`.
4. At **Character code**, specify either `*EBCDIC` or `*ASCII`. You should print the data twice, once specifying `*EBCDIC` and then specifying `*ASCII`.
5. At **Format TCP/IP data**, specify `*YES`, and press Enter twice.
6. Perform Steps 1 through 5 again, but specify a different character code.

Print from stream file:

If you dumped the data to a stream file, follow these steps to print the data:

1. At the command line, specify `PRTCMNTRC`.
2. At **From stream file**, specify pathname, such as `/mydir/mytraces/trace1`, and press Enter.
3. At **Character code**, specify `*EBCDIC` or `*ASCII`. You should print the data twice, once specifying `*EBCDIC` and then specifying `*ASCII`.
4. At **Format TCP/IP data**, specify `*YES`, and press Enter twice.
5. Perform Steps 1 through 4 again, but specify a different character code.

What to do next:
View the contents of a communications trace

View the contents of a communications trace

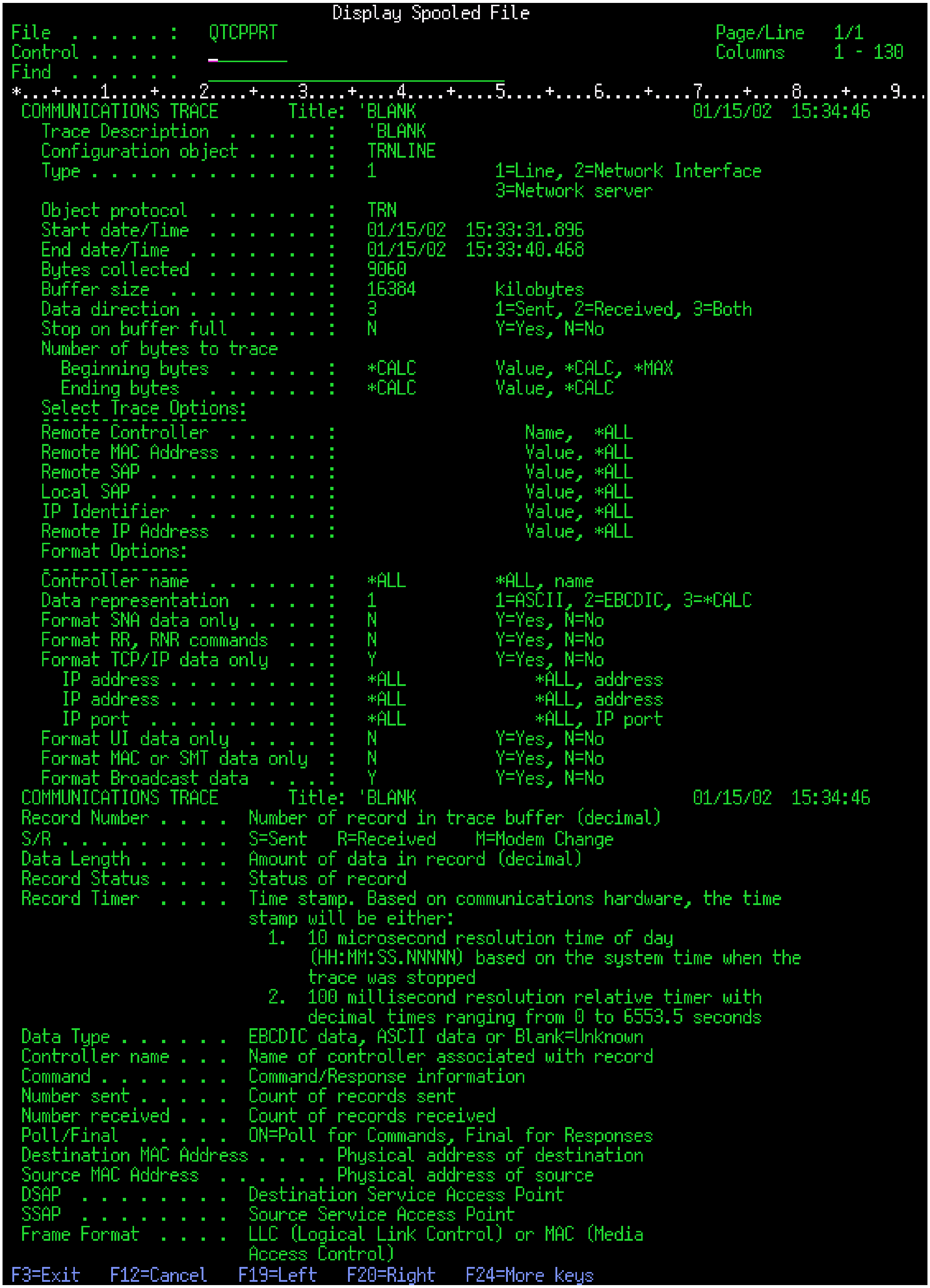
To view the contents of a communications trace, follow these steps:

1. At the command line, specify `WRKOUTQ OUTQ(IBMLIB/IBMOUTQ)`.
2. On the **Work with Output Queue** dialog, press F11 (View 2) to view the date and time of the spooled file with which you want to work. If `More...` appears on the display and you need to continue searching for the spooled file, or page forward or backward through the list of files; otherwise, continue with the next step.
3. Specify `5` in the **Opt** column next to the spooled file you want to display. The last files contain the most recent communications traces.
4. Verify that this is a communications trace for the line traced and that the times that the trace started and ended are correct.

What to do next:
Read a communications trace

Read a communications trace

The communications trace displays several types of information. The first part of the communications trace summarizes the parameters that you specified when you started the trace, such as the name of the **Configuration object**. Page down to find a list of items, such as **Record Number** and **S/R**, with associated definitions; these items represent titles that are later used to identify sections of the communications trace data. It may be useful to refer back to this list as you read the trace data. This image shows the preliminary information in a communications trace.



```
                                Display Spooled File
File  . . . . . :   QTCPPRT                                          Page/Line   1/1
Control . . . . .   ________                                         Columns     1 - 130
Find  . . . . . .   ______________________________
*...+....1....+....2....+....3....+....4....+....5....+....6....+....7....+....8....+....9...
 COMMUNICATIONS TRACE       Title: 'BLANK                                      01/15/02  15:34:46
   Trace Description  . . . . . :   'BLANK
   Configuration object . . . . :   TRNLINE
   Type . . . . . . . . . . . . :   1             1=Line, 2=Network Interface
                                                  3=Network server
   Object protocol  . . . . . . :   TRN
   Start date/Time  . . . . . . :   01/15/02  15:33:31.896
   End date/Time  . . . . . . . :   01/15/02  15:33:40.468
   Bytes collected  . . . . . . :   9060
   Buffer size  . . . . . . . . :   16384         kilobytes
   Data direction . . . . . . . :   3             1=Sent, 2=Received, 3=Both
   Stop on buffer full  . . . . :   N             Y=Yes, N=No
   Number of bytes to trace
     Beginning bytes  . . . . . :   *CALC         Value, *CALC, *MAX
     Ending bytes   . . . . . . :   *CALC         Value, *CALC
   Select Trace Options:
   ---------------------
   Remote Controller  . . . . . :                    Name,  *ALL
   Remote MAC Address . . . . . :                    Value, *ALL
   Remote SAP . . . . . . . . . :                    Value, *ALL
   Local SAP  . . . . . . . . . :                    Value, *ALL
   IP Identifier  . . . . . . . :                    Value, *ALL
   Remote IP Address  . . . . . :                    Value, *ALL
   Format Options:
   ---------------
   Controller name  . . . . . . :   *ALL          *ALL, name
   Data representation  . . . . :   1             1=ASCII, 2=EBCDIC, 3=*CALC
   Format SNA data only . . . . :   N             Y=Yes, N=No
   Format RR, RNR commands  . . :   N             Y=Yes, N=No
   Format TCP/IP data only  . . :   Y             Y=Yes, N=No
     IP address . . . . . . . . :   *ALL              *ALL, address
     IP address . . . . . . . . :   *ALL              *ALL, address
     IP port  . . . . . . . . . :   *ALL              *ALL, IP port
   Format UI data only  . . . . :   N             Y=Yes, N=No
   Format MAC or SMT data only  :   N             Y=Yes, N=No
   Format Broadcast data  . . . :   Y             Y=Yes, N=No
 COMMUNICATIONS TRACE       Title: 'BLANK                                      01/15/02  15:34:46
 Record Number . . . .  Number of record in trace buffer (decimal)
 S/R . . . . . . . . .  S=Sent   R=Received    M=Modem Change
 Data Length . . . . .  Amount of data in record (decimal)
 Record Status . . . .  Status of record
 Record Timer  . . . .  Time stamp. Based on communications hardware, the time
                        stamp will be either:
                          1.  10 microsecond resolution time of day
                              (HH:MM:SS.NNNNN) based on the system time when the
                              trace was stopped
                          2.  100 millisecond resolution relative timer with
                              decimal times ranging from 0 to 6553.5 seconds
 Data Type . . . . . .  EBCDIC data, ASCII data or Blank=Unknown
 Controller name . . .  Name of controller associated with record
 Command . . . . . . .  Command/Response information
 Number sent . . . . .  Count of records sent
 Number received . . .  Count of records received
 Poll/Final  . . . . .  ON=Poll for Commands, Final for Responses
 Destination MAC Address . . . . Physical address of destination
 Source MAC Address  . . . . . . Physical address of source
 DSAP  . . . . . . . .  Destination Service Access Point
 SSAP  . . . . . . . .  Source Service Access Point
 Frame Format  . . . .  LLC (Logical Link Control) or MAC (Media
                        Access Control)
F3=Exit   F12=Cancel   F19=Left   F20=Right   F24=More keys
```

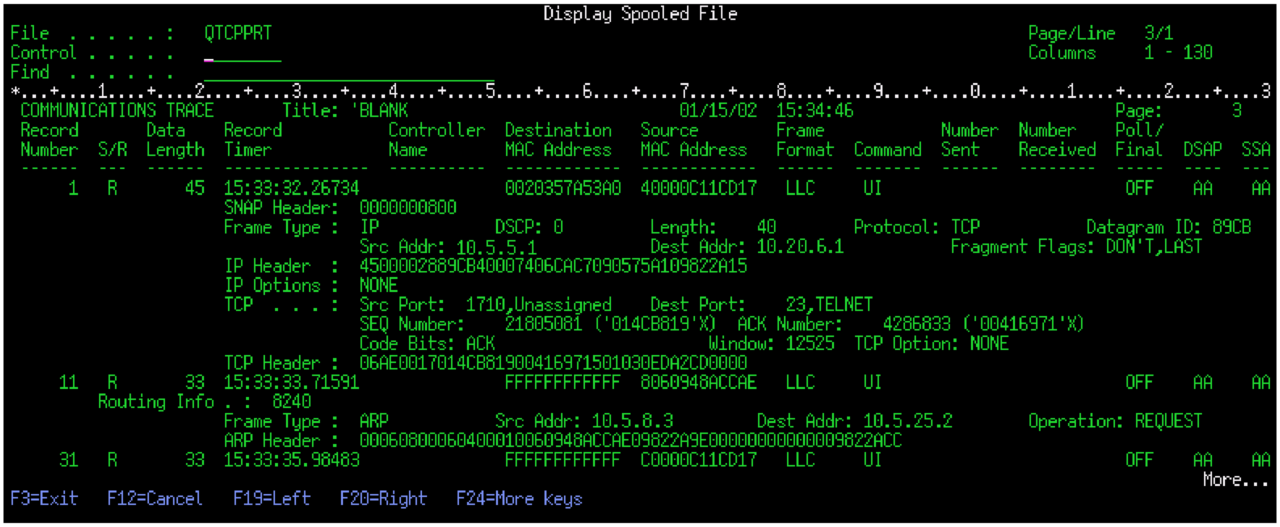
After reading the preliminary information, page down to the actual TCP/IP data in the communications trace. A row of titles, starting with **Record Number**, identifies each section of the data records. Each record number represents a frame, and it includes information such as source and destination IP addresses, length of the complete IP datagram, the type of service (TOS), source and destination ports, and acknowledgment (ACK) numbers. This information should help you debug the problem that you are having with TCP/IP on this iSeries or in the associated network.

If you find an asterisk (*) after a record number, for example, `31*`, be aware that the asterisk represents missing trace data; this occurs when communications trace records are dropped. Communications trace data is collected by the Input-Output Processor (IOP). If the communications line is very busy, the IOP prioritizes all the network traffic and gives a higher priority to the data path input/output than to the communications trace information. Under these circumstances, the IOP may drop some of the communications trace records. This may indicate that the IOP is not capable of handling the excessive speeds or traffic on the network.

If your communications trace is missing data, consider these options:

- Simply acknowledge that your communications line is busy and that frames will be missing from your communications trace.
- Investigate the traffic on the communications line to determine if there is traffic that can be moved to another line or TCP/IP interface.

This image shows the TCP/IP data portion of the communications trace.



```
                                                       Display Spooled File
File  . . . . . :   QTCPPRT                                                                              Page/Line   3/1
Control . . . . .   ________                                                                             Columns     1 - 130
Find  . . . . . .   ______________________________
*...+....1....+....2....+....3....+....4....+....5....+....6....+....7....+....8....+....9....+....0....+....1....+....2....+....3
 COMMUNICATIONS TRACE       Title: 'BLANK                            01/15/02  15:34:46                           Page:       3
 Record       Data    Record           Controller  Destination   Source        Frame            Number  Number    Poll/
 Number  S/R  Length  Timer            Name        MAC Address   MAC Address   Format  Command  Sent    Received  Final  DSAP  SSA
 ------  ---  ------  ---------------  ----------  ------------  ------------  ------  -------  ------  --------  -----  ----  ---
      1   R       45  15:33:32.26734               0020357A53A0  40000C11CD17   LLC     UI                         OFF    AA    AA
                      SNAP Header:  0000000800
                      Frame Type :  IP            DSCP: 0         Length:    40        Protocol: TCP          Datagram ID: 89CB
                                    Src Addr: 10.5.5.1            Dest Addr: 10.20.6.1            Fragment Flags: DON'T,LAST
                      IP Header  :  4500002889CB40007406CAC7090575A109822A15
                      IP Options :  NONE
                      TCP  . . . :  Src Port:  1710,Unassigned    Dest Port:    23,TELNET
                                    SEQ Number:    21805081 ('014CB819'X)  ACK Number:    4286833 ('00416971'X)
                                    Code Bits: ACK                      Window: 12525  TCP Option: NONE
                      TCP Header :  06AE0017014CB81900416971501030EDA2CD0000
     11   R       33  15:33:33.71591               FFFFFFFFFFFF  8060948ACCAE   LLC     UI                         OFF    AA    AA
         Routing Info . :  8240
                      Frame Type :  ARP           Src Addr: 10.5.8.3         Dest Addr: 10.5.25.2       Operation: REQUEST
                      ARP Header :  00060800060400010060948ACCAE09822A9E00000000000009822ACC
     31   R       33  15:33:35.98483               FFFFFFFFFFFF  C0000C11CD17   LLC     UI                         OFF    AA    AA
                                                                                                                         More...
F3=Exit   F12=Cancel   F19=Left   F20=Right   F24=More keys
```

You have completed the communications trace process.

Go to Additional communications trace function to find out how to delete a trace, check the status of a trace, and determine storage space.

Additional communications trace function

These commands and API provide additional communications trace function.

Delete a communications trace

You must delete a communications trace before starting a new trace on the same line. The communications trace can be deleted once the trace has ended. This action deletes the communications trace buffer for the specified line or network interface description.

To delete a communications trace, follow these steps:

1. At the command line, specify `DLTCMNTRC`.
2. At **Configuration object**, specify the name of the line, such as `TRNLINE`.
3. At **Type**, specify the type of resource, either `*LIN` or `*NWI`.

Check a communications trace

You may want to find out if communications traces currently exist on your server. Use Check communications trace (CHKCMNTRC) to return the communications trace status for a specific line or network interface description, or for all of the traces of a specific type that exist on the server. The status is returned to you in a message.

To check the status of a communications trace, follow these steps:

1. At the command line, specify `CHKCMNTRC`.
2. At **Configuration object**, specify the name of the line, such as `TRNLINE`, or specify `*ALL` if you want to check the status of all traces for a specific type.
3. At **Type**, specify type of the resource, either `*LIN` or `*NWI`.

Programmatically check storage space

Use the Check Communications Trace (QSCCHKCT) API to programmatically check the maximum space allocated for traces and the sizes, in bytes, of all traces in active or stopped status on the server. See the Application Programming Interfaces (API) topic for more information on the Check Communications Trace (QSCCHKCT) API.

Chapter 5. TCP/IP configuration files

All reported TCP/IP problems should include a copy of the configuration files used for TCP/IP processing. To obtain a copy of the TCP/IP configuration files, do the following:

1. If you have not created the library IBMLIB or output queue IBMOUTQ, enter the following commands:

   ```
   CRTLIB LIB(IBMLIB)
   CRTOUTQ OUTQ(IBMLIB/IBMOUTQ)
   ```

2. Enter the following commands to add the library IBMLIB to your library list and to change the output queue for your job to output queue IBMOUTQ:

   ```
   ADDLIBLE IBMLIB
   CHGJOB * OUTQ(IBMLIB/IBMOUTQ)
   ```

   Enter the following commands to obtain a list of all physical files used for TCP/IP configuration:

   ```
   WRKF FILE(QUSRSYS/QATOC*) FILEATR(PF)
   WRKF FILE(QUSRSYS/QATM*) FILEATR(PF)
   ```

   To copy the contents of each of the files, you can use option 3 (Copy from the work with files) or you can enter the following command on the command line for each listed file to copy the contents of each file to a separate spooled file in the IBMOUTQ output queue.

   ```
   CPYF FROMFILE(QUSRSYS/QATOCHOST) TOFILE(*PRINT)
        FROMMBR(*ALL)  TOMBR(*FROMMBR)
        MBROPT(*ADD) CRTFILE(*NO) OUTFMT(*HEX)
   ```

Chapter 6. Product activity log

The TCP/IP LIC code creates an entry in the Product Activity Log whenever a TCP/IP datagram is discarded because of a protocol error.

For outbound TCP/IP datagrams, an example of such a protocol error is a failure to establish an X.25 connection over which the datagram was to be sent. In this case, an error is reported to the user and the outbound datagram is discarded.

Inbound datagrams cause an entry in the Product Activity Log to be created when both of these conditions are met:

- The Log Protocol Errors TCP/IP Attribute is set to *YES
- The datagram has failed one of the TCP/IP protocol validity tests specified in RFC 1122, causing the system to discard it. (**Silently discarded** means the following: Discard the received datagram without reporting an error to the originating host device.) Examples of such datagrams are those with checksums or destination addresses that are not valid.

When a datagram is discarded as described above, the IP and TCP/UDP datagrams headers are logged in the detailed data of the Product Activity Log entry. The Reference Code for these Product Activity Log entries is 7004.

Printed in U.S.A.