
# TCP/IP-TOOLS

and

# IPv6/VSE

## Debugging

# Table of Contents

**Preface**

# About this Publication

This is the **TCP/IP-TOOLS and IPv6/VSE Debugging** manual. The manual provides a reference for the problem determination facilities available within TCP/IP-TOOLS and IPv6/VSE.

## Trademarks

The following are lists of the trademark and products referenced in this manual. Symbols for trademarks and registered trademarks do not appear in subsequent references.

### Barnard Software, Inc.

TCP/IP-TOOLS and IPv6/VSE are registered trademarks of Barnard Software, Inc.

### International Business Machines Corporation

IBM, the IBM logo, and ibm.com are trademarks or registered trademarks of International Business Machines Corp., registered in many jurisdictions worldwide. Other product and service names might be trademarks of IBM or other companies. A current list of IBM trademarks is available on the Web at "Copyright and trademark information" at **www.ibm.com/legal/copytrade.shtml**.

Linux is a registered trademark of Linus Torvalds in the United States, other countries, or both.
Microsoft and Windows are trademarks of Microsoft Corporation in the United States, other countries, or both.
UNIX is a registered trademark of The Open Group in the United States and other countries.

## Copyrights

This software and documentation is covered by the following copyright:



## Technical Support

### IBM Customers

IBM IPv6/VSE customers should contact IBM for support.

### BSI Customers

Technical Support is available from Barnard Software, Inc. by phone, mail or email:

Barnard Software, Inc.
806 Silk Oak Terrace
Lake Mary, FL 32746

**Phone:** 1-407-323-4773

**Support:** bsiopti@bsiopti.com

**Sales:** bsisales@bsiopti.com

Support is available from 9:00 a.m. through 5:00 p.m. EST, Monday through Friday.

If a TSR (Technical Support Representative) is not available at the time of your call, please leave a message and a TSR will return your call as soon as possible. Please provide the following information: name, company, phone number, product name, product release level, and a short description of the problem.

## BSIUsers Announcement List Server

When new releases of TCP/IP-TOOLS are available BSI will post an announcement on its BSIUsers announcement list.

To subscribe to the BSIUsers announcement list send an email to this email address

BSIUsers-subscribe@yahoogroups.com

To unsubscribe to the BSIUsers announcement list send an email to this email address

BSIUsers-unsubscribe@yahoogroups.com

### Problem Determination

If you have a problem using a TCP/IP-TOOLS application always check the SYSLST output for additional information and messages. Most messages are written to SYSLST and not to the VSE/ESA or z/VSE system console.

When contacting BSI for technical support always have the applications JCL/commands, console and SYSLST output available for problem determination. The SYSLST output is very important.

While a TCP/IP-TOOLS application is running, you can issue the **AR CANCEL XX,PARTDUMP** command to terminate TCP/IP-TOOLS application and dump the partition to SYSLST. Using the VSE/POWER Flush (F) command cancels the TCP/IP-TOOLS application partition without a dump.

If the TCP/IP-TOOLS application partition stops responding to its console interface, use the **AR DUMP XX** command to obtain a dump of the partition.

**Chapter 1**

# Stack Tracing

The BSTTINET and BSTT6NET TCP/IP stacks have tracing facilities available.

In general, stack tracing is not intended for use by customers and end users. BSI or IBM support personal may ask for tracing to assist in analyzing a specific problem.

# Problem Determination

During problem determination always review the applications JCL/commands and full SYSLST log output. While error messages are written to the z/VSE console, these message only indicate more information is available in the SYSLST log output.

When contacting BSI or IBM for support, please have the applications JCL/commands, full SYSLST log output and console log available. In addition, it can be useful to have the BSTTINET/BSTT6NET (TCP/IP stack) JCL/commands and full SYSLST log output available.

If tracing has been requested by BSI or IBM support personal, output will be sent to the application's SYSLST.

Note: In this manual the z/VSE MSG commands shown reference a jobname of BSTTINET (referring to the BSI IPv4 TCP/IP stack partition jobname) or BSTT6NET (referring to the BSI IPv6 TCP/IP stack partition jobname). When using the commands shown in this manual change BSTTINET/BSTT6NET to the correct jobname value for your system before using the command.

# Tracing Control Commands

### TRACEIP

```
TRACEIP ip-address subnet-mask

E.g.,
MSG BSTTINET,D=TRACEIP 192.168.1.16 255.255.255.255
MSG BSTT6NET,D=TRACEIP fd00:806:1::1 /128

To reset the TRACEIP specification ...
MSG BSTTINET,D=TRACEIP 0.0.0.0 0.0.0.0
MSG BSTT6NET,D=TRACEIP ::0 ::0
```

The TRACEIP command is used to restrict BSTTINET/BSTT6NET output generated by the TRACE2 command to a specific host or subnet. Only traces related to the specific host or subnet are output to SYSLST.

### TRACEID

```
TRACEID id

E.g.,
MSG BSTTINET,D=TRACEID F2
MSG BSTT6NET,D=TRACEID S5

To reset the TRACEID specification ...
MSG BSTTINET,D=TRACEID *
MSG BSTT6NET,D=TRACEID *
```

The TRACEID command is used to restrict BSTTINET/BSTT6NET output generated by the TRACE command to a specific partition. Only traces related to the specific partition are output to SYSLST.

### IP TRACE

The IP TRACE command enables and disabled the stack's internal trace facility.

```
IP TRACE ON|OFF

E.g.,
MSG BSTTINET,D=IP TRACE ON
MSG BSTT6NET,D=IP TRACE OFF
```

The IP TRACE ON|OFF command is used to enable (ON) or disable (OFF) BSTTINET/BSTT6NET internal trace output. Internal trace output is intended for use by BSI support personal. Output can change as traces are added or refined. In general, the timestamp, thread name, thread number, thread trace id and trace specific data are shown in each line.

**Sample output**

```
13:51:53.138F5EC2 devtask      13 OSAX     RSC=E2000001 00000056 00000000 00000000 DUR=   681
13:51:53.139733C2 devtask      13 OSAX     RSC=F1000004 00000056 00000000 00000000 DUR=   126
13:51:53.139DA0C2 devtask      13 OSAX     RSC=E7000000 00000056 00000000 00000000 DUR=   103
13:51:53.13B4A7C2 devtask      13 OSAX     RSC=F2000000 00000056 00000000 00000000 DUR=   368
13:51:53.143C9082 devtask      13 OSAX     RSC=D9000000 00000056 00000000 00000000 DUR=  2175
13:51:53.144AB6C2 devtask      13 OSAX     RSC=F3002800 00000056 00000000 00000000 DUR=   226
13:51:53.1452E682 devtask      13 getbuf   RSC=00000003 FF000000 00000098 00000000 DUR=   131
13:51:53.14576FC2 devtask      13 getbuft  RSC=00000180 FF000000 00000098 00000000 DUR=    72
13:51:53.145B7082 devtask      13 getbufm  RSC=00000016 FF000000 00000098 00000000 DUR=    65
13:51:53.145F8682 devtask      13 getbufc  RSC=00000180 FF000000 00000098 00000000 DUR=    65
13:51:53.14641AC2 devtask      13 getbufa  RSC=03480F94 000005F4 00000098 00000000 DUR=    73
13:51:53.14681302 devtask      13 getbufx  RSC=0347D3E0 000005F4 00000098 00000000 DUR=    64
13:51:53.146C43C2 devtask      13 ni_in    RSC=03480F94 00000000 00000060 00000080 DUR=    67
13:51:53.149577C2 devtask      13 ip_in    RSC=03480F94 007B0223 03480FC0 00000000 DUR=   659
13:51:53.149B2942 devtask      13 localout RSC=03480F94 00000000 00000000 00000000 DUR=    91
13:51:53.149DD202 devtask      13 ipreass  RSC=03480F94 00000000 00000090 00000080 DUR=    43
13:51:53.14A06DC2 devtask      13 signal   RSC=00003FF7 FF000000 00000080 00000060 DUR=    41
13:51:53.14A2FA42 devtask      13 ipdstopt RSC=03480F94 00003FF7 03480F94 00000000 DUR=    41
13:51:53.14A59102 devtask      13 tcp_in   RSC=008D5A98 00000000 00000078 00000080 DUR=    42
13:51:53.14A88AC2 devtask      13 psend    RSC=00003FFF 00000000 00000080 00000080 DUR=    47
13:51:53.14AAF802 devtask      13 psend    RSC=03480F94 00000000 00000080 00000080 DUR=    39
13:51:53.14ADBA02 devtask      13 signal   RSC=00003FEE 007F0398 031F8760 C0A801E2 DUR=    44
13:51:53.14B09682 devtask      13 READY    RSC=00000006 00000234 00000000 00000060 DUR=    46
13:51:53.14BEA902 devtask      13 OSAX     RSC=F3000004 00000040 00000000 00000000 DUR=   225
13:51:53.14C1EE82 tcpinp        6 signal   RSC=00003FEF 00A798A8 00000080 008E6410 DUR=    52
13:51:53.14C3F0C2 tcpinp        6 preceiv2 RSC=03480F94 00000000 00000088 00000080 DUR=    33
13:51:53.14C5B9C2 tcpinp        6 tcpinp   RSC=03480F94 00000000 00000000 00000000 DUR=    28
13:51:53.14DDF3C2 tcpinp        6 CKSUM    RSC=03480FAC 00000000 00000000 C3D2E2E4 DUR=   388
13:51:53.14E1AC82 tcpinp        6 TCPDEMUX RSC=C0A801E3 00000000 00000000 00000060 DUR=    59
13:51:53.14E3B582 tcpinp        6 TCPDEMUX RSC=00000005 00000000 00000000 00000060 DUR=    33
```

The fields shown are

Time-stamp
Thread name
Thread number
Thread trace ID
Resource data
Duration since last trace (micro-seconds)

**IP TRACE2**

The IP TRACE2 command is used to enable and disable the stack's packet trace facility.

```
IP TRACE2 ON|OFF

E.g.,
MSG BSTTINET,D=IP TRACE2 ON
MSG BSTT6NET,D=IP TRACE2 OFF
```

The IP TRACE2 ON|OFF command is used to enable (ON) or disable (OFF) BSTTINET/BSTT6NET packet trace output. Packet trace output is intended for use by BSI support personal.

The BSTTINET/BSTT6NET packet trace output is useful for determining the flow of TCP socket activity.

***IPv4 Sample output***

```
10:07:10.13D9CB80 Out      40 FC00 10 00000C4C 0002C6F9 C0A801E2 00AF 1001 00000000000000000000000000000000
10:07:10.149F7542 In       40 E000 10 0002C6F9 00000C4D C0A801E3 1001 00AF 00000000000000000000000000000000
10:07:20.DC41C700 In       62 E000 18 0002C6F9 00000C4D C0A801E3 1001 00AF 0000000000000000061002838F
10:07:20.DC7D4000 Out      40 FC00 10 00000C4D 0002C70F C0A801E2 00AF 1001 00000000000000000000000000000000
10:08:11.654D0AC2 Out      40 FC00 10 00000C4C 0002C70F C0A801E2 00AF 1001 00000000000000000000000000000000
10:08:11.661D5881 In       40 E000 10 0002C70F 00000C4D C0A801E3 1001 00AF 00000000000000000000000000000000
10:09:12.BC8B8B02 Out      40 FC00 10 00000C4C 0002C70F C0A801E2 00AF 1001 00000000000000000000000000000000
10:09:12.BD46F300 In       40 E000 10 0002C70F 00000C4D C0A801E3 1001 00AF 00000000000000000000000000000000
```

The fields shown are

Time-stamp

In/Out Inbound/Outbound packet indicator

Packet length (decimal/IPv4 or hexadecimal/IPv6)

TCP Window Size (hexadecimal)

TCP Flags (hexadecimal)

TCP Sequence number (hexadecimal)

TCP ACK number (hexadecimal)

Destination IP address (hexadecimal)

Destination Port number (hexadecimal)

Source Port number (hexadecimal)

Packet Data (16 bytes, hexadecimal)

### *IPv6 Sample Output*

```
16:39:39.62E09EC2 Out 0004 FE00 02 00998A83 00000000 FD000806000100000000000000000002 1006 0015 02041.
16:39:39.633751C1 In  0004 3F88 12 590326CC 00998A84 FD000806000100000000000000000003 0015 1006 02041.
16:39:39.63A458C1 Out 0000 FE00 10 00998A84 590326CD FD000806000100000000000000000002 1006 0015 00000.
16:39:39.7B68DE02 In  0083 3F88 18 590326CD 00998A84 FD000806000100000000000000000003 0015 1006 32323.
16:39:39.7C59FCC2 Out 0000 FE00 10 00998A84 59032750 FD000806000100000000000000000002 1006 0015 00000.
16:39:39.7CC8FB82 Out 000A FE00 18 00998A84 59032750 FD000806000100000000000000000002 1006 0015 55534.
16:39:39.7D1F3082 In  0000 3F88 10 59032750 00998A8E FD000806000100000000000000000003 0015 1006 00000.
16:39:39.7D614682 In  0024 3F88 18 59032750 00998A8E FD000806000100000000000000000003 0015 1006 33333.
16:39:39.7DD999C2 Out 000A FE00 18 00998A8E 59032774 FD000806000100000000000000000002 1006 0015 50415.
16:39:39.86FE7B82 In  0000 3F88 10 59032774 00998A98 FD000806000100000000000000000003 0015 1006 00000.
16:39:39.B0A11041 In  006D 3F88 18 59032774 00998A98 FD000806000100000000000000000003 0015 1006 32333.
16:39:39.B14BBC81 Out 0006 FE00 18 00998A98 590327E1 FD000806000100000000000000000002 1006 0015 51554.
16:39:39.B19C0001 In  0000 3F88 10 590327E1 00998A9E FD000806000100000000000000000003 0015 1006 00000.
16:39:39.B29952C1 In  0043 3F88 18 590327E1 00998A9E FD000806000100000000000000000003 0015 1006 32323.
16:39:39.B32AFBC1 Out 0000 FE00 10 00998A9E 59032824 FD000806000100000000000000000002 1006 0015 00000.
16:39:40.F39820C2 In  0000 3F88 11 59032824 00998A9E FD000806000100000000000000000003 0015 1006 00000.
16:39:40.F6404880 In  0043 3F88 19 590327E1 00998A9E FD000806000100000000000000000003 0015 1006 32323.
16:39:40.0B02AF40 Out 0000 FE00 10 00998A9E 59032825 FD000806000100000000000000000002 1006 0015 00000.
```

The fields shown are

Time-stamp

In/Out Inbound/Outbound packet indicator

Payload length (hexadecimal)

TCP Window Size (hexadecimal)

TCP Flags (hexadecimal)

TCP Sequence number (hexadecimal)

TCP ACK number (hexadecimal)

Destination IP address (hexadecimal)

Destination Port number (hexadecimal)

Source Port number (hexadecimal)

Packet Data (16 bytes, hexadecimal)

### Combining Traces

The IP TRACE command can be used at the same time as the IP TRACE2 command. When used together they allow you to get a packet trace and an internal trace at the same time.

```
MSG BSTTINET,D=TRACEIP 192.168.1.1 255.255.255.255
MSG BSTTINET,D=TRACEID S1
MSG BSTTINET,D=IP TRACE ON
MSG BSTTINET,D=IP TRACE2 ON

... create problem

MSG BSTTINET,D=IP TRACE2 OFF
MSG BSTTINET,D=IP TRACE OFF
MSG BSTTINET,D=SEGMENT * $$ LST CLASS=...
```

In the above example, the stack will output internal traces for the partition S1 and packet traces for the remote host 192.168.1.1. These stack trace can be combined with application traces to provide a complete picture of what is happening within the application and the TCP/IP stack.

**Chapter 2**

# Capture Facility

The BSTTINET and BSTT6NET TCP/IP stacks have an external packet capture facility available. This facility was introduced in Build 249. IBM IPv6/VSE customers need IBM IPv6/VSE 1.1 with APAR DY????? applied.

The packet capture facility allows customers or support personal to capture inbound and outbound packets, writing them to a VSAM ESDS file. Once the capture has been turned off, the VSAM ESDS file can be transferred to a PC and viewed using the Ethereal/Wireshark open source packet sniffer.

### Using the Capture Facility

The CAPTURE command is used to enable and disable the external packet capture facility. When the packet capture facility is in use, packets are written to the BSTTCAP VSAM ESDS file. The DLBL for the BSTTCAP file can be added to System Standard Labels or to the JCL used to run the stack. When the CAPTURE ON command is used, the BSTTCAP file is automatically opened when the stack processes the next packet. When the CAPTURE OFF command is used, the BSTTCAP file is automatically closed after the next packet is processed by the stack. The BSTTCAP file is reset each time it is opened.

After trace data has be captured in the BSTTCAP file, you can FTP the file to a PC in BINARY mode where the data can be viewed using Ethereal/Wireshark.

### CAPTURE

```
CAPTURE ON|OFF


E.g.,
MSG BSTTINET,D=CAPTURE ON
MSG BSTT6NET,D=CAPTURE OFF
```

The CAPTURE ON|OFF command is used to enable (ON) or disable (OFF) BSTTINET/BSTT6NET packet capture facility.

### Allocating the BSTTCAP File

The following is a sample job stream for allocating the BSTTCAP file

```
// DLBL IJSYSUC,'VSESP.USER.CATALOG',,VSAM
// EXEC IDCAMS,SIZE=IDCAMS
   DELETE IPV6.VSE.PACKET.CAPTURE.FILE CLUSTER PURGE
   DEFINE CLUSTER -
   (NAME(IPV6.VSE.PACKET.CAPTURE.FILE) -
    SPEED -
    REUSE -
    SHR(2 3) -
    NONINDEXED -
    RECSZ(1500 18425) -
    VOL(SYSWK1))-
   DATA(NAME(IPV6.VSE.PACKET.CAPTURE.FILE.DATA) CISZ(18432) CYL(10 0))
   LISTC ENT(IPV6.VSE.PACKET.CAPTURE.FILE) ALL
/*
```

The highlighted fields in the above sample JCL may be changed. The VSAM ESDS cluster name, volume and number of cylinders to allocate are user modifiable. The CISZ should not be changed. Each cylinder allocated will hold slightly less than 1MB of trace data. Therefore, a cylinder allocation of 100 cylinders will hold a little less than 100MB of trace data.

### Add BSTTCAP to Standard Labels

This sample JCL will add a label to System Standard labels. You should also add the label for the BSTTCAP file to your normal system IPL procedure.

```
// JOB STDLABEL
// OPTION STDLABEL=DEL
BSTTCAP
/*
/&
// JOB STDLABEL
// OPTION STDLABEL=ADD
// DLBL BSTTCAP,'IPV6.VSE.PACKET.CAPTURE.FILE',,VSAM,CAT=VSESPUC
/*
/&
```

### Capturing Data

The CAPTURE command is used to enable (ON) or disable (OFF) the capture of data.

To enable the external packet capture facility ...

```
MSG BSTT6NET,D=CAPTURE ON
```

The BSTTCAP file will automatically open when the next packet is processed by the stack.

To disable the external packet capture facility ...

```
MSG BSTT6NET,D=CAPTURE OFF
```

The BSTTCAP file will automatically close after the next packet is processed by the stack.

**Transferring BSTTCAP Data for Viewing**

The following sample BSTTFTPC JCL can be used to transfer the BSTTCAP file to a remote host for viewing by the Ethereal/Wireshark packet viewer.

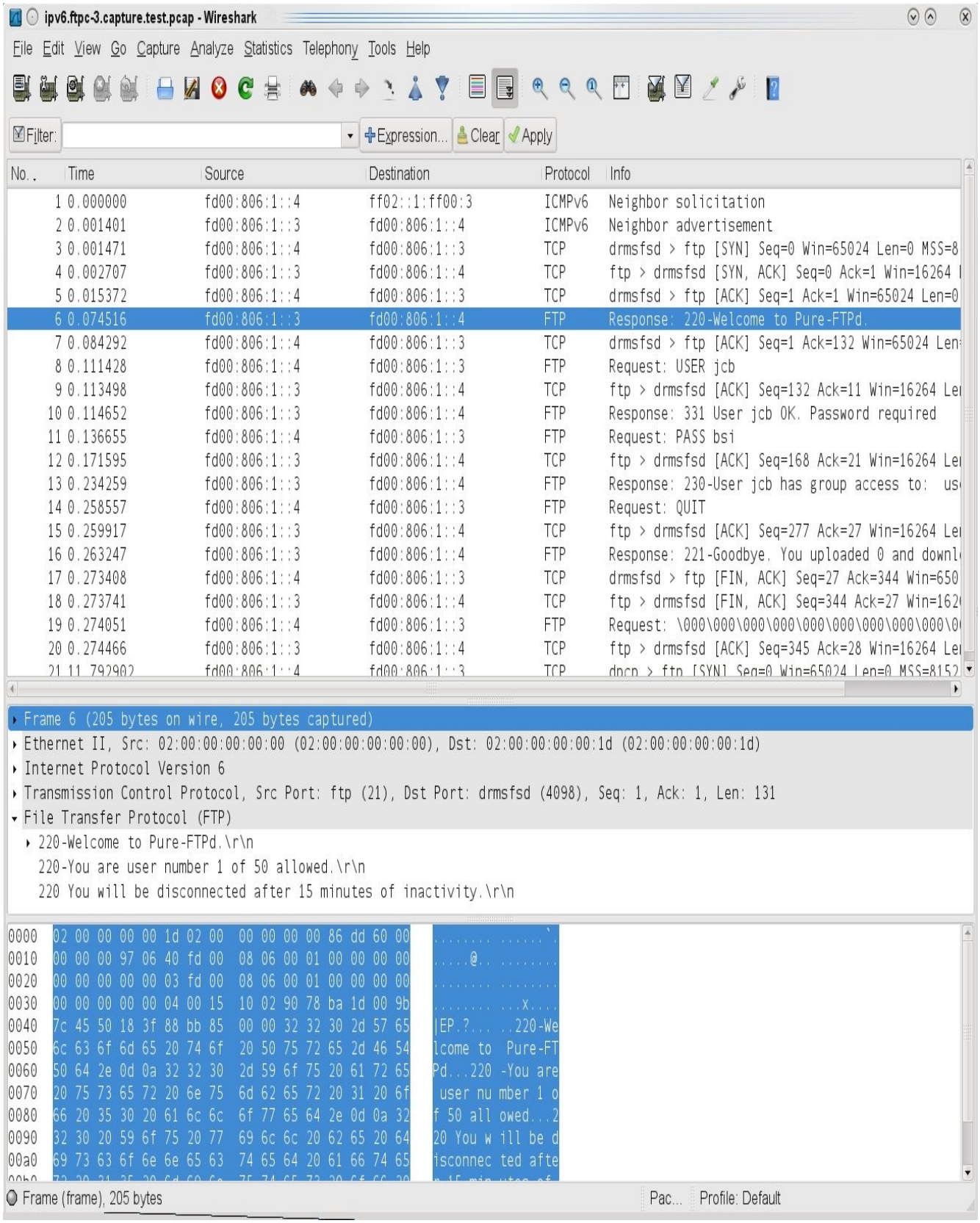
```
// EXEC BSTTFTPC,SIZE=BSTTFTPC
ID 66
OPEN ::FFFF:192.168.1.60
USER userid
PASS password
*
CWD directory
INPUT VSAM BSTTCAP
TYPE I
PASV
STOR file.name.pcap
*
QUIT
/*
```

The highlighted fields in the above sample JCL can be modified by you for your installation.

You can now use Wireshark/Ethereal to open the packet capture file.

The following screen shot is an example of how the capture trace looks in wireshark.

Debugging


ipv6.ftpc-3.capture.test.pcap - Wireshark
File Edit View Go Capture Analyze Statistics Telephony Tools Help
Filter:
Expression... Clear Apply
No. Time Source Destination Protocol Info
1 0.000000 fd00:806:1::4 ff02::1:ff00:3 ICMPv6 Neighbor solicitation
2 0.001401 fd00:806:1::3 fd00:806:1::4 ICMPv6 Neighbor advertisement
3 0.001471 fd00:806:1::4 fd00:806:1::3 TCP drmsfsd > ftp [SYN] Seq=0 Win=65024 Len=0 MSS=8
4 0.002707 fd00:806:1::3 fd00:806:1::4 TCP ftp > drmsfsd [SYN, ACK] Seq=0 Ack=1 Win=16264
5 0.015372 fd00:806:1::4 fd00:806:1::3 TCP drmsfsd > ftp [ACK] Seq=1 Ack=1 Win=65024 Len=0
6 0.074516 fd00:806:1::3 fd00:806:1::4 FTP Response: 220-Welcome to Pure-FTPd.
7 0.084292 fd00:806:1::4 fd00:806:1::3 TCP drmsfsd > ftp [ACK] Seq=1 Ack=132 Win=65024 Len
8 0.111428 fd00:806:1::4 fd00:806:1::3 FTP Request: USER jcb
9 0.113498 fd00:806:1::3 fd00:806:1::4 TCP ftp > drmsfsd [ACK] Seq=132 Ack=11 Win=16264 Le
10 0.114652 fd00:806:1::3 fd00:806:1::4 FTP Response: 331 User jcb OK. Password required
11 0.136655 fd00:806:1::4 fd00:806:1::3 FTP Request: PASS bsi
12 0.171595 fd00:806:1::3 fd00:806:1::4 TCP ftp > drmsfsd [ACK] Seq=168 Ack=21 Win=16264 Le
13 0.234259 fd00:806:1::3 fd00:806:1::4 FTP Response: 230-User jcb has group access to: us
14 0.258557 fd00:806:1::4 fd00:806:1::3 FTP Request: QUIT
15 0.259917 fd00:806:1::3 fd00:806:1::4 TCP ftp > drmsfsd [ACK] Seq=277 Ack=27 Win=16264 Le
16 0.263247 fd00:806:1::3 fd00:806:1::4 FTP Response: 221-Goodbye. You uploaded 0 and downl
17 0.273408 fd00:806:1::4 fd00:806:1::3 TCP drmsfsd > ftp [FIN, ACK] Seq=27 Ack=344 Win=650
18 0.273741 fd00:806:1::3 fd00:806:1::4 TCP ftp > drmsfsd [FIN, ACK] Seq=344 Ack=27 Win=162
19 0.274051 fd00:806:1::4 fd00:806:1::3 FTP Request: \000\000\000\000\000\000\000\000\0
20 0.274466 fd00:806:1::3 fd00:806:1::4 TCP ftp > drmsfsd [ACK] Seq=345 Ack=28 Win=16264 Le
21 11.792902 fd00:806:1::4 fd00:806:1::3 TCP dncp > ftp [SYN] Seq=0 Win=65024 Len=0 MSS=8152
Frame 6 (205 bytes on wire, 205 bytes captured)
Ethernet II, Src: 02:00:00:00:00:00 (02:00:00:00:00:00), Dst: 02:00:00:00:00:1d (02:00:00:00:00:1d)
Internet Protocol Version 6
Transmission Control Protocol, Src Port: ftp (21), Dst Port: drmsfsd (4098), Seq: 1, Ack: 1, Len: 131
File Transfer Protocol (FTP)
220-Welcome to Pure-FTPd.\r\n
220-You are user number 1 of 50 allowed.\r\n
220 You will be disconnected after 15 minutes of inactivity.\r\n
0000 02 00 00 00 00 1d 02 00 00 00 00 00 86 dd 60 00
0010 00 00 00 97 06 40 fd 00 08 06 00 01 00 00 00 00
0020 00 00 00 00 00 03 fd 00 08 06 00 01 00 00 00 00
0030 00 00 00 00 00 04 00 15 10 02 90 78 ba 1d 00 9b
0040 7c 45 50 18 3f 88 bb 85 00 00 32 32 30 2d 57 65
0050 6c 63 6f 6d 65 20 74 6f 20 50 75 72 65 2d 46 54
0060 50 64 2e 0d 0a 32 32 30 2d 59 6f 75 20 61 72 65
0070 20 75 73 65 72 20 6e 75 6d 62 65 72 20 31 20 6f
0080 66 20 35 30 20 61 6c 6c 6f 77 65 64 2e 0d 0a 32
0090 32 30 20 59 6f 75 20 77 69 6c 6c 20 62 65 20 64
00a0 69 73 63 6f 6e 6e 65 63 74 65 64 20 61 66 74 65
Frame (frame), 205 bytes
Pac...
Profile: Default

**Chapter 3**

# Application Tracing

When writing or debugging socket applications it can be very useful to have traces of your application. Application level traces are available and can greatly assist you in developing and debugging applications.

# Enabling Traces

Application traces can be enabled in several ways. First, a JCL SETPARM can be used to start tracing when the application is executed. Second, the TRACEEZ stack command can be used to enable or disable application traces in a specified partition. And finally, the application itself can enable or disable tracing using a special call.

### SETPARM IPTRACE

To enable or disable application tracing using the SETPARM IPTRACE JCL statement ...

```
// SETPARM IPTRACE=XXXXXXXX
```

The X's in this statement are either 'Y' to enable a specific trace option or 'N' to disable a specific trace option. The meaning of the trace flags is discussed later in the manual.

### TRACEEZ

Using the TRACEEZ stack command ...

```
MSG BSTTINET,D=TRACEEZ id XXXXXXXX
MSG BSTT6NET,D=TRACEEZ id XXXXXXXX

E.g.,
MSG BSTTINET,D=TRACEEZ F2 YYNYNNNN
```

The X's in this statement are either 'Y' to enable a specific trace option or 'N' to disable a specific trace option. The meaning of the trace flags is discussed later in the manual. The id is the partition ID of the application.

### EZA SETSOCKOPT

Using the EZA SETSOCKOPT call ...

```
SETSOCKOPT.OPTNAME=SO_DEBUG
```

The application itself can enable tracing by issuing an EZA SETSOCKOPT call with the SO_DEBUG option enabled.

# EZA Traces

### Setting Trace Options

Trace options may be set using any of the following methods:

| | |
|---|---|
| JCL: | // SETPARM IPTRACE='yyyyyyyy' |
| Stack Command: | TRACEEZ xx yyyyyyyy |
| EZA CALL | SETSOCKOPT.OPTNAME=SO_DEBUG |

### Trace Options

There are 8 trace option flags available. Each trace option flag is indicated by the positional 'N' or 'Y' in the IPTRACE or TRACEEZ setting. (SETSOCKOPT.OPTNAME=SO_DEBUG will always use a setting of 'YNNYNNNN' for 'ON' or 'NNNNNNNN' for 'OFF'.) The 8 trace flags are:

```
'Y.......' produce base EZA trace information
'.Y......' produce trace information for BSI internal control blocks
'..Y.....' produce console messages on entry and exit to the EZA interface
'...Y....' produce trace information on SYSLST, not direct to LST queue
'....Y...' produce trace information for internal WAIT lists
'.....Y..' produce one line entry and exit trace messages
'......Y.' Force 'Y.......' for any call which results in an error
'.......Y' Trace full SEND/RECEIVE buffer
```

All flags can be used in combinations.

The default is to write all trace information (except console traces '..Y....') directly the VSE/POWER LST queue with a job name of 'EZALOGxx' (xx is the partition identifier). Many customers prefer to always use the '...Y....' flag to intersperse the trace output with normal or debug program output. Also, with heavy processing, the XPCC communication can slow the IP processing to the point that time based failures occur. Should this occur, use the '...Y...' flag as it is much faster to write to partition SYSLST. If writing to SYSLST, the trace will be interspersed with normal job output messages and may actually cause an overprint of the program output as the tracing function uses "write before advancing one line' commands.

### *Sample Output*

The is the sample output from an EZA SEND request. Each parameter of an EZASOKET or EZASMI call is detailed. This makes debugging very simple.

```
17:39:50.18226EC2 0004 ENTRY    IPS6 FUNCT    @=005B0430 V=14 SEND             S=00000050
                                     TOKEN    @=00540948 V=C2019001 005B7280 005FAD80 00540958
                                     S        @=00529741 V=0000     D=00000
                                     NBYTE    @=0052971D V=00000008 D=0000000000000008
                                     BUF      @=005295C8 V=C2E2E3E3 C5E9F0F3
                                     ALET     @=00000000 V=N/A
                                     FLAGS    @=005296C3 V=00000000 D=0000000000000000
                                     ECB      @=00000000 V=N/A
                                     ERRNO    @=005296BF V=00000000 D=0000000000000000
                                     RETCODE  @=0052973D V=00000000 D=0000000000000000

17:39:50.1DF6F640 0004 EXIT     IPS6 FUNCT    @=005B0430 V=14 SEND             S=00000050
                                     TOKEN    @=00540948 V=C2019001 005B7280 005FAD80 00540958
                                     S        @=00529741 V=0000     D=00000
                                     NBYTE    @=0052971D V=00000008 D=0000000000000008
                                     BUF      @=005295C8 V=N/A
                                     ALET     @=00000000 V=N/A
                                     FLAGS    @=005296C3 V=00000000 D=0000000000000000
                                     ECB      @=00000000 V=N/A
                                     ERRNO    @=005296BF V=00000008 D=0000000000000008
                                     RETCODE  @=0052973D V=00000008 D=0000000000000008
```

For each call, all parameters are listed. Following the name of the parameter is the address of the data (@=) and the hex value of the data (V=). When applicable, the decimal value of the data is shown (D=). The first line of each call includes a time-stamp and call sequence number.

### CICS TS Traces

When EZA tracing is enabled under CICS TS user trace entries are output to the CICS TS trace facility. Trace output is created only when running in the application domain (AP). Tracing under CICS/VSE is not supported. CICS TS user trace entries are created of type 38 (x'0026').

To enable or disable the tracing within CICS TS, use the CETR transaction.

***Sample Output***

```
AP 0026 USER  EVENT - APPLICATION-PROGRAM-ENTRY - IP6ENTRY - ..**....00000012CONNECT        **..CO.zV*....ER..z.......RT
              TASK-00047 KE_NUM-0042 TCB-00352000 RET-805EDF26 TIME-11:03:22.72 81501411 INTERVAL-00.0000309531    =003837=

                1-0000  E4E2C5D9 404040                                                  *USER                          *
                2-0000  00225C5C 0068B680 F0F0F0F0 F0F0F1F2  C3D6D5D5 C5C3E340 40404040 40404040  *..**....00000012CONNECT       *
                  0020  5C5C000A C3D600A9 E55C0004 000CC5D9  0401A90C 00000000 000CD9E3 0401A96C  ***..CO.zV*....ER..z.......RT..z%*
                  0040  00000000 0018E3D6 00A9EA68 C2920021  0387EE80 101B5E80 101B6580 000CF0F1  *......TO.z..Bk...g....;.......01*
                  0060  0401A970 00000000 000CF0F2 0401A93A  0002115C                          *..Z.......02..Z....*         *
                3-0000  C9D7F6C5 D5E3D9E8                                                  *IP6ENTRY                      *

AP 0026 USER  *EXCU - APPLICATION-PROGRAM-EXCEPTION - IP6EXITE - ..**....00000012CONNECT     **..CO.zV*....ER..z..../..RT

               TASK-00047 KE_NUM-0042 TCB-00352000 RET-805EDF26 TIME-11:03:25.30 10517504 INTERVAL-00.0000169375    =004182=
                1-0000  E4E2C5D9 C5E7C3                                                  *USEREXC                       *
                2-0000  00225C5C 0068B680 F0F0F0F0 F0F0F1F2  C3D6D5D5 C5C3E340 40404040 40404040  *..**....00000012CONNECT       *
                  0020  5C5C000A C3D600A9 E55C0004 000CC5D9  0401A90C 00000461 000CD9E3 0401A96C  ***..CO.zV*....ER..z..../..RT..z%*
                  0040  FFFFFFFF 0018E3D6 00A9EA68 C2920021  0387EE80 101B5E80 101B6580 000CF0F1  *......TO.z..Bk...g....;.......01*
                  0060  0401A970 00000000 000CF0F2 0401A93A  0002115C                          *..z.......02..z....*         *
                3-0000  C9D7F6C5 E7C9E3C5                                                  *IP6EXITE                      *
```

The Data-1 area is the type information.
USER or USEREXC (exception)

The Data-2 area is composed of multiple variable length strings of the following format:
- length (halfword)
- ID (two characters)
- source address of data (fullword)
- data (variable length)

The following IDs are used, and the data provided:
'**' header, data has three fields:
- sequence number (8 digits in display format)
- command (16 bytes of text)
- end tag (2 bytes of '**')

'CO' command, 2 bytes hex
'ER' errno,   4 bytes hex
'RT' retcode, 4 bytes hex
'TO' token,  16 bytes hex
'01' Parm1,   4 bytes hex
'02' - '10', Parameters 2 - 12 in same format as Parm1

The Data-3 area is the entry/exit information.
IP6ENTRY, IP6EXIT (exit without error), IP6EXITE  (exit with error, EXCEPTION trace)

# ASM SOCKET Traces

### Setting Trace Options

Trace options may be set using any of the following methods:
JCL: // SETPARM IPTRACE='yyyyyyyy'
Stack Command: TRACEEZ xx yyyyyyyy

### Trace Options

For compatibility with the EZA interface, there are 8 trace option flags available, but only 3 are referenced by the ASM SOCKET interface. Each trace option flag is indicated by the positional 'N' or 'Y' in the IPTRACE or TRACEEZ setting. The 3 trace flags available to the ASM SOCKET interface are:

```
'Y.......' produce base ASM SOCKET trace information
'.Y......' produce trace information for BSI internal control blocks
'...Y....' produce trace information on SYSLST, not direct to LST queue
'.......Y' Trace full SEND/RECEIVE buffer
```

All flags can be used in combinations.

The default is to write all trace information (except console traces '..Y....') directly the VSE/POWER LST queue with a job name of 'EZALOGxx' (xx is the partition identifier). Many customers prefer to always use the '...Y....' flag to intersperse the trace output with normal or debug program output. Also, with heavy processing, the XPCC communication can slow the IP processing to the point that time based failures occur. Should this occur, use the '...Y...' flag as it is much faster to write to partition SYSLST. If writing to SYSLST, the trace will be interspersed with normal job output messages and may actually cause an overprint of the program output as the tracing function uses "write before advancing one line' commands.

### *Sample Output*

```
17:13:11.5FF5DA40 0012 ENTRY    CSAP CSPL S T @=0053DE78 V=E2D60102 0055B000 03011200 00000000 *SO...í^.........*
                                                  00000000 00000000 00000000 00008CA0 *.............ðµ*
                                                  0054B508 00000000 0054C790 0000000A *.è§......èG°....*
                                                  002DCE60 E2C3D2E3 F6F60000 E2C3D2C4 *..ó-SCKT66..SCKD*
                                                  F6F60000 0000478C                   *66....åð       *
                                       S=00000478
                                  CSWK @=0055B000 V=00000150 0055B000 02474E98 002EA884 *...&.í^..å+q..yd*
                                                  8057B000 0055B000 0053DE78 0054B508 *Øï^..í^..ëúÌ.è§.*
                                                  0000000A 0054C780 A050070E 00000000 *.....èGØµ&......*
                                                  0053DE78 0055B000 005602A0 0054B000 *.ëúÌ.í^..í.µ.è^.*
                                                  002EA6A0 0053DD08 F6F6F6F6 000D0000 *..wµ.ëù.6666....*
                                                  00000000 00000000 5053DF46 0055B000 *........&ëÿã.í^.*
                                                  0055B000 0053DE78 0054B508 0000000A *.í^..ëúÌ.è§.....*
                                                  0054C780 A050070E 00000000 0054C780 *.èGØµ&.......èGØ*
                                                  8054D41C 005602A0 0054B000 0053DC60 *ØèM..î.µ.è^..ëü-*
                                                  0053DD08 0053DD08 00000000 00000000 *.ëù..ëù.........*
                                                  C2E2C9C6 D3C1C7E2 C2880000 024A5380 *BSIFLAGSBh...¢ëØ*
                                                  002DCE60 00000000 00000000 00000000 *..ó-............*
                                                  00000000 00000000 00
                                                  00000000 00000000 00000000 00000000 *................*
                                                  00000000 00000000 00000000 00000000 *................*
                                                  00000000 00000000 00000000 00000000 *................*
                                                  00000000 00000000 002EA6A0 00558000 *..........wµ.íØ.*
                                                  F6F6F6F6 000D0000 00000000 00000000 *6666............*
                                                  5054F3C6 00000000 0055B000 024A5D6C *&è3F.....í^..¢)%*
                                                  00000000 00000000 00000000 00000000 *................*
                                                  00000000 00000000 8057B000 00000000 *........Øï^.....*
                                                  00000000                            *....           *
                                  BUF  @=0054C790 V=55534552 206A6362 0D0A4040        *íëáê.¦ÄÂ..     *

17:13:11.618D5600 0012 EXIT     CSAP CSPL S T @=0053DE78 V=E2D60102 0055B000 03011200 00000000 *SO...í^.........*
                                                  00000000 00000000 00000000 00008CA0 *.............ðµ*
                                                  0054B508 00000000 0054C790 0000000A *.è§......èG°....*
                                                  002DCE60 E2C3D2E3 F6F60000 E2C3D2C4 *..ó-SCKT66..SCKD*
                                                  F6F60000 0000478C                   *66....åð       *
                                  CSWK @=0055B000 V=00000150 0055B000 02474E98 002EA884 *...&.í^..å+q..yd*
                                                  8057B000 0055B000 0053DE78 0054B508 *Øï^..í^..ëúÌ.è§.*
                                                  0000000A 0054C780 A050070E 00000000 *.....èGØµ&......*
                                                  0053DE78 0055B000 005602A0 0054B000 *.ëúÌ.í^..í.µ.è^.*
                                                  002EA6A0 0053DD08 F6F6F6F6 000D0000 *..wµ.ëù.6666....*
                                                  00000000 00000000 5053DF46 0055B000 *........&ëÿã.í^.*
                                                  0055B000 0053DE78 0054B508 0000000A *.í^..ëúÌ.è§.....*
                                                  0054C780 A050070E 00000000 0054C780 *.èGØµ&.......èGØ*
                                                  8054D41C 005602A0 0054B000 0053DC60 *ØèM..î.µ.è^..ëü-*
                                                  0053DD08 0053DD08 00000000 00000000 *.ëù..ëù.........*
                                                  C2E2C9C6 D3C1C7E2 C2880000 024A5380 *BSIFLAGSBh...¢ëØ*
                                                  002DCE60 00000000 00000000 00000000 *..ó-............*
                                                  00000000 00000000 00000000 00000000 *................*
                                                  00000000 00000000 00000000 00000000 *................*
                                                  00000000 00000000 00000000 00000000 *................*
                                                  00000000 00000000 002EA6A0 00558000 *..........wµ.íØ.*
                                                  F6F6F6F6 000D0000 00000000 00000000 *6666............*
                                                  5054F3C6 00000000 0055B000 024A5D6C *&è3F.....í^..¢)%*
                                                  00000000 00000000 00000000 00000000 *................*
                                                  00000000 00000000 8057B000 00000000 *........Øï^.....*
                                                  00000000 00000000 8057B000 00000000 *........Øï^.....*
                                                  00000000                            *....           *
```