Unicenter

TCPaccess
Prefixed Messages

Version 6.0

Computer Associates

The Software That Manages eBusiness

Booklist

Contents

Chapter 1

Overview of Messages

This manual describes the prefixed error message that can be issued by the following products:

- Unicenter TCPaccess
- Unicenter TCPaccess FTP

Note: Not all messages are issued by both products.

The following table describes:

- Types of messages issued
- How the messages are organized
- The log to which information is printed

Message Description

All messages are issued from the IJT task group to a single log data set and use the following form of identification:

pppmmxxxs

ppp A single letter followed by two numbers which identify the component:

- T00 Messages issued by IFS common routines.
- T01 Messages issued by the Unicenter TCPaccess routine.
- T02 Messages issued by the IUCV routine.
- T03 Messages issued by the Component Trace routine.

mm A pair of letters that identifies the subcomponent:

- AP Messages issued by the API common application support routines.
- CF Messages issued by Configuration.
- CO Message issued by Commutator service routines.
- DN Messages issued by the DNR task group.
- EX Messages issued.

F3 Messages issued by FTP3.

GD Messages issued by gateway dæmon (GateD).

IF Messages issued by infrastructure (IFS) task group.

IJ Messages issued.

IP Messages issued.

IU Messages issued by the IUCV transport provider.

LL Messages issued by the Link layer routine.

NT Messages issued by NETSTAT.

OE Messages issued by the UNIX System Services (OpenEdition) MVS transport provider.

PM Messages issued by the Port Mapper (IFS routines only).

RT Messages issued by the TelnetRTM.

S4 Messages issued by Spool #4.

SF Messages issued by Server FTP.

SN Messages issued by the Simple Network Management Protocol (SNMP).

SO Messages issued by the Socket API layer.

ST Messages issued by Server Telnet.

TC Messages issued by the Transport layer (TCP, UDP, Raw) routine.

TR Messages issued by the Trace facility

UD Messages issued by Server UDP mini services.

US Messages issued by User SMTP. (VT – Messages issued by VTAMAPPL.

VT Messages issued by VTAMAPPL.

xxx Number in the range of 000 through 999, uniquely identifying the message.

s Letter indicating the severity level of the message. It will be one of the following:

R Response message produced in response to a command.

F Fatal message signifying that a catastrophic error occurred.

E Error message indicating that some kind of error occurred and action should be taken to correct it.

W Warning message indicating that an action did not happen entirely correctly, but TCP could make adjustments or use defaults to correct the situation.

I Informational message informing the operator about operations and actions in progress.

S Statistic message displaying metrics to use for performance tuning, problem determination, and usage accounting.

D Debug message to help diagnose problems or provide additional information about operations and actions.

T Trace messages recording specific events as they occur.

Messages issued from the DNR, Port Mapper, SNMP, and GateD C routines are printed to the following DD names:

- DNRLOG and DNRERR
- MAPLOG and MAPERR
- SNMLOG and SNMERR
- GTDLOG, GTDTRC, and GTDERR

Messages issued from IFS and the commutator are printed to the T01LOG DD name.

Chapter 2

CON*nnn* CONVXL8 Messages

This chapter describes the messages issued by the CONVXL8 program.

CON*nnn*

CON100E Usage:CONVXL8 *InputDatasetName OutputDatasetName*

Reason: The arguments to this program are the input data set name and the output data set name. They must both be present.

Action: Reenter the command specifying both arguments.

CON101E Unable to open input file *filename*

Reason: The CONVXL8 program was unable to open the input file.

Action: Verify that the input data set exists and reissue the command.

CON102E Unable to open output file *filename*

Reason: The CONVXL8 program was unable to open the output file.

Action: Verify that the output data set exists and has the proper attributes. It must have RECFM=F, LRECL=256, and BLKSIZE=256.

CON103E Unable to write title to output file *filename*

Reason: The CONVXL8 program was unable to write the title to the output file.

Action: Verify that the output data set exists and that you have the proper authority to write to it making sure that there is sufficient space for the file.

CON104E Error processing input file

Reason: The input file contained an error. Other messages will describe the error.

CON105E Unable to write table to output file *filename*

Reason: The CONVXL8 program was unable to write a table to the output file. The file name is printed.

Action: Verify that the output data set exists and that you have the proper authority to write to it and that there is sufficient space for the file.

CON106I Processing completed normally.

Reason: The CONVXL8 program completed processing normally.

CON107E Input line too long:*inputline*

Reason: An input line contained too many characters. The maximum supported is 255 characters. The input line is printed.

Action: Correct the input and rerun the program.

CON108E Invalid character in input line: *inputline*

Reason: An input line contained a character that is not zero-nine or A-F. Only these characters are allowed outside of a comment. The input line is printed.

Action: Correct the input and rerun the program.

CON109E Input line contains too many codes: *inputline*

Reason: Exactly 16 codes must be specified on a line. That is 32 hexadecimal characters. The input line is printed.

Action: Correct the input and rerun the program.

CON110E Input line contains wrong number of bytes: *inputline*

Reason: The input line contained more then 16 codes or it contained an odd number of characters. The input line is printed.

Action: Correct the input and rerun the program.

CON111E Input contains wrong number of lines: *inputline*

Reason: Too many translate codes were present in the input file. The input line is printed.

Action: Correct the input and rerun the program.

CON112E Premature end of file on input.

Reason: There were not enough lines of translate codes in the input file.

Action: Correct the input and rerun the program.

Chapter 3

DNR1*nn* DNR Log Messages

This chapter describes messages written to logs by the Domain Name Resolver (DNR).

DNR100 Through DNR199

DNR100E *parm*(*mem*) : FORMAT ERROR ON LINE *num*

Reason: An error occurred parsing the line extension syntax in the specified DNR secondary configuration member.

parm Parameter name in the DNR Primary Configuration Member (DNRCFG*xx*).

mem Name of the secondary configuration member where the error occurred.

num Line number of the error (in decimal).

Action: Correct the indicated entry or remove the entry from the configuration member. Refer to the *Customization Guide* for line extension syntax. Stop the DNR task group and restart it to ensure the configuration information is processed accurately.

DNR101E *parm*(*mem*) : LENGTH ERROR ON LINE *num*

Reason: An error occurred parsing the specified DNR secondary configuration member.

parm Parameter name in the DNR Primary Configuration Member (DNRCFG*xx*).

mem Name of the secondary configuration member where the error occurred.

num Line number of the error (in decimal).

Action: Decrease the length of the indicated field or remove the entry from the configuration member. Refer to the *Customization Guide* for maximum field lengths. Stop the DNR task group and restart it to ensure the configuration information is processed accurately.

DNR102E *parm*(*mem*) : MISSING FIELD ON LINE *num*

Reason: An error occurred parsing the specified DNR secondary configuration member.

parm Parameter name in the DNR Primary Configuration Member (DNRCFG*xx*).

mem Member name of the secondary configuration member where the error occurred.

num Line number of the error (in decimal).

Action: Add the missing field or remove the entry from the configuration member. Refer to the *Customization Guide* for the field definitions. Stop the DNR task group and restart it to insure the configuration information is processed accurately.

DNR103E *parm*(*mem*) : ILLEGAL PARAMETER ON LINE *num*

Reason: An error occurred parsing the specified DNR secondary configuration member.

parm Parameter name in the DNR Primary Configuration Member (DNRCFG*xx*).

mem Member name of the secondary configuration member where the error occurred.

num Line number of the error (in decimal).

Action: Correct the field or remove the entry from the configuration member. Refer to the *Customization Guide* for field specifications. Stop the DNR task group and restart it to ensure the configuration information is processed accurately.

DNR104E *parm*(*mem*) : ILLEGAL DOMAIN NAME ON LINE *num*

Reason: An occurred parsing the specified DNR secondary configuration member.

parm The parameter name in the DNR Primary Configuration Member (DNRCFG*xx*).

mem The member name of the secondary configuration member where the error occurred.

num The line number of the error (in decimal).

Action: Correct the field or remove the entry from the configuration member. Refer to the *Customization Guide* for field specifications. Stop the DNR task group and restart it to ensure the configuration information is processed accurately.

DNR Network Trace Messages

This section describes DNR network trace messages DNR120I and DNR121I. The DNR includes an optional trace of DNR network requests. Trace outputs DNR request information to the SYSPRINT data set. DNR120I and DNR121I display if the DNR task group DNRCFG*xx* member specifies TRACE (the default is NOTRACE).

Example

Here is an example of the trace output provided:

```
DNR120I : Wed Sep 05 16:03:55 1990: Request #12 - Sending DNS Query #16 RD

    Question           : TERP.UND.EDU                    Type : WKS
    Domain Server      : 129.192.224.160                Transmit No.  : 1
DNR121I :   Wed Sep 05 16:02:35 1990: Request #0 - Received a DNS Response for
            Query #1 RD RA RCODE = No error
    Question           :INTERLINK.COM            Type :   NS
    Answer             :INTERLINK.COM            Type :   NS      UUNET.UU.NET
    Answer             :INTERLINK.COM            Type :   NS
SEISMO.CSS.GOV
    Authorization      :INTERLINK.COM            Type :   NS      UUNET.UU.NET
    Authorization      :INTERLINK.COM            Type :   NS
SEISMO.CSS.GOV
    Additional Info    :UUNET.UU.NET             Type :   A       192.48.96.2
    Additional Info    :SEISMO.CSS.GOV           Type :   A       192.12.141.25
    Domain Server      :129.192.224.160
```

DNR120I REQUEST #*req_id* - SENDING DNS QUERY # *msg_id* [RD]
QUESTION :*q_name* TYPE: *q_type*
DOMAIN SERVER :*nsia* TRANSMIT NO.: *scnt*

Reason: A Domain Name System message has been sent. This message appears only if the DNR task group DNRCFG*xx* member specifies TRACE (the default is NOTRACE).

req_id ID (in decimal) from the client's request.

msg_id Message ID (in decimal) number of the network query.

[RD] If present, indicates the recursion desired bit is set.

q_name Domain name of the node of the query.

q_type Specifies one of these types of queries:

A	Requests the Internet address(es) of the host specified in *q_name*.
CNAME	Requests the canonical name for the alias specified in *q_name*.
HINFO	Request general information about the host specified in *q_name*.
MX	Requests a list of hosts willing to act as a mail exchange for the host specified in *q_name*.
NS	Requests domain name servers that are authoritative for the domain specified in *q_name*.
PTR	Requests a pointer to a location in the domain name space.
WKS	Requests the well-known services supported by the host specified in *q_name*.

nsia Internet address of the remote name server queried.

scnt Transmission count (in decimal).

Action: None. This is an Informational message.

DNR121I REQUEST *req_id* RECEIVED A DNS RESPONSE FOR QUERY #*msg_id indicator*
RCODE = *rcode*
QUESTION: *rname* TYPE : *qtype*
ANSWER: *rdata* TYPE : *qtype*
AUTHORIZATION: *rdata* TYPE : *qtype*
.
.
.
ADDITIONAL INFO: *rdata* TYPE : *qtype*
.
.
.
DOMAIN SERVER: *rdata*

Reason: A Domain Name System message was received. This message is displayed only if the DNR task group DNRCFG*xx* member specifies TRACE (default is NOTRACE).

req_id ID (in decimal) from the client's request.

msg_id Message ID (in decimal) of the network query copied into the response.

indicator One of the strings described in this table:

AA Indicates that the responding name server is an authority for the domain name in the question section.

RD Indicates the recursion desired bit was set in the query and is copied into the response.

RA Indicates that recursive query support is available in the name server.

rcode One of these response codes is specified in the response:

No Error	Signifies there is no error condition.
Format Error	Signifies the name server was unable to interpret the query.
Server Failure	Signifies the name server was unable to process the query due to a problem with the name server.
Name Error	Signifies the domain name in the question section does not exist.
Not Implemented	Signifies the name server does not support the requested kind of query.
Refused	Signifies the name server refuses to perform the specified operation for policy reasons.

rname Name of the node to which the specified data (that is, rdata) pertains.

rdata Data portion of the response. The data varies according to query type.

If no data is present and the AA bit is set in the response, this indicates the *rname* is a valid name but there is no data configured for this type of request.

qtype value varies depending on the query type:

A *ia*	Shows a dotted decimal internet address of the host specified in *rname*.
CNAME *cname*	Shows a domain name, *cname*, specifing the canonical or primary name for the name specified in *rname*.
HINFO *cpu os*	Shows the CPU type, *cpu; os* is operating system type of the *rname* host.
MX *pref exchc*	A decimal that specifies the preference given to the exchange host among others at the same host. Lower values are preferred. *exch* is a domain name that specifies a host willing to act as a mail exchange for the owner.
NS *ns*	Is a domain name that specifies a host that should be authoritative for the domain specified in *rname*.
PTR *pn*	Is a domain name that points to some location in the domain name space.
SOA	Data is not traced by the DNR task group.
KS *ia pro* serv	Displays a list of well-known services *serv* (in decimal) supported by the protocol given in *pro* (in hexadecimal) on the Internet address specified in *ia*.

Action: None. This is an Informational message.

DNR Internal Trace Messages

This section describes DNR internal trace messages DNR122I through DNR152I. The DNR includes an optional trace of DNR request processing. The trace, if selected, outputs DNR request information to the SYSPRINT data set. DNR122I through DNR124I display if the DNR task group DNRCFGxx member specifies INTERNALTRACE (the default is NOINTERNALTRACE). All DPL return codes are documented in the DNR Return Codes in *Unicenter TCPaccess Communications Server Unprefixed Messages*.

Example

The following are examples of the trace output provided:

```
DNR122I :      Wed Sep 05 16:02:45 1990: Request #4 - Processing Request

    GET HOST BYVALUE SYNC GLOBAL BLOCK
    Name Buffer (1024)
    Value Buffer (4) 81c0e068 ( 129.192.224.104)
    Time(30)
DNR123I :      Wed Sep 05 16:01:44 1990: Request #1: Rescheduled Query
               Response Received
    GET HOST BYVALUE
DNR124I :      Wed Sep 05 16:01:44 1990: Request #2 Reply Sent (CCOMP)
               Processing Completed (CDONE)
    GET HOST BYVALUE ASYNC GLOBAL BLOCK
    Name Buffer   (21)  'UNIX.MD.INTERLINK.COM.'
    Value Buffer  (4)  81c0e078 ( 129.192.224.120 )
    Time (0)
    Action Code=x'00'  Error Code=x'00'  Diagnostic Codes=x'00'
```

DNR122I REQUEST #*req_id* PROCESSING REQUEST GET request opsync [opglob] [opblk] NAME BUFFER (nalen) VALUE BUFFER (valen) QNAME BUFFER (qnlen) TIME (time) SIZE (size)

Reason: Processing of a DNR request has started. This message displays input buffer length and contents and the length of the return data areas.

req_id The message ID (in decimal) of the request. This is included when the retry and response trace messages display.

request One of these strings based on the value of the DPL function code:

HOST BYNAME DPL function code = 1.

HOST BYVALUE DPL function code = 2.

HOST BYALIAS DPL function code = 3.

NETWORK BYNAME DPL function code = 4.

NETWORK BYVALUE DPL function code = 5.

SERVICE BYNAME DPL function code = 6.

SERVICE BYVALUE DPL function code = 7.

PROTOCOL BYNAME DPL function code = 8.

PROTOCOL BYVALUE DPL function code = 9.

HOSTSERV BYNAME DPL function code = 10.

HOSTINFO BYNAME DPL function code = 11.

ROUTE BYNAME DPL function code = 12.

RPC BYNAME DPL function code = 13.

RPC BYNAME SERVER BYNAME DPL function code = 15.

VALUE DPL function code = 14.

SOA BYNAME DPL function code = 16.

opsync One of these strings based on the DPL asynchronous/synchronous option bit:

ASYNC Signifies the asynch/synch option bit was set.

SYNC Signifies the asynch/synch option bit was not set.

opglob One of these strings based on the DPL global/local option bit if the bit is applicable for the request:

GLOBAL Signifies the global/local option bit was not set.

LOCAL Signifies the global/local option bit was set.

opblk	One of these strings based on the DPL block/noblock option bit if the bit is applicable for the request: BLOCK Signifies the block/noblock option bit was not set. NOBLOCK Signifies the block/noblock option bit was set.
nalen	Value (in decimal) of DPL NALEN field.
valen	Value (in decimal) of DPL VALEN field.
qnlen	Value (in decimal) of DPL QNLEN field if applicable for the request.
size	Value (in decimal) of the DPL SIZE field if applicable for the request.
time	Value (in decimal) of the DPL TIME field if applicable for the request.

Action: None. This is an Informational message.

DNR123I REQUEST #*req_id*: RESCHEDULED QUERY RESPONSE *reason* GET *request*

Reason: A DNR request was rescheduled. Requests are rescheduled if a network reply was received for the request or if the retry timer has expired. Only requests that send Domain Name System requests are rescheduled; requests that involve local static table complete immediately.

Syntax Description

req_id	The message ID (in decimal) of the request.
reason	One of these strings explaining why the request was rescheduled: RETRY signifies the request is being rescheduled because the retry timer expired. RECEIVE signifies the request is being rescheduled because a network response was received.
request	A string representing the value of the DPL function code Refer to DNR122I description for specific values.

Action: None. This is an Informational message.

DNR124I REQUEST #*req_id* REPLY SENT (CCOMP) PROCESSING COMPLETED (CDONE) GET request opsync [opglob] [opblk] NAME BUFFER (nalen) REQUEST VALUE BUFFER (valen) QNAME BUFFER (qnlen) SIZE (size) TIME (time) ACTION CODE=acode ERROR CODE=ercode DIAGNOSTIC CODES=dgcode

Reason: DNR request has completed. This message only appears if the DNR task group DNRCFG*xx* member specifies INTERNALTRACE (default is NOINTERNALTRACE).

req_id The message ID (in decimal) of the request:

CCOMP signifies a response is returned to the calling program.

CDONE signifies the request processing is complete.

Both CCOMP and CDONE appear if a request involved local static tables or was issued as a BLOCK request. If the request specified NOBLOCK, the request processing may not complete when a response is returned to the application program (CCOMP). If a NOBLOCK request is received, the DNR consults the DNR cache and completes the request but continues processing the request if the information was not found. CDONE indicates the DNR is through processing the request.

request A string representing the value of the DPL function code. See DNR122I description for specific values.

opsync ASYNC or SYNC (see DNR122I).

opglob GLOBAL or LOCAL (see DNR122I).

opblk BLOCK or NOBLOCK (see DNR122I).

nalen The value of DPL NALEN field.

valen The value of DPL VALEN field.

qnlen The value of DPL QNLEN field if applicable for the request.

size The value of the DPL SIZE field if applicable for the request.

time The value of the DPL TIME field if applicable for the request.

acode The DPL action code (in hexadecimal).

ercode The DPL error code (in hexadecimal).

dgcode The DPL diagnostic code (in hexadecimal).

Action: None. This is an Informational message.

DNR125I REQUEST #*req_id* - API HAS ACKNOWLEDGED SENDING DNS QUERY #*msg_id*

Reason: The DNR has sent a query (*req_id*) to the API and the API acknowledges that it was sent.

Action: None. This is an Informational message.

DNR150E *call* FAILED: R15=*rcode* ACTION CODE=*acode* ERROR CODE=*errcode* DIAGNOSTIC CODE=*dgcode*

Reason: The DNR issued the operation call and it failed.

call	The DIRSRV call that's failing: AOPEN, ACLOSE, or a TPL function.
rcode	The return code in Register 15 (in hexadecimal).
acode	The API action code (in hexadecimal).
ercode	The API error code (in hexadecimal).
dgcode	The API diagnostic code (in hexadecimal) for AOPEN, ACLOSE, or the TPL function.

Action: Determine the error from the return codes listed.

DNR150I *call* SUCCESSFUL

Reason: The DNR has issued the operation call and it was successful.

call can be AOPEN, TOPEN, TBIND, TCLOSE, or ACLOSE.

Action: None. This is an Informational message.

DNR151I TCLOSE INITIATED DUE TO *reason*

Reason: The DNR is closing the transport provider (UDP) connection due to the reason given.

reason Can have one of these meanings:

- Transport Provider Termination signifies the ACP task group notified the DNR that it is no longer providing services
- API Termination signifies The API task group notified the DNR that it is no longer providing services
- DNR Termination signifies the DNR task group is terminating

Action: None. This is an Informational message.

DNR152I ACLOSE INITIATED DUE TO *reason*

Reason: The DNR is closing the API connection due to the reason given.

reason Can have one of these meanings:

- API Termination signifies the API task group notified the DNR that it is no longer providing services
- DNR Termination signifies the DNR task group is terminating

Action: None. This is an Informational message.

DNR153I ROUTINE CRRLKUP RETURNED INVALID VALUES rrp = aaaaaaaaa, rrcnt = x'cccc', type = x'tt

Reason: Routine *crrlkup* searches the DNR cache for resources records of type *tt*. The routine returns a pointer (*aaaaaaaa*) to the first in a linked list of resource records of type *tt* and the count (*cccc*) of resource records in the list. One of the DNR completion routines has determined that either the pointer to the first resource record or the count of resource records is invalid. This check is done to prevent storage overlays. The associated DIRSRV request is terminated with error code DESYSERR. The link list of resource records, *d rrs*, is printed in hexadecimal format for diagnostic support.

The fields in the DNR153I message are:

aaaaaaaa (hex)	Address of the resource record, *d_rr*.
cccc (hex)	Count of resource records as calculated by *crrlkup*.
tt (hex)	Resource crecord type code used by *crrlkup*.
xxxxxxxx (hex)	Each resource record is dump (10 words per record).

Action: This message is indicative of an internal logic error in the DNR. Issue the console command DNR SNAP ALL to get a formatted dump of the DNR control blocks. If the problem is possible to recreate, recreate the problem while DNR is running with the INTERNALTRACE option specified in DNRCFG00.

After getting the output of the DNR SNAP command, contact Customer Support for assistance.

DNR300 Through DNR399

This section describes DNR dump operator command messages, DNR300 through DNR399. The DNR includes an option that lets an operator display the Domain Name System (DNS) cache and static tables built from reading the DNR configuration members.

Example

Here is an example of the dump output provided:

```
DNR300I DNR DUMP CACHE

DNS Cache :
    . NS NS.NIC.DDN.MIL.
        MIL.
            DDN.MIL.
                NIC.DDN.MIL.
                    NS.NIC.DDN.MIL. A 192.67.67.53
                    NS.NIC.DDN.MIL. A 1.2.3.4
        COM.
            ACC.COM.  NS  SALT.ACC.COM.
                A.ACC.COM.  DENAMERR
                SALT.ACC.COM.  A  129.192.64.1
                SALT.ACC.COM.  MX  10  SALT.ACC.COM.
                SALT.ACC.COM.  MX  20  SATURN.ACC.COM.
                SALT.ACC.COM.  HINFO  MICROVAX-II UNIX
                SALT.ACC.COM.  WKS  not displayed
                SATURN.ACC.COM.  WKS  DENODATA
                VAX1.ACC.COM.  CNAME  SALT.ACC.COM.
        ARPA.
            IN-ADDR.ARPA.
                127.IN-ADDR.ARPA.
                    0.127.IN-ADDR.ARPA.
                        0.0.127.IN-ADDR.ARPA.
                            1.0.0.127.IN-ADDR.ARPA. PTR LOCALHOST
Static Alias Table :
    ACSS  ZEUS.ACC.COM.
    SALT  SALT.ACC.COM.
Static Search List :
    .
    ACC.COM.
    COM.
Static Nameserver List :
    .  NS.NIC.DDN.MIL.  192.67.67.53
    ACC.COM.  SALT.ACC.COM.  129.8.10.90
Static Network Preference List :
    129.192.64.0
Static Host Name Table :
    ZEUS.ACC.COM.  129.192.192.130
    ZEUS.ACC.COM.  129.192.192.131
Static Network Table :
    ISI-NET   128.9.0.0
    COLUMBIA_PDN   129.192.128.0
Static Protocol Table :
    TCP 6
    UDP 17
Static Services Table :
    TCP/DOMAIN     53
    UDP/DISCARD     9
```

```
Static RPC Table :
    PORTMAPPER     100000 PORTMAP SUNRPC
    STATUS    100024
DNR301I END DNR DUMP
```

DNR300I DNR DUMP *type*

Reason: A DNR DUMP operator command was issued from the console. This message displays the DNS cache and static tables based on the command parameters. The type parameter indicates the operator's request.

type can be one of the values in the following list. The output for each value follows the list.

blank	The operator requested a dump of the DNS cache and all DNR static tables.
CACHE	The operator requested a dump of the DNS cache.
STATIC	The operator requested a dump of all static tables.
STATIC(ALIAS)	The operator requested a dump of the Static Alias Table.
STATIC(SEARCHLIST)	The operator requested a dump of the Static Search List.
STATIC(NAMESERVER)	The operator requested a dump of the Static Nameserver List.
STATIC(NETPREF)	The operator requested a dump of the Static Network Preference List.
STATIC(HOST)	The operator requested a dump of the Static Host Name Table.
STATIC(NETWORK)	The operator requested a dump of the Static Network Table.
STATIC(PROTOCOL)	The operator requested a dump of the Static Protocol Table.
STATIC(SERVICES)	The operator requested a dump of the Static Services Table.
STATIC(RPC)	The operator requested a dump of the Static RPC Table.

These DNR DUMP types are described on the following pages.

Action: None. This is an Informational message.

CACHE

DNS Cache :
dname [data]
dname [data]
dname [data]
.
.
.
dname [data]

dname The domain name of the node.

data The RR type and data if information is stored or the status of the name A DNR DUMP or DNR DUMP CACHE operator command was issued. The DNR dumps the contents of the DNS cache. The DNS cache is a tree-structured cache with each level containing a name. Each name in turn can have data (RRs) associated with it.

This list describes the possible data text:

no data The name is a valid name but there is no RR data stored for the name (a child is present).

DENAMERR The domain name does not exist.

A *ia* An Address (A) RR is stored for *dname*.

ia is a dotted decimal internet address of the host specified in *dname*.

CNAME *cname* A Canonical (CNAME) RR is stored for *dname*.

cname is a domain name that specifies the canonical or primary name for the name specified in *dname*.

HINFO *cpu os* A Host Information (HINFO) RR is stored for *dname*.

cpu is the CPU type.

os is the operating system type of the *dname* host.

MX *pref exch* A Mail Exchange (MX) RR is stored for *dname*.

pref is a decimal that specifies the preference given to the exchange host (*exch*) among others at the same host (*dname*). Lower values are preferred.

exch is a domain name that specifies a host willing to act as a mail exchange for the *dname* host.

NS *ns*	A Name Server (NS) RR is stored for *dname*. *ns* is a domain name that specifies a host that is authoritative for the domain specified in *dname*.
PTR *pn*	A Pointer (PTR) RR is stored for *dname*. *pn* is a domain name that points to some location in the domain name space.
WKS not displayed	Well Known Service (WKS) RR stored for *dname*. WKS information is not dumped.
type DENODATA	*type* is a DNS RR type as previously described. DENODATA means that the domain name is valid but the authoritative name server does not have the specific RR configured for *dname*.

STATIC(ALIAS)
Static Alias Table :
alias replace
alias replace
.
.
.
alias replace

Reason: A DNR DUMP, DNR DUMP STATIC, or DNR DUMP STATIC(ALIAS) operator command was issued. The DNR dumps the contents of the Static Alias Table read from DNRCFGxx ALIAS(DNRALC*xx*) member.

alias	The alias.
replace	The replacement string.

STATIC(SEARCHLIST)
Static Alias Table :
domain
domain
.
.
.
domain

Reason: A DNR DUMP, DNR DUMP STATIC, or DNR DUMP STATIC(SEARCHLIST) operator command was issued. The DNR dumps the contents of the Search List Table read from DNRCFG*xx* SEARCHLIST(DNRSLC*xx*) member and search list entries built based on the subsystem name entered in the Alias Configuration member (ALIAS(DNRALC*xx*)).

One parameter, *domain*, is returned. *domain* is a fully-qualified domain name used to append to an incomplete domain name search string.

STATIC(NAMESERVER)
Static Nameserver List:
domain nameserver nsiaddr
domain nameserver nsiaddr
.
.
.
domain nameserver nsiaddr

Reason: A DNR DUMP, DNR DUMP STATIC, or DNR DUMP STATIC(NAMESERVER) operator command was issued. The DNR dumps the contents of the Name Server Table read from DNRCFG*xx* NAMESERVER(DNRNSC*xx)* member.

Syntax Description

domain	Fully-qualified domain name.
nameserver	Name server authoritative for the domain referenced by domain.
nsiaddr	Internet address in decimal dot notation of the host referenced by *nameserver*.

STATIC(NETPREF)
Static Network Preference List :
network
network
.
.
.
network

A DNR DUMP, DNR DUMP STATIC, or DNR DUMP STATIC(NETPREF) operator command was issued. The DNR dumps the contents of the Network Preference Table read from DNRCFG*xx* NETWORKPREF(DNRNPC*xx*) member. The single parameter, *network*, is returned. *network* is a network address in dotted decimal notation.

STATIC(HOST)
Static Host Name Table:
hostname hostaddress
hostname hostaddress
.
.
.
hostname hostaddress

Reason: A DNR DUMP, DNR DUMP STATIC, or DNR DUMP STATIC(HOST) operator command was issued. The DNR dumps the contents of the Host Name Table read from DNRCFG*xx* HOSTTABLE(DNRHST*xx*) member.

hostname Fully qualified host name.

hostaddress Internet address in dotted decimal notation assigned to the host name referenced in *hostname*.

STATIC(NETWORK)

Static Network Table:

networkname networkaddress
networkname networkaddress
.
.
.
networkname networkaddress

Reason: A DNR DUMP, DNR DUMP STATIC, or DNR DUMP STATIC(NETWORK) operator command was issued. The DNR dumps the contents of the Network Table read from DNRCFG*xx* NETWORK(DNRNET*xx*) member.

networkname Network name.

networkaddress Network address in dotted decimal notation assigned to the network referenced in *networkname*.

STATIC(PROTOCOL)

Static Protocol Table:

protocolname portnumber
protocolname portnumber
.
.
.
protocolname portnumber

Reason: A DNR DUMP, DNR DUMP STATIC, or DNR DUMP STATIC(PROTOCOL) operator command was issued. The DNR dumps the contents of the Protocol Table read from DNRCFG*xx* PROTOCOL(DNRPRT*xx*) member.

protocolname Protocol name.

portnumber Protocol number assigned to the protocol name referenced in *protocolname*.

STATIC(SERVICES)
Static Service Table:
protocol/service portnumber
protocol/service portnumber
.
.
protocol/service portnumber

Reason: A DNR DUMP, DNR DUMP STATIC, or DNR DUMP STATIC(SERVICES) operator command was issued. The DNR dumps the contents of the Services Table read from DNRCFG*xx* SERVICES(DNRSVC) member.

protocol	Protocol name.
service	Service name.
portnumber	Port number (in decimal) associated with the *protoco*l and *service* name pair.

STATIC(RPC)
Static RPC Table:
rpcname rpcnumber [rpcalias] . . . [rpcalias]
rpcname rpcnumber [rpcalias] . . . [rpcalias]
.
.
.
rpcname rpcnumber [rpcalias] . . . [rpcalias]

Reason: A DNR DUMP, DNR DUMP STATIC, or DNR DUMP STATIC(RPC) operator command was issued. The DNR dumps the contents of the RPC Name Table read from DNRCFG*xx* RPCNAMES(DNRRPC*xx*) member.

rpcname	RPC name.
rpcnumber	RPC number (in decimal) assigned to the RPC name referenced in *rpcname*.
rpcalias	Alias for the name referenced in *rpcname* (maximum five).

DNR3011 END DNR DUMP

Reason: The DNR dump is completed.

Action: None. This is an Informational message.

DNR302I DNR DUMP CACHE(DATA(*name*))
[STATIC ALIAS TABLE ENTRY :] [*name replace*]
CACHE ENTRY :
name*.[*searchlistentry*] - *data
name*.[*searchlistentry*] - *data
.
.
.
name*.[*searchlistentry*] - *data

Reason: The DNR includes an option that enables an operator to display the Domain Name System (DNS) cache for a specific name. Included in the dump is the display of the logic used to search the directory. That is, if the search string is in the Static Alias Table or if the search string is not a fully-qualified name, the display includes the search information. This is the possible data text:

```
not in cache - The domain name is not in the DNS cache.
```

See the DNR300I DNR DUMP CACHE data description for valid data information.

name	The domain name given in the operator command.
replace	The alias replacement string if *name* is an alias given in the Alias Table. If replace is an internet address, a message is displayed and the dump is complete (DIRSRV returns if replace is an internet address).
searchlistentry	The search list entry used to create a fully qualified domain name if *name* is not a fully-qualified domain name.
data	The RR type and data if information is stored, or the status of the name.

Action: None. This is an Informational message.

Example

The following is an example of a DNR302I message:

```
DNR302I DNR DUMP CACHE(DATA(MVS))
Static Alias Table Entry :
    MVS   ZEUS
Cache Entry :
    ZEUS. - not in cache
    ZEUS.ACC.COM.  A  129.192.192.129
    ZEUS.ACC.COM.  HINFO  IBM  MVS
    ZEUS.COM. - not in cache
```

A DNR DUMP CACHE(DATA(*name*)) operator command was issued. The DNR dumps the Static Alias Table if appropriate and the contents of the DNS cache.

DNR303I DNR DUMP CACHE(NAMES))
DNS Cache :
***dname* [*data*]**
***dname* [*data*]**
***dname* [*data*]**
.
.
.
***dname* [*data*]**

Reason: A DNR DUMP NAMESERVER(*name*) operator command was issued. The DNR dumps the Static Alias Table if appropriate and the dynamic nameserver list built for the name.

dname	Domain name of the node.
data	String indicating if information is stored or the status of the name The DNR includes an option that enables an operator to display the DNS cache names only. This is the possible data text:
no data	Name is a valid name but there is no RR data stored for the name (a child is present).
RRS	There are RRs stored for this *dname*.
DENAMERR	Domain name does not exist.

Action: None. This is an Informational message.

Example

The following is an example of a DNR303I message.

```
DNR303I DNR DUMP CACHE(NAMES)
DNS Cache :
A DNR DUMP NAMESERVER(name)) operator command was issued. The DNR dumps the
Static
Alias Table if appropriate and the dynamic nameserver list built for the name.
.  (root) RRS
     MIL.
         DDN.MIL.
             NIC.DDN.MIL.
                 NS.NIC.DDN.MIL.  RRS
     COM.
         ACC.COM.  RRS
             A.ACC.COM.  DENAMERR
             SALT.ACC.COM.  RRS
             SATURN.ACC.COM.  RRS
             AX1.ACC.COM.  RRS
     ARPA.
             IN-ADDR.ARPA.
                 127.IN-ADDR.ARPA.
                     0.127.IN-ADDR.ARPA.
                     0.0.127.IN-ADDR.ARPA.
                     1.0.0.127.IN-ADDR.ARPA.  RRS
```

DNR304I DNR DUMP NAMESERVER(name)
[STATIC ALIAS TABLE ENTRY :]
[*name replace*]
NS LIST :
***name*.[*searchlistentry*] :**
nsiaddr nameserver domain
nsiaddr nameserver domain
.
.
.
nsiaddr nameserver domain
***name*.[*searchlistentry*] :**
nsiaddr nameserver domain
nsiaddr nameserver domain
.
.
.
nsiaddr nameserver domain

Reason: The DNR includes an option that enables an operator to display the name server list built for a name. The name server list is built using name servers configured in the Nameserver Configuration Member and the Name server information received in DNS responses. Name servers are in turn sorted by the information given in the Network Preference Configuration Member.

name	Domain name given in the operator command.
replace	Alias replacement string if name is an alias given in the Alias Table. If replace is an internet address, a message is displayed and the dump is complete (DIRSRV returns if replace is an internet address).
searchlistentry	Search list entry used to create a fully-qualified domain name if name is not a fully-qualified domain name.
nsiaddr	Internet address in decimal dot notation of the host referenced by *nameserver*.
nameserver	Name server authoritative for the domain referenced by *domain*.
domain	Fully qualified domain name.

Action: None. This is an Informational message.

Example

This is an example of a DNR304I message:

```
DNR304I DNR DUMP NAMESERVER(MVS)
Static Alias Table Entry :
ZEUS  ZEUS.ACC.COM.
NS List :
129.192.224.128  SATURN.ACC.COM.  ACC.COM.
129.192.64.1  SALT.ACC.COM.  ACC.COM.
192.67.67.53  NS.NIC.DDN.MIL.
128.20.1.2  AOS.BRL.MIL.  .
```

A DNR DUMP NAMESERVER(*name*) operator command was issued. The DNR dumps the Static Alias Table if appropriate and the dynamic *nameserver* list built for the name.

Chapter 4

GTD*nnn* GateD Log Messages

This chapter describes messages written to logs by the GateD facility.

GTD*nnn*

GTD001E timer_peek: gettimeofday: *YYYY*

Reason: A call to *gettimeofday*( ) returned an error in subroutine *timer_peek*. This condition should never occur since the implementation of *gettimeofday*( ) never sets an error value.

YYYY Reason of the *errno* value returned by *gettimeofday*( ).

GTD002E timer_dispatch: alarm: *YYYY*

Reason: Call to *alarm*( ) returned a negative value that is., a negative amount of time left from the previous call to *alarm*( ). *YYYY* is the Reason of the *errno* value returned by *alarm*( ).

YYYY Reason of the *errno* value returned by *alarm*( ).

GTD003N Abort *XXXX(NNNN)* version *NNNN: YYYY*

Reason: A call was made to *task_quit*( ) specifying a non-zero error code. GateD is aborting. This message is generally preceded by one or more messages that identify in further detail an error that let to the abort. A system dump should be generated when this occurs.

XXXX GateD task name.

NNNN PID of the task.

NNNN GateD version number.

YYYY Reason of the last *errno* value set.

GTD004E Assertion failed *XXXX(NNNN)*: file "*VVVV*", line *YYYY*

Reason: GateD performed an assertion test by calling the function *assert*( ) and the assertion test failed. GateD operation aborts.

XXXX GateD task name.

NNNN PID of the task.

VVVV Program name.

YYYY Line number in program.

GTD005E task_receive_packet: *XXXX* recvfrom/recvmsg: *YYYY*

Reason: A call to the socket function *recvfrom*() failed due to the reason listed in the errno Reason.

XXXX GateD task name.

YYYY Reason of the *errno* value set by *recvfrom*.

GTD006N Commence routing updates

Reason: Notifies user that routing updates are starting.

GTD007E task_main: select: *YYYY*

Reason: A call to the socket function *mvsselect*( ) failed due to the reason listed in the *errno* Reason.

GTD008E task_main: no accept method for *XXXX* socket *YYYY*

Reason: A call to *mvsselect*() returned with the socket marked readable. The socket is currently in a state requiring a call to *accept*(), however, no accept routine is set up for this GateD task.

XXXX GateD task name.

YYYY Socket identifier.

GTD009E task_main: no receive method for *XXXX* socket *YYYY*

Reason: A call to *mvsselect*() returned with the socket marked *readable*. The socket is currently in a state requiring a call to *read*(), however, no read routine is set up for this GateD task.

XXXX GateD task name.

YYYY Socket identifier.

GTD010E task_main: no connect method for *XXXX* socket *YYYY*

Reason: A call to *mvsselect*() returned with the socket marked *writable*. The socket is currently in a state requiring a call to *connect*(), however, no connect routine is set up for this GateD task.

XXXX GateD task name.

YYYY Socket identifier.

GTD011E task_main: no write method for *XXXX* socket *YYYY*

Reason: A call to *mvsselect*() returned with the socket marked *writable*. The socket is currently in a state requiring a call to *write*(), however, no write routine is set up for this GateD task.

XXXX GateD task name.

YYYY Socket identifier.

GTD012E task_main: no exception method for *XXXX* socket *NNNN*

Reason: A call to *mvsselect*() returned with the socket marked with an exception. The socket is currently in a state requiring a call to an exception handler, however, no exception handler is set up for this GateD task.

XXXX GateD task name.

YYYY Socket identifier.

GTD013I task_reconfigure re-initializing from *XXXX*

Reason: GateD was signalled to reinitialize from configuration file *XXXX*. This should normally not occur, as the signaling of GateD for reinitialization is not supported.

Action: None. This is an Informational message.

GTD014I task_reconfigure reinitializing done

Reason: Reinitialization of GateD has completed.

Action: None. This is an Informational message.

GTD015E task_reconfigure *NNNN* of *MMMM* adv_entry elements not freed

Reason: An error was encountered freeing the *adv_entry* elements during reconfiguration. This may result in storage fragmentation since all *adv_entrys* could not be released.

NNNN Count of *adv_entries* not freed.

MMMM Total number of *adv_entries.*

GTD016N task_terminate: *XXXX* terminate signal received

Reason: GateD was signalled to terminate. *XXXX* is the name of the signal that was received and started the termination process.

GTD017E task_receive_signal: can not toggle tracing to console

Reason: SIGUSR1 signal was issued but no trace has been allocated. The message is written to the console.

GTD018E task_receive_signal: Ignoring unknown signal SIG*XXXX*(*YYYY*)

Reason: The GateD signal processing subroutine, *task_receive_signal*, received a signal for which it was not configured to process.

SIGXXXX The signal name.

YYYY The signal number.

GTD019E task_close: close: *XXXX*.*YYYY*: *ZZZZ*

Reason: An attempt to close a socket by calling the socket function *close*() failed due to the reason listed in the *errno* Reason.

XXXX GateD task name.

YYYY Socket identifier.

ZZZZ Reason of the *errno* value set by *close*().

GTD020E : task_addr_local: task *XXXX* address *YYYY ZZZZ*

Reason: A call to the socket function *bind*() failed due to the reason listed in the *errno* Reason.

XXXX GateD task name.

YYYY Local address.

ZZZZ Reason of the *errno* value set by *bind*().

GTD021E : task_set_option: task *XXXX* socket *YYYY* option *OOOO*(*NNNN*) : *ZZZZ*

Reason: A call to the socket function *setsockopt*() failed due to the reason listed in the *errno* Reason.

XXXX GateD task name.

YYYY Socket identifier.

OOOO Option name.

NNNN Option number.

ZZZZ Reason of the *errno* value set by *setsockopt*().

GTD022E task_mem_malloc: Can not malloc(*NNNN*) for *YYYY*

Reason: The memory allocation subroutine, *task_mem_malloc*, was called to allocate storage but the call to *malloc*() failed. This is probably due to insufficient space in the product address space. Check WTPLOG13 for virtual storage usage by the address space and increase region size, if necessary.

NNNN Number of bytes requested.

YYYY GateD task name.

GTD023E task_mem_calloc: Can not calloc(*NNNN*, *SSSS*) for *XXXX*

Reason: The memory allocation subroutine, *task_mem_calloc*, was called to allocate storage but the call to *calloc*() failed. This is probably due to insufficient space in the product address space. Check WTPLOG13 for virtual storage usage by the address space and increase region size, if necessary.

NNNN Number of bytes requested.

YYYY GateD task name.

GTD024E task_mem_realloc: Can not realloc (,*NNNN*) for *XXXX*

Reason: The memory allocation subroutine, *task_mem_realloc*, was called to reallocate storage but the call to *realloc*() failed. This is probably due to insufficient space in the product address space. Check WTPLOG13 for virtual storage usage by the address space and increase the region size, if necessary.

NNNN Amount of storage being reallocated.

YYYY GateD task name.

GTD025E task_block_sbrk: *YYYY*

Reason: The memory allocation subroutine, *task_block_sbrk*, was called to allocate a storage block but the call to *malloc*() failed. This is most likely due to insufficient space left in the product address space. Check WTPLOG13 for a report of virtual storage usage by the address space and increase the region size, if necessary.

GTD026E task_get_socket: domain *DDDD* type SOCK_*TTTT* protocol *PPPP YYYY*

Reason: A call to the socket function *socket*() failed due to the reason listed in the *errno* Reason.

DDDD	Address family domain.
TTTT	Socket type.
PPPP	Socket protocol.
YYYY	Reason of the *errno* value set by *socket*().

GTD027E task_get_addr_remote: getpeername(*XXXX*): *YYYY*

Reason: A call to the socket function *getpeername*() failed due to the reason listed in the *errno* Reason.

XXXX	GateD task name.
YYYY	Reason of the *errno* value returned by *getpeername*().

GTD028E task_get_addr_remote: getsockname(*XXXX*): *YYYY*

Reason: A call to the socket function *getsockname*() failed due to the reason listed in the *errno* Reason.

XXXX	GateD task name.
YYYY	Reason of the *errno* value returned by *getsockname*().

GTD029E main: gethostname: *YYYY*

Reason: A call made to the socket function *gethostname*() returned an error with *errno* set. If *YYYY* is HOST_NOT_FOUND, GateD issued a *gethostname*() socket call that failed. This is most likely because the Domain Name Resolver is inactive, inoperable, or configured incorrectly.

Action: Check that hostname is set and that it is not more than MAXHOSTNAMELENGTH (64). If *YYYY* is HOST_NOT_FOUND, verify that the Domain Name Resolver is configured correctly and is operating properly. Issue a DNRGET HOST BYNAME hostname command to determine if the value is being passed properly. Refer to the Customization Guide for information about DNRGET.

YYYY Reason of the *errno* value returned by *gethostname*().

GTD030E : task_connect: task *XXXX* address *YYYY ZZZZ*

Reason: A call to the socket function *connect*() failed due to the reason listed in the *errno* Reason.

XXXX GateD task name.

YYYY Remote address.

ZZZZ Reason of the *errno* value set by *connect*().

GTD040I Tracing to "*SSSS*" suspended

Reason: GateD tracing to the file identified by *SSSS* has been suspended.

SSSS Data set name or *ddname.*

Action: None. This is an Informational message.

GTD041E Could not open "*SSSS*": *YYYY*

Reason: A call to the *fopen*() function to open the trace data set failed due to the reason listed in the *errno* Reason.

SSSS Data set name or *ddname.*

YYYY Reason of the *errno* value set by fopen().

GTD042E trace_on: can not open "*SSSS*" for writing: *YYYY*

Reason: A call to the *fopen*() function to open the trace data set failed due to the reason listed in the *errno* Reason.

SSSS	Data set name or *ddname.*
YYYY	Reason of the *errno* value set by fopen().

GTD043I tracing to "*SSSS*" started

Reason: GateD has started tracing to the data set identified by *SSSS*.

Action: None. This is an Informational message.

GTD044I trace_rotate: rotating "*SSSS*"

Reason: The trace subroutine to switch between trace data sets, *trace_rotate*, was called to rotate to data set *SSSS*. Rotation of trace data sets is not supported.

Action: None. This is an Informational message.

GTD045E trace_rotate: rename(*MMMM*, *NNNN*): *YYYY*

Reason: GateD was unable to rotate the trace files due to the reason listed in the *errno* Reason. Trace file rotation is not supported.

MMMM	Existing filename.
NNNN	New filename.
YYYY	Reason of the *errno* value set by *rename*().

GTD046E *XXXX*: unknown trace flag: *MMMM*

Reason: The program identified by the GateD task name issued a call to the GateD tracing function and passed to it a tracing flag which was unknown to the tracing subroutine.

XXXX	GateD task name.
MMMM	The unknown trace flag.

GTD047E Could not open "*SSSS*": *YYYY*

Reason: A call to *fopen*() to open the dump data set failed for the reason listed in the *errno* Reason.

SSSS Data set name or *ddname.*

YYYY Reason of the *errno* value set by *fopen*().

GTD048E trace_do_dump: can not open "*SSSS*" for writing:*YYYY*

Reason: A call to *fopen*() to open the dump data set failed for the reason listed in the *errno* Reason.

SSSS Data set name or *ddname.*

YYYY Reason of the *errno* value set by *fopen*().

GTD049I trace_dump: processing dump to *XXXX*

Reason: Informational message indicating that a dump is being written. *XXXX* is the name of the dump file.

Action: None. This is an Informational message.

GTD050I trace_dump: dump completed to *XXXX*

Reason: Informational message indicating that the dump has completed. *XXXX* is the path of the dump file.

Action: None. This is an Informational message.

GTD060W path_rt: OSPF ASE tag 0x*XXXX* has invalid length

Reason: *Aspath* routine *path_rt*() discovered an OSPF ASE that contains an invalid length. The OSPF ASE has the tag *0xXXXX*.

GTD080I GTD080I EVENT CCCC NNNN AAAA -> DDDD SSSS
GTD080I EVENT CCCC *NNNN AAAA/MMMM -> BBBB SSSS*
GTD080I EVENT CCCC *NNNN AAAA SSSS*
GTD080I EVENT CCCC *NNNN AAAA/MMMM SSSS*

Reason: The specified interface has had a change of state.

CCCC identifies the change in the interface status.

SSSS identifies the up to date interface state.

CCCC	Reason of the change to the interface
NNNN	Interface name
AAAA	Interface address
DDDD	Point to point destination
MMMM	Subnet mask
BBBB	Broadcast address
SSSS	Interface state

Action: None. This is an Informational message.

GTD081E if_conf_addaddr: Unknown class for interface *AAAA(XXXX)*

Reason: The network class could not be determined for the interface with the IP address *AAAA* interface address and name *XXXX*.

GTD100W if_rtup: UP route for interface *XXXX AAAA/BBBB*

Reason: The specified interface has become active or has had a change in preference.

XXXX	Interface name
AAAA	Interface address.
BBBB	Interface subnet mask.

GTD101W if_rtdown: DOWN route for interface *XXXX AAAA/BBBB*

Reason: The specified interface is now inactive (down). All routes associated with this interface are now marked down.

XXXX Interface name.

AAAA Interface address.

BBBB Interface subnet mask.

GTD120W inet_init: Invalid martian entry at *XXXX/YYYY*

Reason: The martian entry identified by the destination mask is invalid. This is most likely due to an incorrect martian initialization list in the GTDCFG00 member.

XXXX Martian entry destination.

YYYY Martian entry mask.

GTD140E parse_open: error opening *SSSS: YYYY*

Reason: A call to fopen() to open the configuration data set failed for the reason listed in the errno Reason.

SSSS Data set name or ddname.

YYYY Textual Reason of errno value.

GTD141E parse_open: no file specified

Reason: GateD subroutine parse_open was called to open the configuration data set but no configuration data set name was supplied. Make sure that a valid configuration data set (member) name is specified on the GateD() statement in the TCPCFG00 configuration member.

GTD142E parse_eof: error closing *XXXX: YYYY*

Reason: A call to close() to close the configuration data set failed due to the reason listed in the errno Reason.

XXXX Parse data set name.

YYYY Textual Reason of errno value.

GTD160I Start *XXXX*[*DDDD*] version *MMMM* built *YYYY*

Reason: This is an informational message indicating normal initiation of GateD.

XXXX GateD task name.

DDDD Process ID.

MMMM GateD version.

YYYY errno message.

Action: None. This is an Informational message.

GTD161W main: *WARNING* IP forwarding disabled!

Reason: This is a warning message to inform the user that IP forwarding is disabled. IP forwarding is controlled by the specification of the FWD or NOFWD keywords on the IP statement within the TCPCFG00 configuration member.

GTD162W main: *WARNING* UDP checksums disabled in kernel

Reason: This is a warning message to inform the user that checksums usage has been disabled. Checksums are always calculated when sending UDP datagrams, forcing the UDP checksum flag to on, hence this message should never occur.

GTD163N No config file, one interface and a default route, GateD exiting

Reason: GateD requires a configuration data set to properly initiate. A configuration data set was not specified or could not be opened. GateD cannot initialize completely and is exiting.

GTD180E parse_parse: *DDDD* parse error(s)

Reason: There were one or more (*DDDD*) parsing errors encountered during the parsing of the configuration data set (member).

GTD181E *XXXX*: duplicate switch: *SSSS*

Reason: While parsing the invocation parameters for GateD, the parsing task XXXX encountered the switch *SSSS* more than once. The switches passed to the GateD subroutine gatedsub are done programmatically and cannot be changed by user intervention.

GTD182E *XXXX*: missing argument for switch: *SSSS*

Reason: While parsing the invocation parameters for GateD, the parsing task XXXX encountered the switch *SSSS* without arguments required for that particular switch. The switches passed to the GateD subroutine gatedsub are done programmatically and cannot be changed by user intervention.

GTD183E *XXXX*: invalid switch: *SSSS*

Reason: While parsing the invocation parameters for GateD, the parsing task *XXXX* encountered the switch *SSSS*. This switch is invalid. The switches passed to the GateD subroutine gatedsub are done programmatically and cannot be changed by user intervention.

GTD184E gatedsub: extraneous information on command line: *SSSS*

Reason: GateD has detected extraneous information, *SSSS*, on the command line. The parameters passed to the GateD subroutine gatedsub are done programmatically and cannot be changed by user intervention.

GTD185E Usage: *XXXX* [-c] [-C] [-n] [-N] [-t[*flags*]] [-f *config_file*] [*trace_file*]

Reason: GateD has encountered a problem parsing the command line parameters and switches. This message is issued to identify the command format required by GateD. However, the parameters are passed to GateD programmatically and cannot be changed by the user, with the exception of the configuration file specified in the GATED() keyword of the IP statement within the TCPCFG*xx* configuration member.

GTD200N rip_recv_response: bad metric (*DDDD*) for net *AAAA* from *RRRR*

Reason: A rip routing update was received from the router specified by the source address that contained a routing update with an invalid metric for the specified destination. Run a packet trace utility such as TCPEEP to verify that the remote router is truly generating incorrect routing updates.

DDDD Metric.

AAAA rtp destination.

RRRR Source address.

GTD201N rip_recv_response: bad mask (*MMMM*) for net *AAAA* from *RRRR*

Reason: A rip routing update was received from the router specified by the source address that contained an invalid network mask for the specified network. Run a packet trace utility such as TCPEEP to verify that the remote router is truly generating incorrect routing updates.

MMMM Network mask.

AAAA Network address.

RRRR Source address.

GTD202N rip_recv_response: bad router (*MMMM*) for net *AAAA* from *RRRR*

Reason: A rip routing update was received from the router specified by the source address that contained an invalid router for the specified network. Run a packet trace utility such as TCPEEP to verify that the remote router is truly generating incorrect routing updates.

MMMM tp router.

AAAA rtp destination.

RRRR Source address.

GTD203E rip_init: UDP checksums *DISABLED* in kernel; RIP disabled

Reason: The RIP routing protocol requires RIP datagrams sent over UDP to be sent with UDP checksums. However, the use of UDP checksums was disabled and RIP automatically disables itself for this reason. This condition should never happen since UDP datagrams are generated with checksums and the UDP checksum flag to set to on.

GTD204E rip_init: no service for route available, using *DDDD*

Reason: A call to the socket function getservbyname() failed to locate the service id for service route, protocol UDP. RIP will use the port identified by *DDDD* for the transmission of datagrams. Make sure that the domain name resolver is active and that the DNRSVC00 configuration member contains an entry with the following:

```
UDP   ROUTE  520  ROUTER ROUTERD   # RIP
```

GTD205E rip_init: is routed or an old copy of GateD running?

Reason: RIP was unable to successfully bind to the RIP routing port (520 by default). RIP execution is terminating for this reason.

GTD220E rnode_delete: inconsistent annotation

Reason: While attempting to delete a routing node entry from the routing radix tree, a consistency check failed due to inconsistent mask values.

GTD221E rnode_delete: couldn't find our annotation

Reason: While attempting to delete a routing node entry from the routing radix tree, the specific entry could not be located within the tree.

GTD222E rnode_delete: Orphaned Mask *MMMM* at *NNNN*

Reason: While attempting to delete a routing node entry from the routing radix tree, an attempt to demote routes below the route being deleted resulted in a mask with an entry that is still in use and is now orphaned from the rest of the tree.

MMMM Mask value.

NNNN Node address.

GTD240I REDIRECT: *TTTT* redirect from *SSSS*: *DDDD/MMMM RRRR: MSG*

Reason: GateD received a redirect from the specified source address for the identified host or network. GateD's reaction to the redirect will be based on the GateD configuration parameters for redirect processing.

TTTT Redirect type (host/network).

SSSS Source address of redirect send.

DDDD Intended destination.

MMMM Destination mask.

RRRR New router.

MSG Possible message explaining GateD's reaction to the redirect.

Action: None. This is an Informational message.

GTD260E GTD260E rt_event_active: fatal state error
GTD260E rt_event_inactive: fatal state error
GTD260E rt_event_preference: fatal state error
GTD260E rt_event_gateway: fatal state error
GTD260E rt_event_unreachable: fatal state error
GTD260E rt_event_change: fatal state error
GTD260E rt_event_bitset: fatal state error
GTD260E rt_event_reset: fatal state error
GTD260E rt_event_initialize: fatal state error

Reason: These messages are issued when one of the above listed routines are called while in an invalid state.

GTD261W rt_change: interface not found for net *AAAA* gateway *XXXX*

Reason: While attempting a routing change, a route to the listed destination was encountered via the listed router. However, GateD could not locate an interface that had associated with it the same network or subnet address as the specified router.

AAAA Routing destination address.

XXXX Router address.

GTD280I target_build: Ignoring source gateway *AAAA* not on attached net

Reason: GateD could not locate an interface which had associated with it the same network or subnet address as the listed source gateway *AAAA*.

Action: None. This is an Informational message.

GTD300I ospf_parse_virt_parse: priority option ignored for virtual link to *AAAA*

Reason: OSPF does not support the setting of a priority for virtual links and is ignoring this setting for the listed virtual link *AAAA*.

Action: None. This is an Informational message.

GTD320E OSPF SENT *AAAA -> BBBB: YYYY*

Reason: OSPF issued a call to task_send_packet() to issue a socket sendto() function to send an OSPF packet to the listed destination. The call failed due to the reason listed in the errno Reason.

AAAA Source address.

BBBB Destination address.

YYYY Reason of the errno value set by sendto().

GTD321W ospf_ifachange: interface *AAAA (SSSS)* configured in two areas

Reason: OSPF detected that the listed interface was configured into more than one area. Change the OSPF initialization parameters to have the interface in only one area if this causes OSPF to operate incorrectly.

AAAA Interface address.

SSSS Interface name.

GTD322W ospf_ifachange: system does not support multicast; ignoring interface *AAAA (SSSS)*

Reason: The transmission or receipt of IP multicast packets is not supported. As a result, OSPF interfaces must be configured with the NONBROADCAST form of the INTERFACE statement, with the exception of point-to-point interfaces, which must be configured with the broadcast form of the INTERFACE statement. Correct configuration member GTDCFG00 and restart GateD.

AAAA Interface address.

SSSS Interface name.

GTD323W ospf_ifachange: ignoring NON-BROADCAST specification for point-to-point interface *AAAA(SSSS)*

Reason: Point-to-point interfaces such as those running the CLAW channel protocol cannot be defined as NONBROADCAST. Alter the GateD configuration member GTDCFG00 such that the specified interface is defined using the broadcast version of the INTERFACE statement.

AAAA Interface address.

SSSS Interface name.

GTD324I ospf_ifachange: priority option ignored for point-to-point interface *AAAA* (*SSSS*)

Reason: Point-to-point interfaces such as those running the CLAW channel protocol cannot be defined with a priority. The priority option is ignored for these interfaces.

AAAA Interface address.

SSSS Interface name.

Action: None. This is an Informational message.

GTD325W ospf_ifachange: router ID changed from *AAAA* to *BBBB*; terminating OSPF

Reason: A mismatch occurred between the OSPF router ID and the interface router ID. OSPF is terminating.

AAAA OSPF router ID.

BBBB Interface router ID.

GTD326W ospf_startup: Router ID not defined

Reason: OSPF could not determine the router ID. The router ID is determined either by default (the first interface encountered by GateD) or by the specification of the routerid host; statement within the GTDCFG00 configuration member.

GTD340E mon_open: can not get socket for OSPF monitor response

Reason: A call to task_get_socket() to establish a new socket for receiving OSPF monitor responses failed. As a result, no session is created and OSPFMON will not operate. If the subsystem identifier for the product address space is something other than ACSS, make sure that OSPFMON is invoked with the -x command line parameter.

GTD341E mon_open: can not connect to *PPPP*: *YYYY*

Reason: A call to the socket function connect() failed due to the reason listed in the errno Reason.

PPPP IP destination.

YYYY Reason of errno value set by connect().

GTD342E mon_open: fdopen: *YYYY*

Reason: A call to fopen() failed due to the reason listed in the errno Reason, *YYYY*. Make sure that the monitor configuration file name was correctly specified on the OSPFMON command line.

GTD360E find_nh_entry: FAILED! LinkStateType *SSSS* LinkStateID *AAAA* AdvRtr *BBBB* NextHop *HHHH*

Reason: An attempt by OSPF to locate the next hop entry for the specified link and advertised router failed.

SSSS	Link state type.
AAAA	Link state identifier.
BBBB	Advertised router.
HHHH	Next hop.

GTD400I krt_flash: Can not install 127/255 gateway 127.0.0.1: reject routes

Reason: The GateD local routing process does not support reject routes.

Action: None. This is an Informational message.

GTD401E kvm_terminate: kvm_close error

Reason: A call to the kvm_close() function failed. This should never happen.

GTD402E krt_init: *MMMM*

Reason: A call to the kvm_openfiles() function failed. This should never happen.

GTD420E krt_ifread: ioctl SIOCGIFCONF: *YYYY*

Reason: A socket ioctl() function failed while attempting to obtain the current interface configuration. The reason for the failure is described by the errno Reason, *YYYY*.

GTD421E krt_ifread: *SSSS*: ioctl SIOCGIFFLAGS: *YYYY*

Reason: GateD issued a socket ioctl() function call to obtain the interface flags for the specified interface. The call failed due to the reason listed in the errno Reason.

SSSS Interface name.

YYYY Reason of the errno value set by ioctl().

GTD422E krt_ifread: *SSSS*: ioctl SIOCGIFMETRIC: *YYYY*

Reason: GateD issued a socket ioctl() function call to obtain the interface metric for the specified interface. The call failed due to the reason listed in the errno Reason.

SSSS Interface name.

YYYY Reason of the errno value set by ioctl().

GTD423E krt_ifread: *SSSS*: ioctl SIOCGIFDSTADDR: *YYYY*

Reason: GateD issued a socket ioctl() function call to obtain the interface destination address for the specified interface. The call failed due to the reason listed in the errno Reason.

SSSS Interface name.

YYYY Reason of the errno value set by ioctl().

GTD424E krt_ifread: *SSSS*: ioctl SIOCGIFNETMASK: *YYYY*

Reason: GateD issued a socket ioctl() function call to obtain the interface network mask for the specified interface. The call failed due to the reason listed in the errno Reason.

SSSS Interface name.

YYYY Reason of the errno value set by ioctl().

GTD425E krt_ifread: *SSSS*: ioctl SIOGIFBRDADDR: *YYYY*

Reason: GateD issued a socket ioctl() function call to obtain the interface broadcast address for the specified interface. The call failed due to the reason listed in the errno Reason.

SSSS Interface name.

YYYY Reason of the errno value set by ioctl().

GTD426E krt_ifread: no broadcast address for *AAAA* (*SSSS*)

Reason: No broadcast address or a zero broadcast address was obtained for the specified interface.

AAAA Interface address.

SSSS Interface name.

GTD440E krt_rtread: rthost and/or rtnet not in namelist

Reason: An attempt to locate the internal routing tables failed for either the host routing table (rthost), the network routing table (rtnet) or both. Existing routes from the kernel cannot be extracted.

GTD441E krt_rtread: reading hashsize: kvm_read error

Reason: An attempt to read (copy) the routing table hash size failed. This error should never occur as the kvm_read() function is implemented via the built-in memcpy() function and always returns a positive return code.

GTD442E krt_rtread: rthashsize not in namelist

Reason: An attempt to locate the routing table hash value succeeded but returned a zero hash table size.

GTD443E krt_rtread: reading hash bucket: kvm_read error

Reason: An attempt to read (copy) the routing table hash bucket (the area containing the routing table chain heads) failed. This error should never occur as the kvm_read() function is implemented via the built-in memcpy() function and always returns a positive return code.

GTD444E krt_rtread: reading mbuf: kvm_read error

Reason: An attempt to read (copy) a routing table entry failed. This error should never occur as the kvm_read() function is implemented via the built-in memcpy() function and always returns a positive return code.

GTD460E krt_symbols: kvm_nlist error

Reason: GateD subroutine kvm_nlist() was called to locate various control block addresses by name. This function, native to certain UNIX implementations, is emulated. One or more of the names in the name list (nlist) could not be identified.

GTD461E krt_symbols: reading kernel version: kvm_read error

Reason: An attempt to read (copy) the product version identifier failed. This error should never occur as the kvm_read() function is implemented via the built-in memcpy() function and always returns a positive return code.

GTD462I krt_symbols: reading IP forwarding enable flag: SSSS

Reason: An attempt to read (copy) the IP forwarding flag failed. This error should never occur as the kvm_read() function is implemented via the built-in memcpy() function and always returns a positive return code.

Action: None. This is an Informational message.

GTD463I krt_symbols: reading UDP checksum enable flag: SSSS

Reason: An attempt to read (copy) the UDP checksum enabled/disabled flag failed. This error should never occur as the kvm_read() function is implemented via the built-in memcpy() function and always returns a positive return code.

Action: None. This is an Informational message.

Chapter 5

LOD*nnn* LOADXL8 Messages

This chapter describes the messages issued by the LOADXL8 program.

LOD*nnn*

LOD800E Usage:LOADXL8 *InputMemberName OutputDatasetName*

Reason: The arguments to this program are the input member name and the output data set name. They must both be present.

Action: Reenter the command, specifying both arguments.

LOD801E Unable to open input file *filename*

Reason: The LOADXL8 program was unable to open the input file. The translate table being processed must be in the same data set as the LOADXL8 program itself.

Action: Verify that the LOADXL8 program is in the same data set as the translate table member to be processed.

Note: The parameter is a member name and not a data set name. The data set must be in the STEPLIB concatenation.

LOD802E Unable to open output file *filename*

Reason: The LOADXL8 program was unable to open the output file.

Action: Verify that the output data set exists and has the proper attributes. It must have RECFM=F, LRECL=256, BLKSIZE=256.

LOD803E Unable to write title to output file *filename*

Reason: The LOADXL8 program was unable to write the title to the output file.

Action: Verify that the output data set exists and that you have the proper authority to write to it and that there is sufficient space for the file.

LOD804E Error processing input file

Reason: The input translate table did not pass validation. Other messages explain the validation failure.

Action: See following messages.

LOD805E Unable to write table to output file *filename*

Reason: The LOADXL8 program was unable to write a table to the output file.

Action: Verify that the output data set exists and that you have the proper authority to write to it and that there is sufficient space for the file.

LOD806I Processing completed normally.

Reason: The LOADXL8 program completed processing normally.

Action: None. This is an Informational message.

LOD807E Input load module is not a translate table

Reason: The proper eyecatcher was not present in the translate table.

Action: Verify that the module specified really is a translate table.

LOD808E Input translate table is not a character table

Reason: The translate table type field did not indicate character translation.

Action: Verify that the module specified really is a translate table.

LOD809E Input translate table is not ASCII 8

Reason: The flag fields in the translate table did not indicate a type of ASCII eight-bit translate table.

Action: Verify that the module specified really is a translate table.

LOD810E Offset to table, *offset* is greater than length, *length*

Reason: The offset field to the translate table is greater than the indicated length of the table.

Action: Verify that the module specified really is a translate table.

LOD811E Offset to title, *offset* is greater than length, *length*

Reason: The offset field to the translate table title is greater than the indicated length of the table.

Action: Verify that the module specified really is a translate table.

LOD812I Processing translate table:*table*, Title:*name*, Date:*date*, Time:*time*

Reason: The name, title, assembly date, and assembly time fields of the translate table are printed.

Action: None. This is an Informational message.

Chapter 6

MAP*nnn* Portmapper Log Messages

This chapter describes messages written to logs by the Portmapper.

MAP*nnn*

MAP000I PORTMAP INITIALIZATION SUCCESSFULLY COMPLETED.

Reason: The MAP task group was successfully started and was able to process the primary configuration member without error.

Action: None. This is an Informational message.

MAP001E PORTMAP INITIALIZATION FAILED - REASON CODE = *dec* X'*hex*'.

dec The return code from MAPXINIT in decimal.

hex The return code from MAPXINIT in hexadecimal. The MAP task group failed to initialize due to at least one error.

These are the valid values for dec X'hex' :

4 X'0004'	Processing of primary configuration member MAPCFG*xx* failed.
8 X'0008'	IGPOL for ISRB failed.
12 X'000C'	SCHED failed.

Action: Correct the problem that caused the task group initialization to fail, then restart the MAP task group.

MAP002E PORTMAP PROCESSING OF CNFG STATEMENTS FAILED. SYSPARM MEMBER 'MAPCFG*xx*' EITHER NOT FOUND OR IS IN ERROR.

Reason: The module MAPICNFG could not locate or could not successfully parse the primary configuration member MAPCFG*xx*. The single parameter character *xx* is the suffix of the primary configuration member name.

Action: Make sure the primary configuration member exists and the control statements have the proper syntax and valid values. This message should be preceded by a message from MAPICNFG (messages MAP010E through MAP015E), which explains the error in more detail.

MAP010E PORTMAP UNKNOWN COMMAND IN CONFIGURATION MEMBER *mem*. COMMAND '*cmd*' IS NOT VALID.

Reason: While parsing the contents of the primary MAP configuration member, an invalid MAP configuration command was encountered.

mem The primary MAP configuration member.

cmd The specified invalid command.

Action: Correct the configuration statements in the MAP configuration member mem, then restart the MAP task group.

MAP011E PORTMAP ONLY ONE *cmd* COMMAND IS ALLOWED IN THE CONFIGURATION MEMBER *mem*.

Reason: While parsing the contents of the primary MAP configuration member, a command was encountered more than once.

cmd The repeated command.

mem The primary MAP configuration member.

Action: Correct the primary MAP configuration member so that it contains only one of each command.

MAP012E PORTMAP REQUIRED KEYWORD *k* WAS NOT SPECIFIED ON THE *cmd* STATEMENT IN CONFIGURATION MEMBER *mem*.

Reason: A required keyword was not specified as an operand in the specified command in the specified configuration member.

k	The required keyword.
cmd	The specified command.
mem	The primary MAP configuration member.

Action: Add the required keyword operand to the appropriate command.

MAP013E PORTMAP PARSING OF INPUT PARAMETERS IN CONFIGURATION MEMBER *mem* FAILED, REASON CODE = *dec* X'*hex*'.

Reason: The IPARSE command could not successfully parse a command in the MAP primary configuration member.

mem	The primary configuration member.
dec	The IPARSE return code in decimal.
hex	The IPARSE return code in hexadecimal.

Action: Verify that each keyword on every MAP configuration statement is spelled correctly and that the format of each command is correct, then restart the MAP task group.

MAP014E PORTMAP CONFIGURATION MEMBER *mem* DID NOT CONTAIN A 'MAP' OPTIONS STATEMENT

Reason: The specified MAP primary configuration member (*mem*) did not contain a statement that started with the MAP command.

Action: Correct the primary configuration member, adding a statement that starts with the MAP command.

MAP015E PORTMAP CONFIGURATION MEMBER *mem* DOES NOT EXIST IN 'SYSPARM' DATASET. MAP INITIALIZATION FAILED.

Reason: The MAP primary configuration member (*mem*) could not be located in the data set concatenation referenced by the SYSPARM DD statement.

Action: Make sure that the MAP primary configuration member does exist in one of the data sets in the SYSPARM DD statement. If it does exist, then verify that the CNFG(*xx*) operand of the START MAP command contains the valid suffix.

MAP017E UNDEFINED KEYWORD(*k*) ON COMMAND(*cmd*) MEMBER(*mem*).

Reason: The TSO parser detected an unidentified keyword in the command statement.

k	Unidentified keyword.
cmd	Specified command.
mem	Configuration member.

Action: Correct the value in the configuration parameters and restart the MAP task group.

MAP018E RETURN CODE (*rc*) RECEIVED FROM CONFIGURATION FILE I/O HANDLER.

Reason: An unexpected error was detected in the configuration file I/O handler. The rc parameter is the error/return code. This message is issued whenever an unexpected return code is received from ACPIPRMS.

ACPIPRMS passes these return codes:

00 OK

04 EOF on member

08 Member not found or unreadable

12 DCB not open

16 Invalid parameter list

20 Assembly buffer too small

Action: Report this error to the system administrator.

MAP020I PORTMAP SHUTDOWN SUCCESSFULLY COMPLETED.

Reason: The MAP task group was able to complete processing of directory requests and stop execution.

Action: None. This is an Informational message.

MAP030I PORTMAP COMPLETED EXECUTION --- RETURN CODE = *dec* X'*hex*'.

Reason: The main port map program exited with the return code specified in this message. The task group has completed execution.

dec The return code from PORTMAP in decimal.

hex The return code from PORTMAP in hexadecimal.

Action: If the return code is zero, no action is required. If the return code is not zero, check the SYSPRINT output data set for a message (MAP100I through MAP299), which explains why the MAP task group ended with a nonzero return code.

MAP031E PORTMAP AN ABEND WAS DETECTED DURING PORT MAPPER PROCESSING. PORT MAPPER TASK GROUP TERMINATING.

Reason: An ABEND was detected during the processing of the port mapper. All port mapper processing is terminated. Registered services are not unregistered unless the job is stopped. An IFS formatted dump should accompany this message. If the execution JCL included a SYSUDUMP DD statement, an MVS formatted dump should accompany this message.

Action: Contact Customer Support. Have the following information ready for them:

- The IFS formatted dump and the SYSUDUMP, if available
- If SYSLOG message IEA995I indicates that the active load module is PORTMAP, query SMP/E for the PTF level of PORTMAP before calling for support.

Restart the task group to resume port map processing.

MAP032E PORTMAP LOAD FAILED FOR LMOD DNRSCALL. PORT MAPPER TASK GROUP TERMINATING.

Reason: An ILOAD instruction failed to load module DNRSCALL. This program must exist in the SNSLOAD data set. The SNSLOAD data set must be in either the STEPLIB DD concatenation or in the MVS link list.

Action: Correct the problem that is causing module DNRSCALL not to load, then restart the MAP task group.

Sysprint MAP Messages

This section describes messages that are issued to the SYSPRINT data set.

These messages have the following format:

`MAP`*`xxxy timestamp text`*

MAP — Identifies port mapper message.

xxx — Message number, which may be one of these:
100 – 129 are Informational.
130 – 149 are debugging.
150 – 169 are warning.
170 – 199 are port mapper specific errors.
200 - 299 are RPC library errors.

y — A one-character indication of one of these valid message types:
I Informational message.
D Debugging message.
W Warning message.
E Error messages.

timestamp — Date and time in the form shown in this example:

(ExampleWide)Sat Oct 23 21:33:45 1993

text — Message specific text.

Messages all go to SYSPRINT unless processed by perror(). If processed by perror(), messages go to SYSTERM and have additional text appended to the end. All messages that may be processed by perror() are noted.

MAP100I *timestamp* HAS STARTED SUCCESSFULLY

Reason: The port mapper has started operation for the first time of this run of the API.

Action: None. This is an Informational message.

MAP101I *timestamp* has restarted successfully

Reason: The port mapper restarted operation after either an outage by the API or because the port mapper was stopped by an operator command and then started again by an operator command.

Action: None. This is an Informational message.

MAP102I *timestamp* EXIT COMMAND RECEIVED. PORTMAP TERMINATING

Reason: The port mapper received a command from the operator to stop processing. Portmap services are no longer available. Services currently registered are maintained if the port mapper is restarted before the API is terminated.

Action: None. This is an Informational message.

MAP103I *timestamp* SNS/API HAS ENDED

Reason: The port mapper detected that the API has stopped processing. The port mapper periodically attempts to reestablish service with the API when it is restarted.

Action: None. This is an Informational message.

MAP104I *timestamp* STARTING OPERATION

Reason: The port mapper is attempting to establish service with the API.

Action: None. This is an Informational message.

MAP105I *timestamp* SNS/API SUBSYSTEM ID IS *s*

Reason: The port mapper attempts to use the copy of the API with the subsystem ID specified. The single parameter, *s*, is the subsystem ID of the API that the port mapper is using.

Action: None. This is an Informational message.

MAP130D *timestamp* DEBUGGING MODE IS ENABLED

Reason: This message indicates that the DEBUG PARM was selected when the port mapper was started. When DEBUG mode is used, the port mapper issues messages for most events that occur involving the port mapper. Normally only significant events cause messages to be generated.

MAP131D *timestamp* NULL PROCEDURE CALLED

Reason: An RPC client contacted the port mapper's NULL procedure.

MAP132D *timestamp* REGISTER REQUEST RECEIVED

Reason: An RPC client contacted the port mapper's REGISTER procedure.

MAP133D *timestamp* PROGRAM 0xxxxxxxx VERSION *nnn*

Reason: An RPC register service request was received by the port mapper for the RPC program number and version listed.

xxxxxxxx	Hexadecimal value of RPC program number.
nnn	Decimal value of RPC program version.

MAP134D *timestamp* PORT n PROTOCOL *protocol*

Reason: An RPC register service request was received by the port mapper for the port and protocol listed.

n	Decimal value of UDP or TCP port.
protocol	A string identifying the protocol: either UDP or TCP.

MAP135D *timestamp* REGISTER REQUEST SUCCEEDED

Reason: An RPC service register request completed successfully.

MAP136D *timestamp* UNREGISTER REQUEST RECEIVED

Reason: An RPC client contacted the port mapper's UNREGISTER procedure.

MAP137D *timestamp* PROGRAM 0xxxxxxxx VERSION *nnn*

Reason: An RPC unregister service request was received by the port mapper for the RPC program number and version listed.

xxxxxxxx	Hexadecimal value of RPC program number.
nnn	Decimal value of RPC program version.

MAP138D *timestamp* UNREGISTER REQUEST SUCCEEDED

Reason: An RPC service unregister request completed successfully.

MAP139D *timestamp* GET PORT REQUEST RECEIVED

Reason: An RPC client contacted the port mapper's GET PORT procedure.

MAP140D *timestamp* PROGRAM 0xxxxxxxx VERSION *nnn*

Reason: An RPC client requested the port of the RPC program with the number and version listed.

xxxxxxxx	Hexadecimal value of RPC program number.
nnn	Decimal value of RPC program version.

MAP141D *timestamp* PROTOCOL *protocol*

Reason: An RPC client requested the port of the RPC program with the protocol listed. The single parameter, *protocol*, is a string identifying the protocol: either UDP or TCP.

MAP142D *timestamp* SERVICE NOT REGISTERED ON PORT

Reason: The requested RPC program is not registered with the port mapper.

MAP143D *timestamp* SERVICE REGISTERED ON PORT *nnnnn*

Reason: The requested RPC program is registered on the port listed. The single parameter, *nnnnn*, is returned. *nnnnn* is the UDP or TCP port number in decimal.

MAP144D *timestamp* DUMP MAPS PROCEDURE REQUEST RECEIVED

Reason: An RPC client contacted the port mapper's DUMP PORT MAPS procedure.

MAP145D *timestamp* CALL BROADCAST PROCEDURE REQUESTED

Reason: An RPC client contacted the port mapper's BROADCAST CALL procedure.

MAP150W *timestamp* 1API IS NOT YET OPERATIONAL.

Reason: The port mapper could not establish a session with the API at this time. It periodically retries this operation.

MAP151W *timestamp* WAITING FOR *product* API TO BECOME OPERATIONAL

Reason: The port mapper periodically tries to re-establish service with the API. To avoid flooding the output queue with the same message indicating that the port mapper is still trying, it prints this message only after a predetermined number of retries.

MAP152W *timestamp* REGISTER REQUEST FAILED

Reason: An RPC service REGISTER request failed. This is normally due to the service already being registered.

MAP153W *timestamp* UNREGISTER PROCEDURE PASSED BAD ARGUMENTS

Reason: The port mapper UNREGISTER procedure was contacted by a client but the arguments passed by the client to the port mapper could not be understood.

MAP154W *timestamp* REGISTER PROCEDURE PASSED BAD ARGUMENTS

Reason: The port mapper REGISTER procedure was contacted by a client but the arguments passed by the client to the port mapper could not be understood.

MAP155W *timestamp* UNREGISTER REQUEST FAILED

Reason: An RPC service UNREGISTER request failed. This is normally due to the service not being registered.

MAP156W *timestamp* GET PORT PROCEDURE PASSED BAD ARGUMENTS

Reason: The port mapper GET PORT procedure was contacted by a client but the arguments passed by the client to the port mapper could not be understood.

MAP157W *timestamp* DUMP MAPS PROCEDURE PASSED BAD ARGUMENTS

Reason: The port mapper DUMP PORTMAPS procedure was contacted by a client but the arguments passed by the client to the port mapper could not be understood.

MAP158W *timestamp* CALL PROCEDURE NOT CURRENTLY SUPPORTED

Reason: An RPC client attempted to communicate with the port mapper's nonexistent BROADCAST CALL procedure. This procedure is not currently implemented.

MAP159W *timestamp* UNSUPPORTED PROCEDURE REQUESTED

Reason: An RPC client attempted to communicate with the port mapper's nonexistent BROADCAST CALL procedure. This procedure is not currently implemented.

MAP170E *timestamp* CANNOT BIND TO UDP PORT 111

Reason: The port mapper could not bind a name to UDP port 111. This indicates a problem with either the port mapper or the API. This is a fatal error for the port mapper that now exits.

Action: Recycle both the port mapper and the API to see if the problem clears. This message is followed by a perror() message generated by the socket library. See "perror Messages" in *Unprefixed Messages and Codes* for more information.

MAP171E *timestamp* COULD NOT CREATE UDP SERVICE

Reason: The port mapper could not create the necessary control blocks to provide port mapper service with its UDP endpoint. This is a fatal error for the port mapper that now exits.

MAP172E *timestamp* COULD NOT ALLOCATE STORAGE FOR PMAPLIST

Reason: The port mapper could not allocate storage necessary to register a service. This is a fatal error for the port mapper that now exits.

MAP173E *timestamp* CANNOT BIND TO TCP PORT 111

Reason: The port mapper could not bind a name to UDP port 111. This indicates a problem with either the port mapper or the API. This is a fatal error for the port mapper that now exits.

Action: Recycle both the port mapper and the API to see if the problem clears. This message is followed by a perror() message generated by the socket library. See "perror Messages" in *Unprefixed Messages and Codes* for more information.

MAP174E *timestamp* COULD NOT CREATE TCP SERVICE

Reason: The port mapper could not create the necessary control blocks to provide port mapper service with its TCP endpoint. This is a fatal error for the port mapper that now exits.

MAP175E *timestamp* NULL PROCEDURE COULD NOT SEND REPLY

Reason: The port mapper could not reply to a request of its NULL procedure. This is a fatal error for the port mapper that now exits.

MAP177E *timestamp rpc_procedure_name* ERROR

Reason: The port mapper could not respond properly to an RPC request (*rpc_procedure_name*). This is a fatal error for the port mapper that now exits.

MAP178E *timestamp* TERMINATING DUE TO PREVIOUS ERROR

Reason: A fatal error has occurred in the port mapper. A message describing the nature of this error precedes this one. The port mapper terminates operation.

MAP200E *timestamp* (SVCRUN SVCRUN) SELECT FAILED

Reason: An unexpected error occurred while the port mapper was waiting for requests from clients. This error is fatal and the port mapper terminates operation.

MAP2xxE *timestamp (csect.func_name) text*

Reason: This is the general format of error messages generated by the RPC library used by the port mapper. These errors are normally fatal and are followed by a port mapper specific error.

csect The name of the CSECT that encountered the error.

funcname The entry point to a function within the CSECT that specifically encountered the error.

Refer to the *Unicenter TCPaccess Communications Server RPC/XDR ProgrammerReference* for more information.

Chapter 7

T00CF*nnn* Configuration Messages

The configuration messages are the most generic group of messages and can be written by any task group as well as the Trace and IUCV address spaces. Any message that can be written by any IJT-level parsing routines are in this group. These include messages T00CF000 through T00CF999.

T00CF*nnn* Messages

T00CF002E Unknown configuration command - member = *mem_name*, command number = *cmnd_num*, command = *cmnd_name*

Reason: The configuration member contains an invalid statement. Initialization of APP task group or LU pool failed.

mem_name	Name of the configuration member.
cmnd_num	Logical statement number.
cmnd_name	Invalid statement.

Action: Remove the statement command from the configuration parameters and restart the APP task group. If the problem was LU pool related, correct the invalid statement, then issue a REFRESH LUPARM(*mem_name*) command from the console.

T00CF003E Load of module *mod_name* failed

Reason: The configuration member contains an invalid parameter.

Action: Remove the statement command from the configuration parameters and restart the APP task group. If the problem was LU pool related, correct the invalid statement, then issue a REFRESH LUPARM(*mem_name*) command from the console.

T00CF004E Configuration member *mem_name* does not exist in SYSPARM dataset

Reason: During initialization of the LU pool facility, the LU pool configuration member (*mem_name*) was not specified correctly. This message displays when the startup command script does not correctly identify the APPCFG*xx* or APPLUP*xx* configuration files, or the REFRESH LUPARM(*mem_name*) operator command incorrectly specifies the member name.

Action: Correctly specify the APPCFG*xx* and APPLUP*xx* configuration files and reinitialize the APP task group or issue the REFRESH LUPARM(*mem_name*) command with the correct APPLUP*xx* member.

T00CF005E Missing required keyword = *kywd* - member = *mem_name*, command number = *cmnd_num* command = *cmnd_name*.

Reason: A syntax error was encountered with an APPLUP*xx* command statement. The LU pool facility is not activated.

kywd	Keyword in error.
mem_name	Member name.
cmnd_num	Logical number of the statement.
cmnd_name	Statement containing the syntax error.

Action: Correct the statement and issue the REFRESH LUPARM(*mem_name*) command from the operator console.

T00CF006E Mutually exclusive keywords = *kywd*, and kywd2 in member = *mem_name*, command number = *cmnd_num*, command = *cmnd_name*.

Reason: A syntax error was encountered in an LU pool configuration statement. The LU pool facility is not activated.

kywd	Keyword in error.
kywd2	Keyword in error.
mem_name	Member name.
cmnd_num	Logical number of the statement.
cmnd_name	Statement containing the error.

Action: Eliminate the conflicting keyword specification and issue the REFRESH LUPARM(*mem_name*) command from the operator console.

T00CF007W IGNORE host redirect requires IGNORE network redirect; HONOR used

Reason: *Host_option* of IGNORE specified on ICMPREDIRECT keyword of MEDIA statement conflicts with the *network_option*. HONOR will be used for the *host_option*

T00CF009E Invalid parm=parm_value for keyword=keyword, member=mem_name, command number=cmnd_num, command=cmnd_name

Reason: An invalid parameter value was specified for the keyword shown in the message.

Action: Review the documentation and specify a valid value for the keyword.

parm_value	Value specified for the keyword.
keyword	Keyword containing the incorrect parameter value.
mem_name	Configuration member name.
cmnd_num	Logical number of the statement.
cmnd_name	Statement containing the error.

T00CF010E Incompatible applications requested

Reason: The APPLICATIONS keyword of the IFSPARM statement specifies incompatible applications.

Action: Change the list to contain only compatible applications. See the IFSPARM APPLICATIONS documentation for details.

T00CF011E Applications incompatible with previous run of subsystem

Reason: The APPLICATIONS keyword of the IFSPARM statement specifies applications that are incompatible with a previous run of the subsystem since the last IPL.

Action: Change the list to be compatible with previously specified applications. See the IFSPARM APPLICATIONS documentation for details.

T00CF012E Applications incompatible with requested prefix

Reason: The APPLICATIONS keyword of the IFSPARM statement contains applications that are incompatible with the P=/PRFX= parameter specified in the address space JCL.

Action: Remove the P=/PRFX= parameter from the address space JCL.

T00CF013W (NO)INTERNALIUCV is recognized, but deprecated – please use IFSPARM APPS instead

Reason: The INTERNALIUCV/NOINTERNALIUCV keyword will be processed, but note that it may be deleted in a subsequent release of the product.

Action: Add or omit IUCV as necessary in the applications list. (See the IFSPARM APPLICATIONS documentation for details.)

T00CF014E Error in *kywd* keyword of *stmt* statement.

Reason: An improper value was coded in a keyword of a statement. This error message is usually preceded by another message with more details.

kywd Specified keyword.

stmt Specified statement.

Action: Correct the value in the configuration parameters and restart the APP task group.

T00CF015I Invalid character in *kywd* keyword

Reason: The *keyword* entered was not a valid decimal data. It must be between zero and nine, inclusive.

Action: None. This is an Informational message.

T00CF016E Unrecognized value *value* in *keyword kywd*

Reason: A keyword was coded with a value that did not match a list of expected values.

value Coded value.

kywd Specified keyword.

Action: Correct the value in the configuration parameters and restart the APP task group.

T00CF017E SERVICE statement not processed - application application required for module module

Reason: MODULE(module) was specified in a SERVICE statement, but the required application is not available.

Action: Specify application in the IFSPARM APPLICATIONS list, or remove the SERVICE statement from the configuration.

T00CF018E Configuration statement syntax error - member *mem_name*, command *cmnd_name*, record number *rec_num*.

Reason: The TSO parser detected a syntax error while processing a statement. This message is followed by message T00CF038E describing which statement is in error.

mem_name Member name.

cmnd_name Failing command statement.

rec_num Logical record number of the statement containing the error.

Action: See the *Customization Guide* for more information about parsing. Correct the value in the configuration parameters and restart the APP task group.

T00CF019E *value value* is out of range

Reason: The specified value is out of range.

Action: Correct the specified value and retry.

T00CF020I *driver driver* has been added for device *dev_name*

Reason: The named driver has been added for named device.

Action: None. This is an Informational message.

T00CF021E Invalid parm=*parm_value* for keyword=*keyword*, member=*mem_name*, command number=*rec_num*, command=*cmnd_name*

Reason: An invalid parameter value was specified for *keyword*.

Action: Review the documentation and specify a valid value for *keyword*.

parm_value	Value specified with the keyword.
keyword	Keyword containing the incorrect parameter value.
mem_name	Configuration member name.
rec_num	Statement location in configuration member.
cmnd_name	Failing command statement.

T00CF022E Syntax error; Unexpected closing parenthesis - member=*mem_name*, command number=*rec_num*, command=*cmnd_name*

Reason: An unexpected right parenthesis was encountered without a matching left parenthesis.

Action: Correct the syntax with matching parentheses.

mem_name	Configuration member name.
rec_num	Statement location in configuration member.
cmnd_name	Failing command statement.

T00CF023E Syntax error; Too many levels of parenthesis - member=*mem_name*, command number=*rec_num*, command=*cmnd_name*

Reason: Syntax from this statement does not support the level of sub-specifications.

Action: Review statement documentation and correct the syntax.

mem_name	Configuration member name.
rec_num	Statement location in configuration member.
cmnd_name	Failing command statement.

T00CF024E Syntax error: matching parenthesis missing, member=*mem_name*, command number=*rec_num*, command=*cmnd_name*

Reason: A closing parenthesis is missing in the statement syntax.

Action: Correct the syntax with a matching parenthesis.

mem_name	Configuration member name.
rec_num	Statement location in configuration member.
cmnd_name	Failing command statement.

T00CF025E Invalid value for *subparm* subparameter of *statement*'s *kywd* keyword

Reason: Value of *subparm* parameter is invalid or out of range.

Action: Correct the value and restart.

T00CF028E ARM element name *elementname* invalid: Valid characters are A-Z, 0-9, $, #, @, and _ (underscore)

Reason: The requested ARM element name contains invalid characters.

Action: Change the ARM element name so that it contains only valid characters.

T00CF029E ARM element name *elementname* invalid: First character may not be a number

Reason: The requested ARM element name contains a number as its first character.

Note: ARM element names must not start with a number.

Action: Change the ARM element name to start with a non-numeric character.

T00CF030E ARM element name *elementname* reserved: Begins with A-I or SYS

Reason: The requested ARM element name begins with the characters A-I, or the string SYS. These element names are reserved.

Action: Change the ARM element name to begin with a non-reserved prefix.

T00CF032W *Newvalue* overrides *oldvalue* for port *portnumber*

Reason: The *newvalue* keyword is overriding the *oldvalue* keyword for Telnet port *portnumber*.

Action: Eliminate the *oldvalue* keyword from the configuration.

T00CF033W *stmnt kywd* value *value* too large, reduced to *max_value*

Reason: A value was specified that was too large. It was reduced to a lower value.

stmnt	Statement.
kywd	Keyword.
value	Value specified.
max_value	Maximum value allowed.

Action: To avoid a recurrence of this message, correct the value in the configuration parameters and restart the APP task group.

T00CF034W *stmnt kywd* value *value* too small, increased to *min_value*

Reason: A value was specified that was too small. It was increased to higher value.

stmnt	Statement.
kywd	Keyword.
value	Value specified.
min_value	Minimum value allowed.

Action: To avoid a recurrence of this message, correct the value in the configuration parameters and restart the APP task group.

T00CF035E INTERNALIUCV is valid only with APPS(STACK)

Reason: The INTERNALIUCV keyword is valid only if the STACK application will be activated.

Action: Delete the INTERNALIUCV keyword; if IUCV is to be run internally, specify both STACK and IUCV in the applications list. (See the IFSPARM APPLICATIONS documentation for details.)

T00CF036E Unknown keyword found in member(*mem_name*), command(*cmnd_name*), keyword(*kywd*), record number(*rec_num*)

Reason: The TSO parser detected an unknown keyword. This message is followed by message T00CF038E .

mem_name	Name of the configuration member.
cmnd_name	Command statement.
kywd	Unknown keyword.
rec_num	Record number of the configuration statement that contains the error.

Action: Correct the value in the configuration parameters and restart the APP task group.

T00CF037E APPS(IUCV) and NOINTERNALIUCV are incompatible

Reason: APPS(IUCV) and NOINTERNALIUCV indicate conflicting commands about whether to run IUCV in the address space.

Action: Delete the NOINTERNALIUCV keyword; if IUCV is not to be run internally, delete IUCV from the applications list. (See the IFSPARM APPLICATIONS documentation for details.)

T00CF038E Error statement: *stmnt*

Reason: This message always follows message T00CF018E or T00CF036E. It displays the parameter and value (up to 80 characters) of the configuration statement causing the error.

Action: Take corrective action specified in the associated T00CF018E or T00CF036E message.

T00CF039E Return code(*ret_code*) returned from configuration file I/O handler.

Reason: An unexpected error was detected in the configuration file I/O handler. The code is the error/return code.

Action: Report this error to the system administrator.

T00CF046E *stmnt* statement # *mem_rec_num* in member *mem_name* exceeds 64K

Reason: The parameter text in a statement exceeds 65,586 characters. All further configuration statement processing is halted.

stmnt	Specified statement.
mem_rec_num	Configuration member record number.
mem_name	Configuration member.

Action: Correct the value in the configuration parameters and restart the APP task group.

T00CF047E Duplicate IP for PORT *port* on *cmd* statement

Reason: Two *cmd* statements have identical IP address and PORT.

port	Specified PORT.
cmd	Statement containing the error.

Action: APP task group initialization fails. Correct the error and restart the APP task group.

T00CF066W *stmnt kywd* value *class* is invalid; using *oclass*

Reason: Keyword *kywd* value class is invalid. The previous value *oclass* will be used.

T00CF067W *stmnt kywd* invalid subfield *field* - bypassing

Reason: Keyword *kywd* of statement *stmnt* contains invalid subfield of *field*.

T00CF069W Obsolete keyword AUTH ignored - replaced by CA LMP

Reason: The AUTH keyword is obsolete and has been replaced by the Computer Associates License Management Program (CA LMP). CA LMP is a subcomponent of CAIRIM that provides a standardized and automated approach to the tracking of licensed software.

Action: See the CA Common Services for z/OS and OS/390 *Getting Started* guide for additional information on installing and configuring CAIRIM and CA LMP.

T00CF070E CAIRIM is not active

Reason: This software requires the Computer Associates License Management Program (CA LMP) to perform product key verification. CA LMP is a subcomponent of CAIRIM that provides a standardized and automated approach to the tracking of licensed software. However, LMP product key verification cannot be done because CAIRIM is not started, the maintenance level of CAIRIM does not contain LMP support, or the LMP product keys are not correctly defined to CAIRIM.

Action: Start CAIRIM if it is not started. If CAIRIM is started, make sure that it is at the latest maintenance level and that it is properly set up with the CA LMP service. See CA Common Services for z/OS and OS/390 *Getting Started* for additional information on installing and configuring CAIRIM and CA LMP.

T00CF098I component initialization successful; Member=*mem_name*

Reason: The initialization of a component has completed successfully. In the message, *component* is the component title and *mem_name* is the configuration member

Action: None. This is an Informational message.

T00CF099E component initialization failed; Member=*mem_name*

Reason: The initialization of a component has failed. In the message, component is the component title and *mem_name* is the configuration member name from which the component obtained configuration parameters.

Action: Correct the configuration statements in error and reinitialize the component.

T00CF100W Component *component* not recognized in LOGGING statement

Reason: The component specified by component is unknown in the LOGGING statement.

T00IF108I Application activation beginning

Reason: Activation has started for requested applications.

Action: None. This is an Informational message.

T00IF109I Application activation complete

Reason: The requested applications are now active.

Action: None. This is an Informational message.

T00IJ165W ARM services not available on this machine

Reason: No couple data set is active on the machine, so ARM services are not available. (The address space will monitor ENF events for couple data set availability, and begin ARM services if one becomes available.)

Action: If ARM services are required, define a couple data set to the sysplex. Otherwise, no action is necessary.

Chapter

8

T00EX*nnn* Exit Messages

This chapter describes messages issued by Exit processing. These messages are T00EX000 through T00EX999.

T00EX*nnn*

T00EX001D The *exit_point* exit program *prog_name* returned with return code *rc_num* = *ret_code*

Reason: The exit program *prog_name* was called at exit point *exit_point* and returned a non-zero return code *ret_code* in either return code one or return code two, as indicated by *rc_num*. To learn more about User Exits, refer to your product documentation for information about exit return codes.

T00EX002E An abend has occurred in *exit_point* user exit program *prog_name*.

Reason: The exit program *prog_name* was invoked at exit point *exit_point*, and an abend resulted. The exit interface provides recovery and takes an SVC dump. If the exit point is one at which a request (such as a connection request) can be rejected, the request is rejected.

Action: Save the output and the SVC dump. Contact the programmer or vendor of the exit program.

T00EX003W User exit *prog_name* is disabled.

Reason: The user exit program *prog_name* has been disabled at all exit points.

T00EX004I *'prog_name: message_text'*

Reason: The user exit program *prog_name* called the exit interface to log a message *message_text*.

Action: Contact the programmer or vendor of the exit program to determine appropriate action.

T00EX005E Failed to obtain SFRE for *exit_point* user exit program *prog_name*. Exit will not be invoked

Reason: The exit facility could not get a recovery environment in order to call the exit program *prog_name*. The exit will not be invoked.

Action: Check the message log for indications of problems, such as storage shortage, which might be responsible for the failure. If the problem cannot be resolved, save all output and contact Customer Support.

Chapter 9

T00IF*nnn* IFS Messages

This chapter describes the messages issued by the Infrastructure (IFS) program. These include messages T00IF001 through T00IF999.

T00IF*nnn*

T00IF001I Address space initialization beginning

Reason: An IFS-type address space has been started by an MVS operator command.

Action: None. This is an Informational message.

T00IF002I Address space initialization complete

Reason: Basic address space initialization is complete. Initialization includes establishing a long-term enqueue to prevent a second copy of this address space from starting, identifying PC call tables to MVS, identifying an extended ECB exit to MVS, and initializing the job step task.

Action: None. This is an Informational message.

T00IF003E Address space already started (ENQ failed)

Reason: During initialization of the address space, the long-term enqueue failed. This means that another address space with the same definition type must already be active.

Action: Terminate the existing address space, or direct commands to the existing address space if it performs the desired functions.

T00IF004I Task *#task_num* started for *ver.rel*

Reason: A task, in response to a START command, completed initialization.

Action: None. This is an Informational message.

T00IF005I Task *#task_num* stopped for *tg_name*

Reason: A task, in response to a STOP command, was terminated. The parameter *tg_name* indicates the task group name.

Action: None. This is an Informational message.

T00IF006E *uniq prfx itgb tgid tg_name* Task initialization exit failed, task terminating

Reason: An error occurred during task initialization.

uniq	Unique name of the address space where the failure occurred.
prfx	Prefix name of the address space where the failure occurred.
itgb	String ITGB, indicating IFS Task Group Block.
tgid	Task group identifier.
tg_name	Task group name.

Action: Call Customer Support.

T00IF007E *uniq prfx itgb tg_id tg_name* Error processing EQT, attempting recovery

Reason: A task group error occurred while holding the MVS local lock. A functional recovery routine has gained control.

uniq	Unique name of the address space where the failure occurred.
prfx	Prefix name of the address space where the failure occurred.
itgb	String ITGB, indicating IFS Task Group Block.
tg_id	Task group identifier.
tg_name	Task group name.

Action: Call Customer Support.

T00IF008E *uniq prfx itgb tg_id tg_name* Error in wait, attempting recovery

Reason: A task group error occurred during MVS wait. A functional recovery routine has gained control.

uniq	Unique name of the address space where the failure occurred.
prfx	Prefix name of the address space where the failure occurred.
itgb	String ITGB, indicating IFS Task Group Block.
tg_id	Task group identifier.
tg_name	Task group name.

Action: Call Customer Support.

T00IF009E *uniq prfx itgb tg_id tg_name* Error during dispatch, attempting recovery

Reason: A task group error occurred during local dispatching. A functional recovery routine has gained control.

uniq	Unique name of the address space where the failure occurred.
prfx	Prefix name of the address space where the failure occurred.
itgb	String ITGB, indicating IFS Task Group Block.
tg_id	Task group identifier.
tg_name	Task group name.

Action: Call Customer Support.

T00IF010E *uniq prfx itgb tg_id tg_name* Error after post, task terminating

Reason: A task group error occurred after MVS post. A functional recovery routine has gained control.

uniq	Unique name of the address space where the failure occurred.
prfx	Prefix name of the address space where the failure occurred.
itgb	String ITGB, indicating IFS Task Group Block.
tg_id	Task group identifier.
tg_name	Task group name.

Action: Call Customer Support.

T00IF011E *uniq prfx itgb tg_id tg_name* Error ISRB termination

Reason: A task group error occurred during local dispatching. A functional recovery routine has gained control.

uniq	Unique name of the address space where the failure occurred.
prfx	Prefix name of the address space where the failure occurred.
itgb	String ITGB, indicating IFS Task Group Block.
tg_id	Task group identifier.
tg_name	Task group name.

Action: Call Customer Support.

T00IF012E *uniq prfx itgb tg_id tg_name* Error resuming ISRB

Reason: A task group error occurred during resume dispatching. A functional recovery routine has gained control.

uniq	Unique name of the address space where the failure occurred.
prfx	Prefix name of the address space where the failure occurred.
itgb	String ITGB, indicating IFS Task Group Block.
tg_id	Task group identifier.
tg_name	Task group name.

Action: Call Customer Support.

T00IF013I Parse failed for command *cmnd_name* in member *mem_name*, RC=*ret_code*

Reason: TSO parse service routines failed while parsing an operator command. This occurs when the command syntax is confused or is incorrect. Special characters can also cause parse failures.

Another condition that can generate this error occurs when the Subsystem Recognition Character (SRC) is set to the same as that used by JES (JES2 uses $, JES3 uses *).

The variable *ret_code* is a decimal return code from the TSO Parse Service routine. The return code is one of the following:

4 Command parameters were incomplete and parse was unable to prompt.

8 Parse did not complete. An attention interruption occurred during parse processing.

12 Parse parameter block contains invalid information.

16 Parse issued a GETMAIN and no space was available.

20 Validity checking routine requested termination.

24 Conflicting parameters were found.

Action: Try to reenter a correct command or eliminate special characters.

If your SRC is either $ or *, check with your JES systems programmer about what character you can use.

T00IF014I Invalid verb *cmnd_name* for task *tg_id*

Reason: An invalid command verb was directed to the task group. The command was directed to the task group explicitly.

Action: Try to reenter a correct command verb, direct it to another task, or omit the task group from the command.

T00IF015W Task group ID required

Reason: A START or STOP command was issued, but the required task group type was not supplied with the command.

Action: Reissue the command, specifying a three-character task group name with the start command.

T00IF016W Task group ID *tg_id* not found

Reason: A START, STOP, SET, or command was issued, but the requested task group type is invalid or the component is not installed. The variable *tg_id* identifies the task group.

Action: Specify a valid three-character task group name and reissue the command.

T00IF017W Maximum tasks active for group ID *tg_id*

Reason: A START command was issued, but the requested task group already has the maximum number of active tasks permitted. The variable *tg_id* identifies the task group.

Action: Usable subtasks for the requested task group should already be active. Display them with the TASK command. If no tasks exist and none can be started, stop and restart the address space where the problem occurred.

T00IF018W No tasks active or matching for group ID *tg_id*

Reason: A STOP command was issued, but the requested task group type has no active subtasks. The variable *tg_id* identifies the task group.

Action: Correct the command syntax and retry.

T00IF019W Invalid task group ID–-*tg_id*

Reason: A START, STOP, SET or TASK command was issued, but the format of the requested task group specification is incorrect. The variable *tg_id* identifies the task group.

Action: Specify a valid three-character task group name and reissue the command.

T00IF020I Snap complete to class *cls_name*

Reason: An operator requested, or ABEND initiated snap is complete. The variable cls_name, identifies the JES SYSOUT class to which the snap was dynamically allocated.

Action: If the class is serviced by a JES printer, it is printed. If the class is not serviced by a JES printer, an external writer procedure must be initiated to collect the snap output.

T00IF021I Page services, *object_type object_name action* failed reason=*rsn_code*

Reason: A request for page-fix or page-free service failed.

Possible object types are the following:

- MODULE
- CALLER
- STORAGE
- CNTRLBLK
- Blanks

Possible actions are the following:

FIX-LONG—Long-term page-fix

FIX-SHRT—Short-term page-fix

FIX-FAST—Fast page-fix

FREE —Unfix page

FASTFREE—Unfix page using MVS FASTPATH services.

LOAD—Pagein object

OUT—Pageout

RELEASE—Release contents of object

The variable rsn_code is a decimal reason number for the failure; either four (module nonexistent) or 16 (A(object)=0).

Action: None. This is an Informational message.

T00IF022I Storage at *stor_addr* in KEY *key*

Reason: Displays main storage. In the message text, the hexadecimal parameter *stor_addr* is the beginning address of the storage being displayed, and *key* is the storage key.

Action: None. This is an Informational message.

T00IF023I *c_blk* at *stor_addr*

Reason: Displays a control block in main storage. The variable *stor_addr* is the beginning address, in hexadecimal, of the storage displayed.

Action: None. This is an Informational message.

T00IF024I *addr +disp wrd1 wrd2 wrd3 wrd4 *chars16**

Reason: Displays main storage. Shows sixteen bytes per line.

Addr	Virtual address.
Disp	Displacement from the beginning address of the storage displayed, in hexadecimal display form.
Wrd1	Contents of the first word in hexadecimal display form.
Wrd2	Contents of the second word in hexadecimal display form.
Wrd3	Contents of the third word in hexadecimal display form.
Wrd4	Contents of the fourth word in hexadecimal display form.
chars16	Contents of the four words in character display form.

Action: None. This is an Informational message.

T00IF025I Maintenance level for *tg_name ver.rel*

Reason: Displays the component maintenance status in response to a status command.

Action: None. This is an Informational message.

T00IF026I Route code(*rt_code*) now set

Reason: This is a response to a $ROUTE command to either display or vary route codes for a task group. The variable *rt_code* is a string of route code numbers representing the current settings in effect for the task group.

Action: None. This is an Informational message.

T00IF027I Route code value out of range (1-16)

Reason: This is a response to a $ROUTE command vary route codes for a task group. There is a value in the new route code specification outside the acceptable range of values of 1 to 16.

Action: The request is ignored. Reenter the request, if desired, with a set of correct route code values.

T00IF028I Console ID *cons_id*, console area *area_id* set

Reason: This is a response to a $CON command to either display or vary Console Routing for a task group. The parameter *cons_id* is a hexadecimal word representing the current task console routing.

IF the first half of the word is...	Then the...
nonzero	Routing is to a TSO terminal.
zero	Second half word represents a one byte console ID and a one byte area ID for that console.

area_id corresponds to the WTO AREAID option and forces WTO descriptor codes to eight and nine.

Action: None. This is an Informational message.

T00IF029E Unable to load ICCD handler

Reason: The ICCD handler program could not be loaded. The address space terminates.

Action: Verify that the ICCD handler program (T00SICCD) is in your STEPLIB concatenation, and that you have sufficient private storage above the 16MB line. If the problem persists, contact Computer Associates.

T00IF030E Unable to initialize ICCD

Reason: The ICCD could not be initialized. The address space terminates.

Action: Verify that you have sufficient SQA storage (1 page) below the 16MB line. If the problem persists, note the message preceding this one, contact Computer Associates.

T00IF031E Unable to initialize module tables

Reason: The IFS module tables could not be initialized. The address space terminates.

Action: Verify that all IFS programs are in your STEPLIB concatenation. If the problem persists, contact Computer Associates.

T00IF032I Startup type is *startup_type*

Reason — The address space startup type is *startup_type* (COLD after an IPL or an immediately-previous P CLEAR stop; WARM otherwise).

T00IF033E Application for *task_group* task group not available

Reason — Task group *task_group* could not be started, because the required application is unavailable.

Action — Add the required application to the list in the IFSPARM APPS configuration statement.

T00IF034E Module *mod_name* ILATCH *function_name* failure for latch *latch_name*, latch addr=*ltach_addr*, rtn code=*ret_code*

Reason: — IILATCH call for function function_name for latch *latch_name* failed with a return code of *ret_code*. The latch address is at location *latch_addr*.

Action: — This problem should not occur during normal operation. Look for related messages in the logs. Dump the address space via the operator console and contact Customer Support.

Issue the "/f *jobname*,ilatch free" command to free any held latches.

T00IF035E TCP unable to allocate a timer element.

Reason: — TCP was unable to allocate a timer element for a new connection. Either the maximum number of connections are active or there is no virtual storage for the timer element.

Action: — Increase the virtual storage available to the address space or retry the connection when there are fewer other connections.

T00IF038W The following latch has been held over *number* seconds:

Reason: — A latch has been found that has been held for too long. *number* is the number of seconds used to determine if the latch has been held too long.

Action: — Use the ILATCH DISPLAY command to determine who is holding the latch. Consider terminating that user; consider using the ILATCH FREE command to release the latch.

T00IF039I Freeing the following Latch:

Reason: A latch is being freed by the ILATCH command processor. Subsequent messages display information about the latch being freed.

Action: None. This is an Informational message.

T00IF040E Unable to Free the previous latch: (RC = *ret_code*).

Reason: The ILATCH command processor was unable to free a latch as requested. *ret_code* is the return code from the ILATCH FREE call.

Action: Contact Customer Support and tell them the return code from the message.

T00IF041I End of ILATCH command.

Reason: This is the end of processing an ILATCH command.

Action: None. This is an Informational message.

T00IF042E No latch work area allocated.

Reason: The ILATCH command processor was unable to locate the Latch Work Area and is unable to process the command. The Latch Work Area should be allocated as part of initialization.

Action: Contact Customer Support.

T00IF043E No level 1 latch map allocated.

Reason: The ILATCH command processor was unable to locate the Level 1 Latch Map, and is unable to process the command. The Level 1 Latch Map should be allocated as part of initialization.

Action: Contact Customer Support.

T00IF044E No level 2 latch map allocated.

Reason: The ILATCH command processor was unable to locate the Level 2 Latch Map and is unable to process the command. The Level 2 Latch Map should be allocated as part of initialization.

Action: Contact Customer Support.

T00IF045I ILATCH display

Reason: A latch is being displayed. This message will be followed by messages T00IF046I and T00IF047I for exclusive latches, or T00IF057I and T00IF058I for shared latches.

Action: None. This is an Informational message.

T00IF046I Latch# Addr Asid ASCB ITCB RB/Token Jobname Time

Reason: This is the header information for message T00IF047I.

Action: None. This is an Informational message.

T00IF047I *latchnum latchaddr addr_id ASCBaddr TCBaddr RBaddr jobname time*

Reason: This message is the result of processing an ILATCH command that displays exclusive latches. The latch number specified in the command either was not currently held, or was held less than the specified (or default) time value.

Action: Enter the **ILATCH DISPLAY** command to determine if the latch number is still held, or the amount of time it has been held. Then re-enter the ILATCH command as desired.

latchnum	Latch number.
latchaddr	Latch block address.
addr_id	Address space ID of the holder of the latch.
ASCBaddr	Address of the ASCB of the holder of the latch.
TCBaddr	TCB address or the SRB address.
rbaddr	RB address or the first 8 bytes of the SRB token.
jobname	As displayed by SDSF DA command.
Time	Number of seconds the latch has been held.

T00IF048W Requested latch not found.

Reason: The requested latch is not being held or has not been held as long as the TIME operand of the ILATCH command specifies.

Action: Specify a different TIME keyword or a different latch number. Or it is possible the specified latch is no longer held.

T00IF049E Load failed for routine *routine_name*, abend code=*abend_code*, reason code=*reason_code*

Reason: The routine *routine_name* failed with the abend code specified.

Action: Notify Customer Support.

T00IF050E *Abend* S*sabnd*/U*uabnd* in module *mod_id mod_size*

Reason: A standard IFS recovery routine has gained control from a module ABEND situation and is displaying the ABEND situation.

sabnd	MVS system ABEND code (hexadecimal).
uabnd	Application user ABEND code (decimal).
mod_id	Module identification, including CSECT name, assembly time and date, and version and level of the module.
mod_size	Size of the module in bytes (hexadecimal).

Action: Notify Customer Support.

T00IF051E Psw=*psw1 psw2* Intc=*icde* Instr=*idata* Displ=*disp*

Reason: The PSW, interrupt code, failing instruction, and displacement of the failure in a module (if possible) are formatted. See message T00IF050E.

psw1	First word, in hexadecimal, of the Program Status Word at time of failure
psw2	Second word of the Program Status Word at time of failure. This word contains the address of the failing instruction (hexadecimal).
icde	Hexadecimal interrupt code causing the failure
idata	Failing instruction (hexadecimal)
disp	Displacement of the failing instruction in the abending module, in hexadecimal

Action: Notify Customer Support.

T00IF052W Tg: *tg_id tg_name itgb itcb flg1 flg2 tcb csrb*

Reason: This message contains information formatted for the task group associated with the ABEND. See message T00IF050E .

tg_id	Task group ID.
tg_name	Task group name.
itgb	Hexadecimal address of the failing IFS Task Group Block.
itcb	Hexadecimal address of the failing IFS Task Control Block.
flg1	Hexadecimal value of flag1 in the IFS Task Control Block.
flg2	Hexadecimal value of flag2 in the IFS Task Control Block.
tcb	Hexadecimal address of the failing MVS Task Control Block.
csrb	Hexadecimal address of the associated IFS Command SRB.

Action: Notify Customer Support.

T00IF053W Regs *r1-r2 reg1 reg2 reg3 reg4*

Reason: This message contains formatted register information associated with the ABEND. This line appears four times, each time with four of the sixteen hexadecimal registers. See message T00IF050E .

r1	First register shown on the line, either 0, 4, 8, or C.
r2	Last register shown on the line, either 3, 7, 9, or F.
reg1	Value of the first register in the group, 0, 4, 8, or 12.
reg2	Value of the second register in the group, 1, 5, 9, or 13.
reg3	Value of the third register in the group, 2, 6, 10, or 14.
reg4	Value of the fourth register in the group, 3, 7, 11, or 15.

Action: Notify Customer Support.

T00IF054W Sdwafgls = *erra errb errc errd*

Reason: This message contains formatted information from the MVS System Diagnostic Work Area associated with the ABEND. See message T00IF050E .

erra Hexadecimal value of the byte known as SDWAERRA.

errb Hexadecimal value of the byte known as SDWAERRB.

errc Hexadecimal value of the byte known as SDWAERRC.

errd Hexadecimal value of the byte known as SDWAERRD.

Action: Notify Customer Support.

T00IF055I VTAM interface installed, A(VCVT)=*vcvt*

Reason: This message is in response to a VTAM command, showing the address of the VTAM Communication Vector Table built. The single parameter, *vcvt*, is the hexadecimal address of the VTAM CVT.

Action: None. This is an Informational message.

T00IF056I VTAM interface already installed, A(VCVT)=*vcvt*

Reason: This message is in response to a VTAM command, showing the address of the VTAM Communication Vector Table that has already been installed by a previous VTAM command. The single parameter, *vcvt*, is the address of the VTAM CVT.

Action: None. This is an Informational message.

T00IF057I Latch# Addr Shr_cnt Time

Reason: Header information for message T00IF058I.

Action: None. This is an Informational message.

T00IF058I *latch_num latch_addr Shr_cnt time_held*

Reason: This message is the result of processing an ILATCH command that displays shared latches. The latch number specified in the command either was not currently held, or was held less than the specified (or default) time value.

Enter the ILATCH DISPLAY command to determine if the latch number is still held, or the amount of time it has been held. Then re-enter the ILATCH command as desired.

latch_num	Latch number.
latch_addr	Latch block address.
Shr_cnt	Shared count.
time_held	Number of seconds the latch has been held.

T00IF060E Unknown abend parameter specification (*abend_code*)

Reason: There was a syntax error on the ABEND command. The allowable values are S0C1-S0CF, and S806.

Action: Correct the syntax and retry.

T00IF062I Log started

Reason: A date stamp of the start of the log file.

Action: None. This is an Informational message.

T00IF063E Address space terminating in error

Reason: A fatal error has been recognized during initialization of the IFS address space. The address space will be terminated.

Action: This message will be accompanied by other messages identifying the reason for the termination. Examine the accompanying messages to determine the appropriate corrective action.

T00IF064E Incompatible releases using same subsystem name

Reason: IFS initialization found a subsystem environment from an earlier run of the product, but the software release level of that run does not match the current run. When using the same subsystem name for different releases of the product, it is necessary to shut down with a P CLEAR command to clean up the subsystem environment before starting the new release.

Action: Either restart with a different subsystem name, or bring up the prior release, stop it with a P CLEAR command, and then start the new release again.

T00IF065E Restart prior running release; issue P CLEAR. Restart current release (xxxx)

Reason: Messages T00IF063E, T00IF064E, and T00IF065E always display together.

An incompatibly was recognized between a system startup stage and residual components from the previous running system. The residual components are still resident because the previous termination did not specify a P,CLEAR option.

Action: Reactivate the previous version of TCP/IP, terminate it with the CLEAR option, then restart the newer version.

T00IF066I Virtual storage statistics: *vvv* bytes available *above/below* the line, *uu*% used, *mm*% min, *xx*% max

Reason: This message displays an estimate of the used and available virtual storage, first for the below-the-line area, then for above-the-line. The numbers are extracted from the VSMLIST statistics as reported by MVS. The available byte values are based on subtraction from the region size.

Message display and frequency is governed by the VSREPORT keyword of the IFSPARM statement in the IJTCFG*xx* member. Statistics are recalculated with each display to insure accuracy.

vvv	Current amount of unallocated virtual storage, in bytes.
above/below	Either ABOVE or BELOW the 16MB line.
uu	Current amount used, expressed as a percentage of the region size.
mm	Minimum amount used, expressed as a percentage of the region size.
xx	Maximum amount used, expressed as a percentage of the region size.

Action: None. This is an Informational message.

T00IF067I Region size is *value1*, *value2* below the line

Reason: Both region sizes are displayed for the job as byte counts.

The region size specifies the maximum amount of storage that can be allocated to the job by any single variable-length GETMAIN request. Depending on how the region size was requested, it may include the region above 16MB. The below-the-line value specifies the maximum total storage that can be allocated to below-the-line requests.

These values are set by the IBM-supplied or installation-written exit routine. The region size usually is set equal to the REGION parameter specified by the product JCL (or installation default if not specified), and the region size limit is usually equal to the region size plus 64 KB.

value1 The region size registered.

value2 The size of the region below the 16MB line.

Action: None. This is an Informational message.

T00IF068I Date is *date*

Reason: This message displays during a message logging procedure when a date change is detected. It identifies subsequent log entries with their appropriate logging date.

Note: With the exception of T00IF071E, all messages between T00IF070D and T00IF088D display only when security monitoring is activated via the XSEC statement in the IJTCFGxx member. The security functions to be monitored are specified as keyword selections on the XSEC statement, and the appropriate messages display at runtime.

Some functions occur frequently and monitoring will contribute to the volume of the message log.

Action: None. This is an Informational message.

T00IF070D Entering ACSECPC routine

Reason: The security handling module is invoked.

T00IF071E Bad parms passed to ACSECPC

Reason: This is an Internal error. The TCP/IP system will abend.

Action: Call Customer Support

T00IF072D Processing a sign-on

Reason: This message displays when a user ID attempts to log on.

T00IF073D Processing a sign-off

Reason: This message displays when a user ID logs off.

T00IF074D Processing a dataset check

Reason: This message displays when a data set name is authorized.

T00IF075D Placing an ACEE on the TCB

Reason: This message displays when the ACEE monitoring option is selected.

T00IF076D Removing an ACEE from the TCB

Reason: This message displays when the ACEE monitoring option is selected.

T00IF077D RACLIST directory create function in progress

Reason: This message displays at startup when the private security directory is built.

T00IF078D RACLIST directory delete function in progress

Reason: This message displays at termination when the private security directory is deleted.

T00IF079D Processing authorization call

Reason: This message displays when authorization for a user ID is verified.

T00IF080D Result: R15 - *return_code*, AC#PSRC = *feedback_code*

Reason: This message displays the return and feedback codes from the authorization subsystem interface.

T00IF081D Userid = *userid*, Termid = *termid*

Reason: This message usually follows another message and displays the *userid* and *termid*.

T00IF082D Resource=*resource_name*

Reason: This message usually follows another message and displays the resource (such as a data set name).

T00IF083D SECACEE = *address*

Reason: This is associated with ACEE monitoring and displays the SECACEE address.

T00IF084D TCBACEE = *address*

Reason: This is associated with ACEE monitoring and displays the TCBACEE address.

T00IF085D TCBADDR =*address*

Reason: This is associated with ACEE monitoring and displays the TCB address.

T00IF088D Extracting a password

Reason: This message displays when the password for a userid is verified.

T00IF089I IFS timer services started

Reason: This message is a confirmation that the timer support initialization and allocation routines have successfully completed.

Action: None. This is an Informational message.

T00IF090E IFS timer services initialization failed

Reason: The timer initialization routines failed due to storage or system related reasons.

Action: Notify Customer Support.

T00IF091E IFS timer services allocation failed

Reason: The timer allocation routines failed due to storage or system related reasons.

Action: Notify Customer Support.

T00IF092W Storage usage above|below 16 meg line is at *nnn*%

Reason: Storage utilization is above the alarm percentage specified in the MAXSTGPCT keyword of the IFSPARM statement in the IJTCFG*xx* member. Message T00IF093I repors storage utilization dropping back to acceptable levels.

Action: If the utilization drops below the threshold percentage, an occasional spike may be acceptable. If the system is constantly at this high level, a larger region allocation may be appropriate.

T00IF093I Storage usage above|below 16 meg line is now under *nnn*%

Reason: Storage utilization has dropped below the warning threshold. This message should shortly follow T00IF092W. If it does not, refer to the recommended action procedures following message T00IF092W.

Action: None. This is an Informational message.

T00IF094E BLDL failed for table *name*; rc = *ret_code*

Reason: The system BLDL function failed with return code *ret_code*. The module table is identified by *name*.

Action: One of the most common failures is missing libraries; make sure the load library concatenation is correct. Otherwise, examine the return code in the IBM documentation of the BLDL support.

T00IF095E Load failed for module *name*

Reason: The system load mechanism failed for module *name*.

Action: Make sure the module is cataloged somewhere in the load library chain.

T00IF096W Unable to write SMF record; RC = *ret_code*, reason = *rsn_code*, sub-type *sub_type*

Reason: An attempt to write a record to the SMF data set failed. A *ret_code* of 12 indicates that the SMFEWTM macro failed. The SMFEWTM return code is given in *rsn_code*. The SMF record subtype is *sub_type*. If multiple sequential failures occur, All but the first message will be suppressed to avoid flooding the console or message logs. When a record is successfully written to the SMF data set, the internal indicator is reset and another message is written if there is a subsequent failure.

Action: If *ret_code* is 12, see the IBM MVS System Management (SMF) manual for an Reason of the SMFEWTM return codes and respond appropriately. If *ret_code* value is other than 12, an internal error has occurred. Call Customer Support.

T00IF100I Request complete

Reason: The display from the ILATCH TRACE function is complete.

Action: None. This is an Informational message.

T00IF101E Storage acquisition failure

Reason: Insufficient storage to satisfy the ILATCH tracing request.

Action: If this is a transient condition, retry the tracing request.

T00IF102W Tracing already active

Reason: ILATCH tracing was requested earlier and is still active.

Action: None required.

T00IF103W Tracing not active

Reason: ILATCH trace termination did not find active ILATCH tracing.

Action: None required.

T00IF104W The ILATCH support code is not yet operational

Reason: ILATCH tracing is requested too early in the startup sequence.

Action: Wait until more of the system is active and retry.

T00IF105E Entry count cannot be 0

Reason: The number of entries must be at least one.

Action: Correct the ENTRIES keyword and retry.

T00IF107E The application prefix must be T01, T02, or T03.

Reason: An IFS address space was started with an unrecognized application prefix value. The address space is terminated.

Action: Check the application prefix value on the PARM of the EXEC statement in the startup JCL. (It may be coded as PRFX=, PFX=, or P=.) The value must be one of T01 (for the base product), T02 (for the IUCV address space), or T03 (for the trace address space).

T00IF110W Table saturated *num_secs* seconds ago. *num_lost* lost entries

Reason: *num_lost* entries were lost due to a small table *num_secs* seconds ago.

Action: Increase the ENTRIES value and reactivate the ILATCH trace.

T00IF112I Module Offset Latch@ Latch Time RC Type Count

Reason: Header line for message T00IF113I.

Action: None. This is an Informational message.

T00IF113I *aaaaaaaa +bbbb cccccccc ddddd eeeee ff gggg hhhhh*

Reason: Data display for the ILATCH TRACE command.

aaaaaaaa	Name of module issuing the ILATCH request.
bbbb	Hex offset into module aaaaaaaa.
cccccccc	Address of the requested ILATCH block.
ddddd	Number of the requested ILATCH.
eeeee	Time latch held (in seconds).
ff	Return code from latch request call.
gggg	Either "Excl" or "Shrd" to identify the type.
hhhhh	Current use counter.

Action: None. This is an Informational message.

T00IF114I New logging parameters now in effect

Reason: The operator command LOGGING was successfully executed.

Action: None. This is an Informational message.

T00IF115W Error processing logging command, RC=*ret_code*

Reason: The operator command LOGGING encountered parsing errors. A previous message will identify the actual error.

Action: Correct the syntax error and retry the LOGGING command.

T00IF116I Latch Index Requests Suspensions Suspend Time

Reason: Header information for message T00IF117I. This is output from the ILATCH CONTENTION command.

Action: None. This is an Informational message.

T00IF117I *Latch_index Requests Suspensions Suspend_time*

Reason: This message displays latches suspended due to contention.

Action: None. This is an Informational message.

T00IF118I An SVC dump was suppressed by DAE. Dump title:

Reason: A dump request was mad, but the dump was suppressed by Dump Analysis and Elimination (DAE). For more information about DAE, see the IBM manual *OS/390 MVS Diagnosis: Tools and Service Aids*.

Action: None. This is an Informational message.

T00IF119W An SVC dump was suppressed because a SYS1.DUMP dataset was not available. Dump title:

Reason: A dump request was made, but failed because no dump data sets were available.

Action: Clear any unneeded dump data sets.

T00IF120W An SVC dump request failed. Return code *ret_code* was returned from SDUMPX. Dump title:

Reason: A dump request was made, but failed with return code *ret_code*.

Action: See the IBM manual *S/30 MVS Auth Assembler Services Reference, Vol. 3 (LLA-SDU)* for a description of the SDUMPX return codes. Take any appropriate action. If the return code indicates an error on the part of the caller of SDUMPX, contact Customer Support.

T00IF121I *Dump_title*

Reason: This message follows message T00IF118I, T00IF119W, or T00IF120W, and provides the title of the dump referred to in the previous message.

Action: None. This is an Informational message.

T00IF122E *num_suppressed msg_num* Messages dropped due to storage shortage *nnn mmmm*

Reason: More than a maximum number of messages were queued for processing. This happens when a large number of requests for a message are generated. An internal counter for the message number was exceeded and the system is trying to avoid using up large amounts of storage for the message requests.

Messages are created in storage buffers, and queued to a print facility. This design allows the creation of messages in any programming environment, and lets processing continue without regard to actually printing the message. As a result, it is possible to overrun storage with unprinted messages. To avoid this situation, a count is kept of each message number. If too many of a specific message number are queued, later such messages are suppressed. In addition, a count of suppressed messages is also kept. This allows for the *count* printed in this message.

Note: The count is maintained per message number. This allows other messages through, and only suppresses those that are generated in large numbers.

num_suppressed	Number of messages suppressed.
msg_num	Message number of the messages that were suppressed.

T00IF123E Incompatible Prefixes using same subsystem name

Reason: In this release, the startup JCL for TCP, IUCV, and Trace contains a keyword PRFX=. The value for TCP is T01, the value for IUCV is T02, and the value for Trace is T03. Each also contains a keyword UN=. This keyword must have a unique value for TCP, IUCV, and Trace.

This message indicates that a subsystem (UN=) was used for one PRFX= value, then reused for another PRFX= value without an intervening /P CLEAR or re-IPL.

Action: Restart the address space using the previous PRFX= value and issue a /P CLEAR or re-IPL the system, or change the UN= value for this address space.

T00IF124E Restart using Prefix xxx; issue P CLEAR. Restart using current Prefix.

Reason: See message T00IF123E, which also should have been issued. *xxx* is the PRFX= value used previously.

Action: Restart the address space using the previous PRFX= value and issue a /P CLEAR or re-IPL the system, or change the UN= value for this address space.

T00IF900W *command_action* Command Action Not Supported

Reason: The *command_action* specified is not valid.

Action: Reenter the command with the correct action.

T00IF901W Missing Parameter: *value1 value2*

Reason: A command or configuration was entered but was missing parameters. For configuration errors, *value1* is the configuration member name and *value2* is the statement that contains the missing parameter. For command errors, *value1* is the list of possible values and *value2* is blank.

Action: Add the missing parameter.

T00IF902W Invalid Parameter: *value1 value2 value3*

Reason: A command or configuration had invalid parameters. For configuration errors, *value1* specifies the configuration member name, *value2* gives the statement that contains the invalid parameter, and *value3* is the invalid parameter. For command errors, *value1* specifies the invalid parameter and both *value2* and *value3* are blank.

Action: Correct the invalid parameter and reenter.

T00IF991E Msg *msg_num* from *tg_name* – not found in message table

Reason: A standardized message from the tables was requested, but no prototype was found.

msg_num	Message number requested from the message tables.
tg_name	Name of the task group requesting the message.

Action: The message request is bypassed. Processing continues. Notify development support personnel.

T00IF992E Msg *msg_num* had a formatting error - reason(*reason_code*)

Reason: A prototype message from the tables was found to have format errors. The most likely cause is ampersands not coded in pairs, or the character after a pair of ampersands not being recognized as an editing letter-type designator.

msg_num	Message number requested from the message tables.
reason_code	Reason code for the formatting error.

Action: The message request is bypassed. Processing continues. Notify your development system support group or Customer Support.

T00IF993E Abend *abend* occurred formatting msg *msg_num* for *tg_name*

Reason: An ABEND occurred while issuing a message for the user. The most likely cause is a bad addresses being requested for display. *abend* is a hexadecimal ABEND code associated with the failure.

abend	Hexadecimal ABEND code associated with the failure.
msg_num	Message number requested from the message tables.
tg_name	Name of the task group requesting the message.

Action: The message request is bypassed. Processing continues with recovery attempted. Notify your development system support group or Customer Support.

T00IF994E Component *XX* not recognized

Reason: The IMSG formatting routines encountered a message prefix with an invalid component identifier in positions four-five.

Action: This is an internal problem. Call Customer Support.

T00IF999I

Reason: A general message for terminating multi-line WTO sequences.

Action: None. This is an Informational message.

Chapter

10

T00IJ*nnn* IJT Messages

This section describes messages issued by the IFS task group. These messages include T00IJ000 through T00IJ099.

T00IJ*nnn*

T00IJ000I *command_string*

Reason: An operator command was issued. This message echoes the command to the log.

Action: None. This is an Informational message.

T00IJ001I Invalid command *command*

Reason: The command (*command*) could not be identified by any of the usual methods for further processing. The command is ignored.

Action: Issue a valid command.

T00IJ002I Missing command verb

Reason: The TSO Parse Service routine found no command verb to process.

Action: Reissue the command with a valid command verb.

T00IJ003I Command *cmnd_name* is directed to TGB *rtgid* not *stgid*

Reason: A command defined for use by a specific task group was requested to be directed to another task group where it is not defined. The command is ignored.

cmnd_name	Name of the command.
rtgid	Task group requested in the command.
stgid	Only task group that supports the specified command.

Action: Reissue the command, omitting the task group specification, or select a different command for the specified task group.

T00IJ004I Task group *task_group_id* not found

Reason: The task group *task_group_id* supporting this command is not present. The command is ignored.

Action: Activate the task group by starting a task within the task group.

T00IJ005I Task group *task_group_id* has no active tasks

Reason: The requested task group (*task_group_id*) supporting this command has no active tasks to process this command.

Action: Start a task within the task group, then reissue the command.

T00IJ006I Invalid member name following %

Reason: A command script execution that was requested via the % was ignored because the member name following the % did not match the rules for a member name. Either the name was too long or it contained special characters.

Action: Reissue the command with a valid member name.

T00IJ007I L=*nna* is invalid

Reason: The L= specification on a command is invalid. Either the console number is invalid or the screen area specified by the letter does not exist on the console.

nn Console requested for direction of screen output.

a Screen area requested for direction of command output.

Action: Reissue the command with a valid L= specification or omit the L= specification. Test the validity of the console number alone by omitting the screen area specification (that is, L=nn).

T00IJ008I *mem image*

Reason: Displays command images from the command script being executed.

mem Command script member name.

image Image of the command line being executed.

Action: To suppress listing of every command as it is executed, insert a NOLIST statement in the command script to disable subsequent listing of command images.

T00IJ009I *mem* Rc=*rc image*

Reason: Displays a command image from the script being executed. The command displayed has ABENDed or received a nonzero return code.

mem Command script member name.

rc Hexadecimal return code or ABEND code.

image Image of the command line completing in error.

Action: If NOFLUSH is not active, subsequent commands are bypassed.

T00IJ010I SYSPROC DD failed to open for cmd script processing

Reason: An open failure occurred during an attempt to read and process a command script.

Action: Check to see if the SYSPROC DD statement is present in the JCL procedure for the address space and that the library has not been destroyed.

T00IJ011I Member *mem* not found in SYSPROC library

Reason: The specified command script was not present in any data set defined by the SYSPROC DD statement in the address space JCL or current SYSPROC allocation. The *mem* is the command script member name requested.

Action: Check to see if the requested member is present in the current SYSPROC DD statement in the address space JCL or current SYSPROC allocation.

T00IJ012I *mem* flushed *image*

Reason: Due to a previous command completing in error (nonzero return code), subsequent commands were bypassed.

mem Command script member name.

image Image of the command line whose execution was bypassed.

Action: If the command is not dependent on successful execution of previous commands, insert a NOFLUSH statement in the command script where dependencies do not exist.

T00IJ013R Confirm request to stop A/S -- Reply 'Y' or 'N'

Reason: This is in response to an operator stop command for the address space. To avoid a mistake (such as the wrong address space subsystem recognition character used or another command starting with P intended), the requestor is asked to confirm or cancel the stop request. The stop request is delayed until a response to this message is received.

Action: Determine if the stop request was intended. Reply either Y or N:

Y Continues termination processing. Message T00IJ014I follows.

N Cancels the stop request. Message T00IJ015I follows.

T00IJ014I Stop request confirmed

Reason: A reply of Y was received to message T00IJ013R. Address space termination is scheduled.

Action: None. This is an Informational message.

T00IJ015I Stop request cancelled

Reason: A reply of N was received to message T00IJ013R. Shutdown is cancelled and address space execution continues as if the stop request was not entered.

Action: None. This is an Informational message.

T00IJ016W Stop already requested -- awaiting response

Reason: This is in response to an operator stop command. A stop command was already issued and termination is pending a response to message T00IJ013R. The stop request is ignored.

T00IJ017I Stop request specified P CLEAR. Modules and control blocks will be removed from CSA

Reason: When terminating the address space, the CLEAR option was specified. Modules and control blocks used to interface to this address space are deleted from common storage.

Note: Programs in other address spaces may ABEND if they attempt to execute these interface modules or reference these control blocks

Action: None. This is an Informational message.

T00IJ018I Unknown SYSPARM statement: *statement*

Reason: While reading the IJTCFG*xx* member from the PARM library, IJT initialization found a statement (statement) other than the expected POOLDEF or GTFID statements. The statement is ignored.

Action: Verify the *statement* given. Processing continues with the next recognizable statement.

T00IJ019I *type* task display

Reason: Identifies the header line for the TASK display command. Information follows about active subtasks in the address space. The type of task group displayed is IFS or NON-IFS.

Note: Message numbers T00IJ020I and T00IJ021I follow this message; they are printed without message numbers.

Action: None. This is an Informational message.

T00IJ020I Tbg Flg1 Mx Task# Cnfg-Mem Flg2 Dsp Count Start Time

Reason: Header line for the TASK display command.

Action: None. This is an Informational message.

T00IJ021I tgid gflgs mxtsk memnm tflgs dspcnt mm/dd/yy hh:mm:ss

Reason: This is a detail line for the TASK display command.

tgid	Task group ID.
gflgs	Hexadecimal display of two flag bytes in the IFS Task Group Block.
mxtsk	Maximum number of tasks for the task group of this task (decimal).
memnm	Configuration member used for task group startup.
tflgs	Hexadecimal display of two flag bytes in the IFS Task Control Block.
dspcnt	Number of times this task has been dispatched with work to do (decimal).
mm	Month of the date started.
dd	Day of the date started.
yy	Year of the date started.
hh	Hour of the time started.
mm	Minute of the time started.
ss	Second of the time started.

Action: None. This is an Informational message.

T00IJ022I No *type* tasks to display

Reason: There were no TCBs to display for the requested type (IFS or NON-IFS).

Action: None. This is an Informational message.

T00IJ023I TGB *task_group_id* has no active tasks

Reason: A task display command was requested with a specific task group(s). There were no active tasks within the task group to display.

Action: None. This is an Informational message.

T00IJ024I Pool *category* display

Reason: This is an identifying header line for the POOL display command. Information follows about pool headers in the address space. The category returned is either ATTRIBUTE or COUNTS, depending on the display option requested in the command request.

Note: If the ATTRIBUTE option is selected, message numbers T00IJ025I and T00IJ026I follow this message. If the COUNT option is selected, message numbers T00IJ027I and T00IJ028I follow this message. In either case, the messages that follow T00IJ024I are printed without message numbers.

Action: None. This is an Informational message.

T00IJ025I Name Flgs Sp Lgth Ex# Cn# Thrhld

Reason: This is a column header line for the POOL display command with the ATTRIBUTES option. Messages T00IJ025I and T00IJ0026I are printed without message numbers.

Action: None. This is an Informational message.

T00IJ026I *phdr pflgs sp lng expcnt concnt conthld*

Reason: This is a detail line for the POOL display command with the ATTRIBUTES option.

phdr	Pool header name.
pflgs	Hexadecimal display of two flag bytes in the IFS Pool Header.
sp	Subpool number that pool elements reside in (decimal).
lng	Length of pool elements (decimal).
expcnt	Count of times the pool has expanded (decimal).
concnt	Count of times the pool has contracted (decimal).
conhld	Minimum threshold this pool may be contracted (decimal).

Action: None. This is an Informational message.

T00IJ027I Name Flgs Total Free Lwm Hwm Nmbr Reqs #Exp #Con Wcnt Errs

Reason: This is a column header line for the POOL display command with the COUNTS option. Messages T00IJ027I and T00IJ028I are printed without message numbers.

Action: None. This is an Informational message.

T00IJ028I *phdr pflgs total free lwm hwm numreqs #exp #con waitcnt errcnt*

Reason: This is a detail line for the POOL display command with the COUNTS option.

phdr	Pool header name.
pflgs	Hexadecimal display of two flag bytes in the IFS Pool Header.
total	Total number of elements being managed (decimal).
free	Total number of free elements in this pool (decimal).
lwm	Low water mark or minimum number of free elements reached so far during processing (*decimal*).
hwm	High water mark or maximum number of free elements reached so far during processing (decimal).
numreqs	Number of requests for elements from this pool received so far during processing *(decimal).*
#*exp*	Number of elements to expand this pool when expansion is necessary (*decimal*).
#*con*	Number of elements to contract this pool when contraction is necessary (*decimal*).
waitcnt	Number of times requests had to wait for elements while pool expansion occurred (*decimal*).
errcnt	Number of times requests had errors during processing (*decimal*).

Note: The interpretation of these headings is different for the STAK pool than for the other control block pool names:

- The 'Nmbr Reqs' (number of requests) is not used and remains zero regardless of usage.
- The '#Exp' (number of expansions) in this pool represents the number of times an additional 4K Stack Frame was needed to accommodate a module's working storage requirements. At entry, a default of 4KB is supplied. This number does not mean how many times the pool has been expanded as is the meaning for other pools.
- The "#Con' (number of contractions) is not used and remains zero.

Action: None. This is an Informational message.

T00IJ029W Pool header *pool_header* not found

Reason: A POOL display command was requested with a specific pool header(s). There were no pool headers found by the requested name.

T00IJ030W Invalid pool header Id: *pool_header_id*

Reason: A POOL display command was requested with a specific pool header(s). The pool header name had a bad format or incorrect length.

T00IJ031I *attr* Attribute set *on/off* for *tgid* Tbg/Tcbs *xcept*

Reason: This is in response to a successful set command request.

attr	Name of the attribute being set: DEMO, TEST, GTF, or SUPV.
on/off	Setting value ON or OFF.
tgid	Task group ID.
xcept	Either blanks or the string (EXCEPT AS NOTED) to highlight exceptions.

Action: None. This is an Informational message.

T00IJ032I *attr* Attribute not set *on/off* for *tgid* Tbg/Tcbs *reason*

Reason: This is in response to an unsuccessful set command request.

attr	Name of the attribute being set: DEMO, TEST, GTF, or SUPV.
on/off	Setting value ON or OFF.
tgid	Task group ID.
reason	Reason the setting was not performed.

Action: The requested action is incomplete. For SUPV attribute, it must be set for the task group before subtasks are started. It cannot be changed while subtasks are active. Stop all subtasks before setting the SUPV attribute then restart subtasks.

T00IJ033I Current *subsys_rec_char* is: *c*

Reason: This is a response to a display of the current subsystem recognition character. *c*, is used to identify commands to a particular address space.

Action: None. This is an Informational message.

T00IJ034I New *subsys_rec_char*: *c*

Reason: This is a response to a change of the current subsystem recognition character. This new character, *c*, is used to identify commands to a particular address space.

Action: None. This is an Informational message.

T00IJ035I No *subsys_rec_char* is being used

Reason: The TCP base product is not using a subsystem recognition character. All operator commands to the TCP base address space must be done via the MVS MODIFY command.

Action: None. This is an Informational message.

T00IJ036W Invalid *subsys_rec_char*: *c*

Reason: This is a response to a change of the current subsystem recognition character. This new character for identifying commands to this address space, *c*, is invalid or too many characters were specified. The command is ignored.

Action: Reissue the command with a valid single character.

T00IJ038I GTF ID Set Mod_name Function CB Desc

Reason: This is a column header line for the GTF display command. Information follows about GTF events in the address space.

Action: None. This is an Informational message.

T00IJ039I *symb_evnt on/off mod_name action cblk desc*

Reason: This is a detail line for the GTF display command.

symb_evnt	Symbolic event name.
on/off	Word ON or OFF depending on event setting.
mod_name	Name of the module issuing the trace event.
action	Name of the function related to the trace event.
cblk	Name of the control block being traced.
desc	Description of the event being traced.

Action: None. This is an Informational message.

T00IJ040I All GTF events set *on/off*

Reason: A response, either ON or OFF, to a GTF command without specific events specified. All event tracings are enabled or disabled as requested.

Action: None. This is an Informational message.

T00IJ041I GTF Id=*event_id* set *on/off*

Reason: A response to a GTF command with specific event(s) specified. The specific event is enabled or disabled as requested.

event_id	Keyword name of the GTF event.
on/off	Word ON or OFF depending on the request for enablement or disablement of event tracing.

Action: None. This is an Informational message.

T00IJ042I No GTF events found *on/off*

Reason: This is a response, ON or OFF, to a GTF command to display all events enabled or disabled. There were none found in the requested state.

Action: None. This is an Informational message.

T00IJ043W GTF *type=spec* unknown

Reason: This is a response to a GTF command with an event type specified. No event(s) of the requested type were found.

type Type of GTF event being specified: EVENT, CNTRLBLK, or MODULE.

spec Specification of event type.

Action: Reissue the command with a correct specification.

T00IJ044I GTF Events not defined for this address space - *name*

Reason: This is a response to a GTF command when no event description table could be located for processing. The name is the unique name of the address space where the failure occurred.

Action: None. This is an Informational message.

T00IJ045W Invalid *on/off* parameter

Reason: This is a response, either ON or OFF, to a modify GTF command. No enable (ON) or disable (OFF) action word was specified.

T00IJ046I Cvt=*cvt* Scvt=*scvt* Ucm=*ucm* Smca=*smca* Ipl-vol=*sys_vol*

Reason: This is a response to an MVS display command. Addresses of MVS control blocks are shown. All are hexadecimal except sys_vol, which is decimal.

cvt	Address of the MVS Communication Vector Table.
scvt	Address of the MVS Secondary Communication Vector Table.
ucm	Address of the MVS Unit Control Module.
smca	Address of the MVS System Management Control Area.
sys_vol	Volume serial of the system residence volume.

Action: None. This is an Informational message.

T00IJ047I Lch-Tab=*lchtab* Trace-Tab=*trctab* Nuc=*nucendK* V=R=*vrK* Csa=*csaK*

Reason: This is a continued response to an MVS display command for an MVS/370 system (SP 1).

lchtab	Address of the logical channel word table (hexadecimal).
trctab	Address of the trace table (hexadecimal).
nucend	Lowest address not in the nucleus, representing top-of-nucleus (decimal).
vr	Address of the virtual storage byte following the highest virtual=real storage address (decimal).
csa	Beginning address of the Common Service Area (decimal).

Action: None. This is an Informational message.

T00IJ048I Real-Stor-Onli=*onli*K Mem=Size=*size*M

Reason: This is a continued response to an MVS display command.

onli	Size of actual real storage online at IPL time in kilobytes (decimal).
size	Physical memory size in megabytes (decimal).

T00IJ049I V=R=*vr*K Csa=*csa*K

Reason: This is a continued response to an MVS display command for an MVS/XA system (SP 2).

vr Address of the virtual storage byte following the highest virtual=real storage address (decimal).

csa Beginning address of the Common Service Area (decimal).

Action: None. This is an Informational message.

T00IJ050I NUC: R/W= *rwsa*K (*rwsz*K) R/O= *rosa*K (*r oszK*) X-R/W= *rwsa*K (*rwsz*K)

Reason: This is a continued response to an MVS display command for an MVS/XA system (SP 2).

rwsa Starting virtual address of the read/write nucleus (decimal).

rwsz Size of the read/write nucleus in K-bytes (decimal).

rosa Starting virtual address of the read/only nucleus (decimal).

rosz Size of the read/only nucleus in K-bytes (decimal).

rwsa Starting virtual address of the extended read/write nucleus (decimal).

rwsz Size of the extended read/write nucleus in K-bytes (decimal).

Action: None. This is an Informational message.

T00IJ051I *area=add1*K(*size1*K) X-*xarea=add2*K(*size2*K)

Reason: This is a continued response to an MVS display command for an MVS/XA system (SP 2).

area Name of an area in the LPA (link pack area), which can be one of these:

MLPA The modified link pack area.

FLPA The fixed link pack area.

PLPA The pageable link pack area.

add1 Starting virtual address of the area (decimal).

size1 Size of the area in KB (decimal).

xarea Name of an area in the X-LPA (extended link pack area).

add2 Starting virtual address of this area (decimal).

size2 Size of the area in KB (decimal).

Action: None. This is an Informational message.

T00IJ052I Ssct=*ssct* Name=*name ssitinfo*

Reason: This is a continued response to an MVS display command with the SSCT option keyword.

ssct Hexadecimal address of the Subsystem Communication Table.

name Subsystem name.

ssitinfo Appears when displaying an IFS SSCT and includes IFS related items if applicable, such as the address of the Subsystem Information Table address, the subsystem recognition character, and whether the address space is active or inactive.

Action: None. This is an Informational message.

T00IJ053I Lnklst Dsns System: *smfid*

Reason: This is an identifying header line for the MVS display command. It gives the SMF system identifier for this system. Information follows about data sets in the system link list from SYS1.PARMLIB(LNKLST *nn*).

Action: None. This is an Informational message.

T00IJ054I *dsname*

Reason: This is a detail line for the MVS LNKLST display command. It gives the data set name that is in the MVS/XA System Linklist.

Action: None. This is an Informational message.

T00IJ055I Display lnklst available only under MVS/ESA

Reason: A Display LNKLST command was requested in a non-MVS/XA environment. This information is available programmatically only in MVS/XA systems. The request is ignored.

Action: None. This is an Informational message.

T00IJ056I APF table display A=*addr* ENTRIES=*n*

Reason: This is a response to an APF display command. A series of IJT162 messages follows, one for each table entry that matches the display categories.

addr The hexadecimal address of the APF table.

n The (decimal) number of entries in the APF table.

Action: None. This is an Informational message.

T00IJ057I A(Entry) Volser Dsname

Reason: This is a column header line for the APF display command.

Action: None. This is an Informational message.

T00IJ058I *address volume dsname*

Reason: This is a detail line for the APF display command.

address Hexadecimal memory address of this entry.

volume DASD volume serial on which the data set is authorized.

dsname Data set name authorized by this entry.

Action: None. This is an Informational message.

T00IJ059I *num* Unused bytes

Reason: This is a response to an APF command. It displays the number of unused bytes (*num*) available in the table to avoid IPL-time problems with an IEAAPF *nn* list containing too many entries. The APF table has a maximum size at IPL time of 2040 bytes (there is also a limit of 256 entries). Each entry requires seven bytes plus the nonblank length of the data set name. The Modify APF command can expand the table beyond the 2040 byte limit after IPL completes. This message is not issued if the table has been expanded beyond 2040.

Action: None. This is an Informational message.

T00IJ060I No entry found for *volser dsname*

Reason: This is a response to an APF command. There is no entry in the table matching the display/modify specifications.

volser Volser specified.

dsname Dsname or dsname-stem specified.

Action: Reissue the command with new specifications.

T00IJ061I *dsname* Now *un* authorized on *volser*

Reason: This is a response to a modify APF command. The APF list was modified appropriately.

dsname Dsname or dsname-stem specified.

un Either UN if the data set has been deleted, or blanks if the data set has been added.

volser Volser specified.

Action: None. This is an Informational message.

T00IJ062I *dsname* Now authorized on *vol1* Device *vol2*

Reason: This is a response to a modify APF command. The APF list was modified appropriately.

dsname	Dsname or dsname-stem specified.
vol1	First DASD volume serial on which the data set is authorized.
vol2	Second DASD volume serial on which the data set is authorized.

Action: None. This is an Informational message.

T00IJ063I Entry exists for *volser dsname*

Reason: This is a response to a modify APF command. A request to add a new entry matches an existing entry.

volser	Volser specified.
dsname	Dsname specified.

Action: Reissue the command with new specifications.

T00IJ064I A/S startup date: *mm/dd/yy* time: *hh:mm:ss*

Reason: This is a response to an IFS command. Startup time of the address space is shown.

mm	Month of the date started.
dd	Day of the date started.
yy	Year of the date started.
hh	Hour of the time started.
mm	Minute of the time started.
ss	Second of the time started.

T00IJ065I Asid=*asid* A(ASCB)=*ascb* A(SSCT)=*ssct* A(SSIT)=*ssit*

Reason: This is a response to an IFS command. Address-space-related control blocks are shown.

asid	Hexadecimal address space identification number assigned to this address space.
ascb	Hexadecimal address of the Address Space Control Block for this address space.
ssct	Hexadecimal address space of the MVS Subsystem Communication Table for this address space.
ssit	Hexadecimal address of the IFS Subsystem Information Table for this address space.

Action: None. This is an Informational message.

T00IJ066I Cpu=*cpu_name* Smfid=*smfi* Mvs=*mvs_type*

Reason: This is a response to an IFS command. MVS and IFS information is shown.

cpu_name	IFS-assigned name for this CPU.
smfi	MVS SMF identification for this CPU.
mvs_type	MVS SP level identification for the running System Control Program.

Action: None. This is an Informational message.

T00IJ067I Lx=*lx_num* Ssnam=*ssname* Appl=*app_addr* Sout=*cls* Jobi=*jobid*

Reason: This is a response to an IFS command. MVS and IFS information is shown.

lx_num	Hexadecimal MVS LX assigned for cross-memory usage by this address space.
ssname	MVS subsystem name chosen by this address space.
app_addr	IFS application type identifying this address space.
cls	Character identifying the JES SYSOUT class used by default for this address space.
jobid	Identification string used when starting this address space. This identifier can be used in MVS modify (F) commands to specify commands be directed to this address space.

Action: None. This is an Informational message.

T00IJ068I A(ICVT)=*icvt*

Reason: This is a response to an IFS command. It gives the address of the IFS Communication Vector Table.

Action: None. This is an Informational message.

T00IJ069I *cp_name* CP Ecbrc=*rc*

Reason: This is an identifying header line for the TSO command. Information follows if formatted from the TSO command processor response.

cp_name	Command processor name invoked.
rc	Hexadecimal return code from the ECB for the command processor invoked.

Action: None. This is an Informational message.

T00IJ070I *command*

Reason: This is a response to a TSO operator command. The actual TSO command requested is shown.

Action: None. This is an Informational message.

T00IJ071I *response*

Reason: This is a detail line for the TSO command processor response. It displays any line returned from a TSO command processor. Response lines are formatted as they would appear in response to the TSO command issued directly to a TSO terminal session.

Action: None. This is an Informational message.

T00IJ072I Command processor *cp_name* abended RC=*rc*

Reason: The requested TSO command did not complete normally. The ABEND code or completion code is shown.

cp_name	Command processor name invoked.
rc	Hexadecimal ECB completion code for the command processor invoked.

Action: None. This is an Informational message.

T00IJ073I Stack command failed RC=*rc*

Reason: Internal STACK Macro processing failed in preparation of issuing the TSO command processor. The decimal STACK return code (*rc*) is shown.

Action: Reissue the command, if desired. If failure persists, notify Customer Support.

T00IJ074I VM/CP command response

Reason: This is an identifying header line for the CP command. Information follows if formatted from the CP command processor.

Action: None. This is an Informational message.

T00IJ075I VM/CP command return code: *rc*

Reason: This is a response to a CP operator command. The return code value (*rc*) from a CP command is shown.

Action: None. This is an Informational message.

T00IJ076I *response*

Reason: This is a detail line for the CP command. The *response* is any line returned from a CP command. Response lines are formatted as they would appear in response to the CP command issued directly to CP from the virtual machine console.

Action: None. This is an Informational message.

T00IJ077I CP command complete

Reason: This indicates the end of formatting response lines from the CP command.

Action: None. This is an Informational message.

T00IJ078I Load real address failure before diagnose issued

Reason: This indicates the CP command had an LRA instruction failure.

Action: Reissue the command. If failure persists, notify Customer Support.

T00IJ079I VCMD=*vcmd_reg* RCMD=*rcmd_reg* VBUF=*vbuf_reg* RBUF=*rbuf_reg*

Reason: This is in response to an LRA failure and accompanies IJT196. Diagnostic information is displayed.

vcmd_reg	Hexadecimal value of the VCMD register.
rcmd_reg	Hexadecimal value of the RCMD register.
vbuf_reg	Hexadecimal value of the VBUF register.
rbuf_reg	Hexadecimal value of the RBUF register.

Action: Supply this information to Customer Support.

T00IJ080I Command response exceeds buffer length by *excess* bytes

Reason: A large response text was produced during execution of a CP command. By default, the command obtains 2 KB of storage to hold the command response. The resultant output exceeded this buffer; the number of bytes by which it exceeded 2 KB is shown (*excess*). As much of the command response as obtained is formatted for display.

Action: If total command response is required, notify Customer Support.

T00IJ081I CP command invalid; MVS not in virtual machine

Reason: A CP command was issued to an IFS address space running under a native MVS system. That is, VM/CP is not supporting this system.

Action: None. This is an Informational message.

T00IJ082I *trace_msg on/off trtszK*

Reason: This is a response to a TRACE command. The size of the trace table is shown in kilobytes.

trace_msg	The string INTERNAL TRACE SET.
on/off	The string ON or OFF.
trtsz	The size of the internal trace table (decimal).

Action: None. This is an Informational message.

T00IJ083I *Internal_trace* OFF

Reason: This is a response to a TRACE command.

Action: None. This is an Informational message.

T00IJ084I Maximum of *trtmxpg* pages may be specified

Reason: This is a response to a modify TRACE command attempting to change the table size. The *trtmxpg* is the maximum number of pages in the internal trace table.

Action: None. This is an Informational message.

T00IJ085I Julian date=*date*, Gregorian date=*mm*/*dd*/*yy*

Reason: This is a response from a command showing date information. For the TIME command, this information is the current date. For the STCK command, this information is the converted date.

date	The Julian date (decimal).
mm	A Gregorian month.
dd	A Gregorian day.
yy	A Gregorian year.

Action: None. This is an Informational message.

T00IJ086I Time=*hh*:*mm*:*ss*.*ff*, Stck=*stck*

Reason: This is a response from a command showing time information. For the TIME command, this information is the current time. For the STCK command, this information is the converted date.

hh	The hours.
mm	The minutes.
ss	The seconds.
ff	The hundredths seconds.
stck	The STCK format value, in hexadecimal.

Action: None. This is an Informational message.

T00IJ087I Seconds past midnight=*daysecs*

Reason: This is a response from a command (TIME or STCK) that shows time information. The *daysecs* parameter is the elapsed seconds for the day.

Action: None. This is an Informational message.

T00IJ088E TGB(*tgid*) not found, Vavt Add(*name*) failed

Reason: This is a response to a VAVT command.

tgid The task group ID.

name The VAVT name.

Action: Verify the existence of the *tgid* as it was entered and retry.

T00IJ089E VAVT Add(*name*) Failed, maxusers exceeds slots

Reason: This is a response to a VAVT command. The *name* parameter is the VAVT name.

T00IJ090I Help for cmnd: *cmnd_name*

Reason: This is an identifying header line for the HELP display command. It returns the name of the command for which help is displayed.

Action: None. This is an Informational message.

T00IJ091I *helpdata*

Reason: This is a detail line for the HELP command.

Action: None. This is an Informational message.

T00IJ092I Help command complete

Reason: This indicates the end of formatting response lines from the HELP command.

T00IJ093I No help available for *command*

Reason: This indicates that no help is available for the requested command.

Action: None. This is an Informational message.

T00IJ094I Module Address Size Call Cnt Table Fixed Asm Date/Time APAR-ID

Reason: This is an identifying header line for the MODULE display command.

Action: None. This is an Informational message.

T00IJ095I *mod_nam3 mod_addr size call_cnt PCTBL fixed date_time fix_#*

Reason: This is a detail line for the MODULE command.

mod_name	The module name.
mod_addr	The beginning hexadecimal address in memory of the module.
size	The module size, in hexadecimal.
call_cnt	The module call count, in decimal.
PCTBL	The load table name where the module is defined.
fixed	An indication (FIXED), if the module is in page-fixed storage.
date_time	The date/time of assembly.
fix_#	The source level fix identifier.

Action: None. This is an Informational message.

T00IJ096I Total *tot_siz tot_call*

Reason: This is a summary line for the MODULE command. The total of all module sizes and call counts are displayed.

tot_siz	The total size of all modules, in hexadecimal.
tot_call	The total number of module calls, in decimal.

Action: None. This is an Informational message.

T00IJ097I Missing module name(s)

Reason: The required module name parameter is missing.

Action: Reissue the command with a module name, list of modules, ALL, or an asterisk (*).

T00IJ098W *mod_name* Alias = replace rejected

Reason: The MODIFY MOD command was attempting to replace a module (*mod_name*), but because the specified module name is an alias, the command was rejected.

Action: Reissue the command using the actual module name.

T00IJ099W *mod_name* is PC type - replace rejected

Reason: The MODIFY MOD command was attempting to replace a module, but the specified module name (*mod_name*) is a type called via PC. These modules can be replaced only at address space startup time.

Action: Command rejected. Reissue the command with a module that can be replaced, or restart the address space if this module must be replaced.

T00IJ100W *mod_name* Module not in PC tables

Reason: A requested module (*mod_name*) was not found in any PC or Load Table.

Action: Command rejected. Reissue the command with a module name that exists in the tables.

T00IJ101I LSPACE *operand* display

Reason: This is an identifying header line for the LSPACE display command. The *operand* parameter indicates the operand supplied with LSPACE to specify the DASD subclass to display.

Action: None. This is an Informational message.

T00IJ102I Avail Nmbr Contig Status

Reason: This is an identifying header line for the LSPACE display command.

Action: None. This is an Informational message.

T00IJ103I Cyl Trk Xtnt Cyl Trk Flags

Reason: This is a second identifying header line for the LSPACE display command.

Action: None. This is an Informational message.

T00IJ104I *ucb volser vol_serial flags*

Reason: This is a data line for the LSPACE command. It shows the available space for a DASD volume, one per line.

ucb The UCB device address.

volser The volume serial mounted at the device address.

vol_serial The available space data as returned by MVS. The data is returned in this format:

- acyl, atrk, nmex/ ccyl, ctrk.
- acyl is the number of available free cylinders.
- atrk is the number of available free tracks.
- nmex is the number of extents of free space.
- ccyl is the number of contiguous free cylinders.
- ctrk is the number of contiguous free tracks.

flags The settings of related UCB status bits.

Action: None. This is an Informational message.

T00IJ105I Unsupported device name *dasdid*

Reason: The specified DASD type (*dasdid*) could not be located.

Action: Command rejected. Reissue the command with a valid DASD address, volser, or class.

T00IJ106I UCB *ucb* Not ready or invalid UCB specified

Reason: Information about the specified UCB device address (*ucb*) is unavailable. The device address is invalid or the device is off-line.

Action: Command rejected. Reissue the command with a valid DASD address, volser, or class.

T00IJ107I PCFTBLS Module not usable

Reason: No operand was specified on the command. The PCFTBLS module was not found for the default volser list.

Action: Command rejected. There is no corrective action. Without availability to PCFTBLS, the environment does not support this command form.

T00IJ108I No *attr* Devices mounted

Reason: No DASD volumes were found with the requested mount attribute (*attr*). The command is rejected.

Action: There is no corrective action. Until volumes are mounted with the requested attribute, the message recurs on subsequent requests.

T00IJ109I An error was encountered in the Lspace Svc routine; return code *ret_code*

Reason: An internal error occurred in the MVS SPACE SVC. The command is rejected. Processing continues. The code parameter is the SPACE supervisor return code. Corrective action depends on the return code.

Action: None. This is an Informational message.

T00IJ110I Error in DCQ chain -- function cannot be performed

Reason: An internal error occurred searching the DCQ table for DASD entries. The command is rejected. Corrective action is indeterminate. The system may be corrupted.

Action: None. This is an Informational message.

T00IJ111I Error in DASD UCB chain -- function cannot be performed

Reason: An internal error occurred. After finding DASD in the DCQ table, there do not appear to be any UCB addresses. The command is rejected. Corrective action is indeterminate. The system may be corrupted.

Action: None. This is an Informational message.

T00IJ112I Storage unavailable to build UCB list

Reason: An internal error occurred while processing an IFS SPACE command. The system routine IEFEB4UV could not obtain storage to return the results for the request. The command is rejected. Processing continues. Corrective action is indeterminate.

Action: None. This is an Informational message.

T00IJ113I *ucb volser* Permanent I/O error

Reason: An internal error occurred in the MVS SPACESVC. The command is rejected. Processing continues. Corrective action is indeterminate.

ucb UCB device address.

volser Volume serial mounted at the device address.

Action: None. This is an Informational message.

T00IJ114I *volser* Volume not mounted

Reason: LSPACE information about the specified volume serial (*volser*) mounted at the device address could not be returned. The volume serial could not be located. The command is rejected. Processing continues.

Action: Reissue the command, if desired, with a valid DASD volume serial.

T00IJ115I *ucb* Not a direct access device

Reason: The specified device address (*ucb*) is not DASD type. LSPACE has no meaning. The command is rejected.

Action: Reissue the command with a valid DASD unit address.

T00IJ116I Error: Unknown or invalid device specified

Reason: This is a general catchall for all other errors from LSPACE. The command is rejected. Processing continues. Corrective action is indeterminate.

Action: None. This is an Informational message.

T00IJ117I Virtual storage usage

Reason: This is an identifying header line for the VSM display command.

Action: None. This is an Informational message.

T00IJ118I ------ Allocated ------ -------- In Use -------- Task

Reason: Error Messages T00IJ018I and T00IJ019I make up a two line header for the VSM display command.

Action: None. This is an Informational message.

T00IJ119I Sp Key Regn Below Above Total Below Above Total

Reason: Error Messages T00IJ018I and T00IJ019I make up a two line header for the VSM display command.

Action: None. This is an Informational message.

T00IJ120I *sp key type allocB allocA totalA useB useA totalU task*

Reason: This is a detail line for the VSM command. Allocated and free space for each subpool are shown on a separate line

sp	The subpool number (decimal).
key	The hexadecimal storage key for the subpool.
type	The storage area type containing the subpool, PVT, CSA, SQA, or LSQA.
allocB	The number of blocks of allocated storage below the 16MB line for this subpool (decimal).
allocA	The number of blocks of allocated storage above the 16MB for this subpool (decimal).
totalA	The total size of all blocks (in 1024 byte units) of allocated storage for this subpool (decimal).
useB	The number of blocks (in 1024 byte units) of storage in use for this subpool below the 16 MB line.
useA	The number of blocks (in 1024 byte units) of storage in use for this subpool above the 16 MB line.

totalU The total amount of all storage in use within this subpool (decimal).

task The first entry for a task group identifies the owning task group.

Action: None. This is an Informational message.

T00IJ121I *total type allocB allocA totalA useB useA totalU*

Reason: This is a detail line for the VSM command showing total storage in use for the storage type PVT, CSA, SQA, or LSQA.

total A field indicating whether this is a TOTAL line for the virtual storage grouping or a GRAND total for all virtual storage groupings.

type The storage area type containing the subpool: PVT, CSA, SQA, or LSQA; blank on the GRAND total line.

allocB The number of blocks of allocated storage below the 16 MB line for this subpool (decimal).

allocA The number of blocks of allocated storage above the 16 MB for this subpool (decimal).

totalA The total size of all blocks (in 1024 byte units) of allocated storage for this subpool (decimal).

useB The number of blocks (in 1024 byte units) of storage in use for this subpool below the 16 MB line.

useA The number of blocks (in 1024 byte units) of storage in use for this subpool above the 16 MB line.

totalU The total amount of all storage in use within this subpool (decimal).

T00IJ122I VSMLIST macro received RC=*rc* for *type* data

Reason: The VSMLIST system service has returned a nonzero return code for the specified storage group request.

rc The decimal return code received See *SPL:System Macros and Facilities Volume 1* regarding return codes from VSMLIST.

type The storage area type containing the subpool: PVT, CSA, SQA, or LSQA.

T00IJ123T *bytes* bytes used in VSM work area for *type* data

Reason: This is an informational message shown only when TEST mode is set on to indicate how much of the VSMLIST work area was used on a request.

bytes The number of bytes of data used in the VSMLIST work area of the program (decimal).

type The storage area type containing the subpool: PVT, CSA, SQA, or LSQA.

T00IJ124I *type* request exceeded VSMLIST workarea size (*bytes*)

Reason: The VSMLIST system service has given a 4 return code for the specified storage group request. The amount of data to be returned exceeds the work area.

type Storage area type being processed on the failing request: PVT, CSA, SQA, or LSQA.

bytes Size in bytes allocated for the VSMLIST work area of the program (decimal).

Action: The request is terminated at the failing storage type. Notify Customer Support to develop a circumvention or modification to let the request complete. The size of the program work area must be increased either by reassembly or zap to the load module.

T00IJ125I Unallocated Private - Below *BBB* (*BP*) Above *AAA* (*AP*)

Reason: This message is issued in response to a request to display unallocated storage

BBB Amount of storage (in 1024 byte units) remaining unallocated below the 16 MB line.

BP Percentage of unallocated storage below the 16 MB line.

AAA Amount of storage (in 1024 byte units) remaining unallocated above the 16 MB line.

AP Percentage of unallocated storage above the 16 MB line.

Action: None. This is an Informational message.

T00IJ126I *pgm_name* Being attached with DPMOD *dprty*; SP(0) *un*Shared *rsapfval*

Reason: A demo-only message that displays information about the ATTACH being done.

pgm_name	Name of the program module being attached.
dprty	Dispatching priority given to the new task (decimal).
un	Either a null string or the prefix UN. This is used to designate whether subpool zero is shared or unshared by the attached task.
rsapfval	String indicating the setting of RSAPF parameter used on the ATTACH.

Action: None. This is an Informational message.

T00IJ127I parm length=*parm_len* Parm=*parm_name*

Reason: A test-only message that displays information about the parameter string being passed to the attached program.

parm_len	Decimal length of the parm field being passed.
parm_name	Parameter character string.

Action: None. This is an Informational message.

T00IJ128I *pgm_name* terminated: ECB = *comp_code*

Reason: The attached program completed.

pgm_name	Name of the attached program that is completing.
comp_code	Hexadecimal Event Control Block (ECB) completion code.

Action: None. This is an Informational message.

T00IJ129I Tasklib DD did not open successfully

Reason: The data sets allocated to the TASKLIB DD statement did not open properly.

Action: Examine the data sets allocated to TASKLIB DD and check for improper or unlike data set characteristics.

T00IJ130I Tasklib DD not present, attach bypassed

Reason: The TASKLIB keyword was specified on the attach command. A TASKLIB DD was not present from which to load the attach program.

Action: Reissue the command without the TASKLIB keyword or allocate a TASKLIB DD and retry the original command.

T00IJ131W MEM *addr* | * <Len(*NNN*) | Hexl(*NNN*)><Mod(*Name*)>

Reason: A /MEM command was entered with invalid syntax. The correct syntax is displayed.

Action: Correct the command syntax and retry.

T00IJ132W Storage *start-end* is not addressable

Reason: The /MEM command was used to display memory and storage from location *start* to location *end* was not available

Action: Correct the command syntax and retry.

T00IJ134W Module name *mod_name* not known

Reason: A MODULE name was entered but not found in the system.

Action: Appropriate action depends on the intent of the command. If this was a confirmation of module availability, the MEM request is done; if the module was expected to be available, this may be caused by a mistype or an actual missing module.

T00IJ135W Memory modification is not supported

Reason: Support for memory alteration via the MEM command was not implemented.

T00IJ140I Storage ceiling set to *bb*% (below), *aa*% (above)

Reason: The storage ceiling has been set to *bb*% for virtual storage allocation below the 16 MB line, and to *aa*% for storage above the line. When these thresholds are exceeded, new connection requests are rejected until the storage utilization is reduced. These values default to 90% below and 75% above, and can be modified with the MAXSTGPCT parameter of the IFSPARMS statement in the IJTCFG*xx* configuration parameter.

Action: It may be necessary to increase the REGION parameter on the EXEC statement of the started task, or to provide an exit, such as IEFUSI, to increase the storage limits.

T00IJ141E TRACE table not changed, ITIME ALLOC return code *ret_code*

Reason: Could not allocate an ITIME block to defer the deletion of the current trace table. The current trace table (and size) remains unchanged.

Action: Call Customer Support.

T00IJ150I ARM processing [enabled | disabled]

Reason: ARM processing for the address space was enabled or disabled.

Action: None. This is an Informational message.

T00IJ151I ARM element elementname status

Reason: The named ARM element's status has changed to *status*.

Action: None. This is an Informational message.

T00IJ152I Waiting for ARM predecessors

Reason: The address space is waiting for its ARM predecessors to be ready.

Action: None. This is an Informational message.

T00IJ153I All ARM predecessors ready

Reason: All of the address space's ARM predecessors are ready.

Action: None. This is an Informational message.

T00IJ154I ARM WAITPRED is not necessary

Reason: It is not necessary for an ARM WAITPRED function to be issued.

Action: None. This is an Informational message.

T00IJ160W ARM action failed: Element *elementname* already *status*

Reason: The named ARM action was not completed, because the named ARM element is already in the given status.

Action: If the message is the result of an ARM command, verify that ARM is not in the given status before issuing the command. If the message is not the result of an ARM command, use ARM commands to move ARM to the required status.

T00IJ161W ARM action failed: Element *elementname* not *status*

Reason: The named ARM action was not completed, because the named ARM element is not yet in the proper status.

Action: If the message is the result of an ARM command, verify that ARM is in the given status before issuing the command. If the message is not the result of an ARM command, use ARM commands to move ARM to the required status.

T00IJ163W At least one ARM predecessor timed out

Reason: One or more of the address space's ARM predecessors did not become ready in the required time.

Action: Correct the situation that caused the ARM predecessor(s) to fail to become ready.

T00IJ164W ARM element *elementname* deregistered due to an error

Reason: The named ARM element was deregistered because ENF detected an error in the CDS (for example, the disk volume containing the CDS is not online).

Action: Correct the situation that caused the CDS error.

T00IJ170E ARM action failed: ARM has not been enabled

Reason: The named ARM action was not completed, because ARM processing has not yet been enabled.

Action: If the message is the result of an ARM command, verify that ARM is enabled before issuing the command. If the message is not the result of an ARM command, use ARM commands to move ARM to the required status.

T00IJ171E Invalid request code *code* passed to ARM request handler

Reason: The ARM request handler was passed an invalid request code.

Action: Contact Customer Support.

T00IJ172E ARM action failed: Return code *code*, reason code *code*

Reason: The named ARM action was not completed.

Action: Contact Customer Support.

T00IJ173E ARM registration failed: Invalid element name *elementname*

Reason: The ARM element name provided is invalid.

Action: Correct the ARM element name.

T00IJ174E ARM registration failed: Address space being cancelled

Reason: ARM registration was not completed, because the address space is being cancelled.

Action: Retry the operation after the address space is restarted.

T00IJ175E ARM registration failed: Element name *elementname* is being used by another address space

Reason: ARM registration was not completed, because the named ARM element name is in use by another address space.

Action: Change the ARM element name to a value that is unique across the sysplex.

T00IJ176E Invalid request code *code* passed to ARM ISRB

Reason: The ARM ISRB was passed an invalid request code.

Action: Contact Customer Support.

T00IJ177E Invalid request on ARM command

Reason: An ARM command was issued, specifying an invalid request.

Action: Reissue the command with a valid request.

T00IJ178E Missing request on ARM command

Reason: An ARM command was issued without a request.

Action: Reissue the command with a request.

T00IJ190I ENF events are being monitored for ARM

Reason: The address space is monitoring ENF events for ARM.

Action: None. This is an Informational message.

T00IJ191I ENF events are no longer being monitored for ARM

Reason: The address space is no longer monitoring ENF events for ARM.

Action: None. This is an Informational message.

T00IJ192W Duplicate ENF listen issued; ignored

Reason: A LISTEN request was issued, but the address space is already monitoring ENF events.

Action: If the message is the result of an ARM LISTEN command, verify that ARM is not monitoring ENF events before issuing the command. If the message is not the result of an ARM command, use ARM commands to move ARM to the required status.

T00IJ193E ENF request failed: Return code *code*

Reason: The named ENF request failed.

Action: Contact Customer Support.

T00IJ800I Strobe collection complete. Samples: *aaa* Found: *bbb* Unknown: *ccc*

Reason: This message is displayed by the STROBE command when all sampling is done, and the report is about to be displayed.

Action: None. This is an Informational message.

T00IJ801D Module TCB SRB Percent Size <---TCB distribution---> <SRB distribution --->

Reason: This message displays the report header for the STROBE command.

T00IJ802D *modname tcb_samps srb_samps (p_of_samps) tcb_q1 tcb_q2 tcb_q3 tcb_q4 srb_q1 srb_q2 srb_q3 srb_q4*

Reason: This message is generated by the STROBE command and gives report detail information

modname	Module name.
tcb_samps	Occurrences in TCB mode.
srb_samps	Samples in SRB mode.
(%_of_samps)	Percentage of all samples.
tcb_q1	Occurrences in first quarter of the module in TCB mode.
tcb_q2	Occurrences in second quarter of the module in TCB mode.
tcb_q3	Occurrences in third quarter of the module in TCB mode.
tcb_q4	Occurrences in fourth quarter of the module in TCB mode.
srb_q1	Occurrences in the first quarter of the module in SRB mode.
srb_q2	Occurrences in the second quarter of the module in SRB mode.
srb_q3	Occurrences in the third quarter of the module in SRB mode.
srb_q4	Occurrences in the fourth quarter of the module in SRB mode.

T00IJ810W Strobe capture is still alive: *aaa* of *bbbbb*

Reason: This message is a STROBE reponse to a STATUS request.

T00IJ811W Strobe processing is not active

Reason: A STATUS or OFF command was issued, but STROBE was not active.

T00IJ812I Strobe processing complete, RC=*ret_code*

Reason: This is the normal termination message for the STROBE command.

Action: None. This is an Informational message.

T00IJ803D (*HEXAddr*) *MODname* +*offset*

This message is displayed by the STROBE command for the DETAIL(*name*) option. Every hit is displayed in addition to the usual report, so a considerable number of messages can be expected.

HEXAddr	The PSW address when the sample interrupt hit.
MODname	The module name, as specified in the DETAIL(*name*) option.
+*offset*	The offset from the module load point.

Chapter 11

T00VE*nnn* IPCS Trace Messages

This chapter describes messages issued for the IPCS Trace Message facility. These messages include T00VE001 through T00VE999.

T00VE*nnn*

T00VE081S DUMP ANALYSIS CALL TO IPCS STORAGE ACCESS SERVICE FAILED

Reason: A call to the IPCS routine to access storage failed.

Action: Contact Customer Support.

T00VE201W Unable to access *block* at address *address*

Reason: The requested control block was not in storage.

Action: Contact Customer Support.

T00VE202S Module *module* unable to process *parm* 'data'

Reason: A sub-parameter error was detected by *module*.

Action: Contact Customer Support.

T00VE203W Control Block error *block* at address *address*

Reason: The control block at the specified address was invalid.

Action: Contact Customer Support.

T00VE204I Control block *block1* address for *block2* is null

Reason: Control block *block1*'s pointer to *block2* is zero.

Action: Contact Customer Support.

T00VE205I DUMP ANALYSIS: HANDLER *module* RC out of range

Reason: A module was invoked to format a control block and returned with a return code out of range.

Action: Contact Customer Support.

T00VE206E DUMP ANALYSIS: END OF PROGRAM TABLE

Reason: A search of the program table for a specific module was made but the end of table occurred without finding the module. The module cannot be loaded because its entry cannot be recorded in the program table.

Action: Contact Customer Support.

T00VE207E (IPVD) Error LOADing *module* R15 *rc* R01 *code*

Reason: An error occurred from a LOAD macro.

Action: Make sure the LINK library is in either the STEPLIB or linklist.

T00VE310E CTRACE FILTER UNABLE TO PARSE OPTIONS

Reason: An error from IKJPARS occurred while parsing the options passed via ((*options*)).

Action: Make sure all the options are valid.

T00VE311E CTRACE FILTER - ERROR PARSING GROUPS

Reason: The *group_id* is too long, the filter data is too long, or extraneous data after the filter specification was entered.

Action: Verify that the (*group_id,'filter'*) specification is correct.

T00VE312E CTRACE FILTER - INVALID GROUP NAME *group_id*

Reason: A *group_id* was entered that is invalid.

Action: See your product documentation for valid names.

T00VE313E CTRACE FILTER - TOO MANY SEGMENTS IN CTE

Reason: More trace segments are present than are allowed.

Action: Contact Customer Support.

T00VE314E CTRACE FILTER - UNABLE TO LOAD *module*

Reason: An error occurred trying to load *module*.

Action: Make sure the LINK library is in either the STEPLIB or linklist.

T00VE315E CTRACE FILTER - NO STORAGE FOR PROGRAM TABLE

Reason: An error occurred trying to obtain storage.

Action: Inspect the REGION size for TSO.

T00VE318S CTRACE EXECUTE FILTER FAILURE: *errnum*

Reason: Error *errnum* occurred executing the filter program.

Action: Contact Customer Support.

T00VE319S CTRACE GENERATE FILTER FAILURE: RC=*rc* CC=*cc* *func*

Reason: An error occurred generating the filter. *func* is either STOR or PARS for STORAGE or IKJPARS respectively.

Action: If *func* is PARS, make sure the filter parameters are valid.

T00VE320S CTRACE FILTER - UNABLE TO ACQUIRE NECESSARY STORAGE

Reason: An error occurred trying to obtain storage.

Action: Inspect the REGION size for TSO.

T00VE401W T03PTIPP: Unknown SEGMENT Type *segment_type* CTE *cte_id*

Reason: An unknown trace segment was encountered.

Action: Contact Customer Support.

T00VE402S T03PTIPP: SEGMENT overflow CTE *cte_id*

Reason: A segment was found that extended beyond the buffer.

Action: Contact Customer Support.

T00VE403S T03PTIPP: Unknown CTE *cte_id*

Reason: An unknown component trace entry was encountered.

Action: Contact Customer Support.

Chapter 12

T01AP*nnn* API Common Application Support Routine Messages

This chapter describes messages issued by the API Common Application Support Routines. These messages include T01AP000 through T01AP999.

T01AP*nnn*

T01AP001I Transport Endpoint Error

Reason: A TPL function has failed and the user has called TERROR to format a description of the status at the time of the failure. See T01AP016D for details on interpreting this failure status description. Additional messages may follow to further describe the error.

Action: None. This is an Informational message.

T01AP002I *function* error in module RC=*ret_code*, function RC=*func_rsn*, APCBERRC=*err_code*

Reason: An AOPEN call failed. Refer to the *API Programmers Guide* for general information, and read "API Return Codes" for information about APCB diagnostic errors.

function	The specified function.
module	The routine containing the function.
ret_code	The return code in register 15 returned by the function.
func_rsn	The function return code returned by the caller of the function.
err_code	The error code and diagnostic code from the APCBERRC and APCBDGNC fields.

Action: None. This is an Informational message.

T01AP003E *function* error in *mod_name* RC=*ret_code*, function RC=*func_reason*. Hex dump of TPL follows:

Reason: A function failed. The TPL for the function call is dumped after the message. Read your API programmer's manual for general information, and read "API Return Codes" for information about TPL diagnostic errors.

function	Specified function.
mod_name	Routine containing the function.
ret_code	Return code in register 15 returned by the function.
func_reason	Function return code returned by the caller of the function.

T01AP004E Load failed for CHAR/DBCS translation module *mod_name*

Reason: The attempt to load the specified module, *mod_name*, as single or double-byte character set translation table failed.

Action: Verify that the character set name has been created and is a valid table.

T01AP005E Validation failed for CHAR/DBCS translation module *mod_name*

Reason: An invalid character set translation table (*mod_name*) was specified or defined. This table may have been created incorrectly or was not created for the specified transopt (single or double).

Action: Verify character set matches the translation mode.

T01AP006I *request* SVC99 error=*error* R00=*reason* DSN=*dataset_name*

Reason: A routine requested PDYNAL to perform the specified request. This involves a call to SVC 99 (dynamic allocation).

request Specified request, which can be one of these values:

ALLOCATE	Allocate a DASD data set.
SYSIN	Allocate a file to the JES job stream reader.
SYSOUT	Allocate a file to the JES output spool space.
FREE	Free a DASD, SYSOUT, or SYSIN file.
RENAME	Rename a DASD data set.
MEMCHK	Check to see if a member exists in a PDS.
MEMDEL	Delete a member in a PDS.
MEMREN	Rename a member in a PDS.

error DAIR error code returned from the SVC 99 call.

reason Return code from the PDYNAL routine, which can be one of these:

0	Success.
4	Failure.
8	Data security violation.
C	IDYNAL bug.
10	ICT error.

dataset_name Data set name .

Action: Examine the DAIR error code; it should direct your action. If the PDYNAL return code is other than FOUR, contact Customer Support. Save the T01LOG, JES logs and any dumps produced by the job.

If the message text reads: T01AP006I ALLOCATE SVC99 Error=FFE1 R00=00000004, then the FFE1 is not a valid DAIR error code. It is a special code that indicates that the data set has been migrated; and that the application has not requested that the dataset be recalled.

T01AP007E PCORE get failed for ULPP, module=*mod_name*

Reason: A User Level Protocol (ULP) cannot allocate the memory space necessary for its work area.

Action: Save all output from the job. Check the WTO and JOB logs for related messages. Check the ACSNAP DD for any related dump. Use ACTEST to issue the psnap command to print all of the product's storage into the ACSNAP DD. If there is a proliferation of the same kind of storage, a ptask may be allocating storage in a loop.

Retry the operation first on the current version of the product. If it fails again, bring the product down. Raise the region size on the job. Restart the product and retry the operation again. If the error recurs, contact Customer Support.

T01AP008D GETHBDNR failed due to a bad parm (hex dump of parm field below)

Reason: An internal call to DNR could not resolve a host name, alias name, subsystem name, mail route, and so on. A bad parameter list was passed to subroutine GETHBDNR. This indicates a programming error. The two passed parameters are dumped followed by 20 bytes of argument data.

Action: Contact Customer Support. Save the output from the TCP base product job.

T01AP009D GETHBDNR DIRSRV call failed (hex dump of DPL below)

Reason: An internal DNR call failed for a request to resolve a host name, alias name, subsystem name, mail route, etc. The DPL for the DIRSRV request is dumped after the message. This can be a configuration, initialization, or user error.

If the message occurred at system startup it could be due to a temporary DNR/ACP initialization coordination problem.

Action: Use this message, the DNR logs (as long as TRACE and INTERNALTRACE were set in configuration member DNRCFG*xx*) and the *Programmers Guide* to diagnose this problem. There may be related error messages in the T01LOG. The DPL contents and the messages in the DNR logs should guide your actions.

T01AP010E API ULPP initialization parm error, module=*mod_name*

Reason: An invalid parameter was encountered during initialization of an application layer program.

Client or server initialization is terminated.

Action: This error could be temporary or indicative of other problems within the product. If the problems persists, the ACSNAP log should be checked for dumps. The T01LOG and JES log of the job should be saved. Contact Customer Support.

T01AP011I #DSKIO ACFGAT not found - using default

Reason: The GAT supplied on the SMTP statement was not found.

Action: Add the GAT to the startup parameters.

T01AP012I #DSKIO LRECL is greater than 255

Reason: The LRECL on the GAT for the SMTP statement was greater than 255.

Action: Lower the LRECL below 255.

T01AP013E SMTPEX1 PSW=*PSW_value*, R12=*csect_loc* CMP=*aaaaaaaa*

Reason: The outbound mailer program USMTP terminated with ABEND *aaaaaaaa*. The PSW at the time of the ABEND is *PSW_value*. R12 field *csect_loc* is the location of csect SMTPEX1.

Action: The TCP base product takes a dump of the USMTP ptask and associated resources. Contact Customer Support. Save all output from the TCP base product job or started task.

T01AP014W *module* IBUF and/or OBUF values were adjusted to comply with provider limits

Reason: Telnet server configuration specified API buffer parameters that were larger than the API provider could support. The API has negotiated lower numbers, followed by a hex dump of the TPL. *module* is the module issuing the message.

T01AP015E API error in *module*, PTA *address*, TPL follows:

Reason: An error occurred during an API T-function operation. The program in error and the PTA address are listed in the T01AP016D message. *module* is the module issuing the message; address is the PTA address of the pseudo-task issuing the message.

T01AP016D JOB *job_name* STEP *step_name* APPL *appl* USER *user_name*
TPL tpl_addr APCB apcb_addr QLSN listen_count
SEPM STAT sock_state TSTAT ep_state (state_code) ACTIV tpl_activ FLAGS tpl_flags
TPL ACTCD ret_act_code ERRCD spec_err_code (spec_code) DGNCD diag_err_code diag_code)
TPL EPID ep_id ECBXR ecb/exit_addr OPTCD option_codes
TPL PARM1 parm_1 PARM2 parm_2 PARM3 parm_3
TPL ADBUF adbuf_addr DABUF dabuf_addr OPBUF opbuf_addr
TPL ADLEN adbuf_len DALEN dabuf_len OPLEN opbuf_len

Reason: This message is generated in a GETMAIN area in WTO format by the TERROR VERBATIM function. If an actual WTO is issued by the application program, this message may be seen on the system console (depending on the routing code(s) set in the message by the application program).

job_name	JOB name.
step_name	STEP name.
appl_id	Application ID from APCBAPPL.
user_name	User name from TCEPUID.
tpl_addr	Address of TPL.
apcb_addr	Address of APCB.
listen_count	Listen count from SEPM, in hex.
sock_state	Socket state from SEPM, in hex.
ep_state	Endpoint state from SEPM, in hex.
state_code	Endpoint state, in English.
tpl_activ	TPL active flag.
tpl_flags	TPL flags.
rec_act_code	Recovery action code, in hex.
spec_err_code	Specific error code, in hex.
spec_code	Specific error code, in English.
diag_err_code	Diagnostic error code, in hex.
diag_code	Diagnostic error code in English or module name.

ep_id	Endpoint ID.
ecb/exit_addr	ECB/EXIT routine address.
option_codes	TPL option codes, in hex.
parm_1	Contents of TPLPARM1.
parm_2	Contents of TPLPARM2.
parm_3	Contents of TPLPARM3.
adbuf_addr	ADBUF buffer address, from TPLADBUF.
dabuf_addr	DABUF buffer address, from TPLDABUF.
opbuf_addr	OPBUF buffer address, from TPLOPBUF.
adbuf_len	ADBUF buffer length, from TPLAPLEN.
dabuf_len	DABUF buffer length, from TPLDALEN.
opbuf_len	OPBUF buffer length, from TPLOPLEN.

T01AP017D *text*

Reason: A C program, like mail, has written a message to the log. *text* is the text of the message written to the log.

T01AP022E Invalid module *mod_name* in SERVICE statement

Reason: This is an internal logic error. A SERVICE statement with an invalid module name was accepted during startup, but the module could not be found.

Action: Correct the module name in the configuration and contact Customer Support.

T01AP023I *service* waiting for API *macro* to complete

Reason: This is a normal startup message. Some services use the TCP base product API. The specified service issued the specified API session initialization macro. The service waits for the API macro to complete.

service	The service statement defined in the ACPCFG*xx* member.
macro	An API session initialization macro: either AOPEN or TOPEN.

Action: None. This is an Informational message.

T01AP024I *service* API *macro* completed

Reason: This is a normal startup message. Some services use the TCP base product API. The specified service completed an API session initialization macro.

service	The service statement defined in the ACPCFG*xx* member.
macro	An API session initialization macro: either AOPEN or TOPEN.

Action: None. This is an Informational message.

T01AP025W *service API_call* failed; RC= *ret_code* , ERRC= *error_code* , DGNC= *diag_code*

Reason: An API function call failed for a server application, defined in the APPCFG*xx* configuration member as SERVICE *service*. The API function call return code is given as *ret_code*, the error code as *error_code*, and the diagnostic code as *diag_code*. See *Unprefixed Messages and Codes* for a description of API function call return codes and error codes.

This message is followed by T01AP023I, and the API function call is retried after a short interval. If the API function call is successful, message T01AP024I is issued. Message T01AP025W is reissued only if the API function call error or diagnostic codes change.

Action: This may be a transient condition, particularly if the APP task group is run in a different address space from the transport provider. If the condition persists, consult the API function call error codes for the failure reason and take appropriate action.

service	The service statement defined in the ACPCFG*xx* member.
API_call	API function call.
ret_code	Function return code.
error_code	Error code.
diag_code	Diagnostic Code.

T01AP030E Translate service error for table=*tbl_name* - RC=*rc* EC=*ec*

Reason: EBCDIC/ASCII translate has failed to complete successfully.

tbl_name The character set table.

rc The return code.

ec The error code.

The following table describes the return and error codes:

Diagnostic Code	Return Code	Reason
R15=08	R0=0	Invalid parmlist.
R15=08	R0=1	Invalid parmlist; no character set specified.
R15=08	R0=2	Invalid parmlist; invalid source text address.
R15=08	R0=3	Invalid parmlist; invalid source text address.
R15=08	R0=4	Invalid parmlist; invalid source text length.
R15=08	R0=5	Invalid parmlist; invalid destination text address.
R15=08	R0=6	Invalid parmlist; invalid destination text length.
R15=08	R0=7	Invalid parmlist; pcore failed.
R15=08	R0=8	DBCS odd length; DBCS specified odd length.
R15=12	R0=0	Overflow
R15=12	R0=1	Invalid TRANOPT; operation type not specified.
R15=12	R0=2	Network direction; input/output not specified.
R15=12	R0=3	Wrong character set; invalid character set for request.
R15=12	R0=4	Invalid byte; invalid request for char set.
R15=12	R0=5	Bad Table specified; invalid table definition.
R15=12	R0=6	Bad DBCS header; DBCS table header invalid.
R15=12	R0=7	Mixed data error; error processing mixed data.
R15=12	R0=8	Unknown DBCS character; unknown DBCS character.

T01AP032W *Abend* in API user exit routine

Reason: A user-written exit routine ABENDed.

Action: Examine the system for dumps relating to this ABEND and correct the user exit.

T01AP033W WLM Service: *WLM_groupname* WLM Registration | Deregistration Failed; *ret_code*, reason=*reason_code*

Reason: An attempt to WLM register or deregister the TCP/IP stack or a service application with the MVS Workload Manager. An error was encountered when either the WLM service IWMSRSRG to register or IWMSRDRS to deregister was invoked. Refer to *OS/390 V2R5.0 MVS Workload Management Service* for Reason of error and reason codes.

Action: Call Customer Support and report the problem.

WLM_groupname	The WLM Groupname.
Registration | Deregistration	The message specifies registration or deregistration.
ret_code	WLM general return code.
reason_code	WLM reason code.

T01AP034W BindSec WLM function failed for Port=port Jobname=*jobname* rc=*ret_code* reason=*rsn_code*

Reason: A MVS Workload Management service returned a non-zero return code. The service was called as a result of an application binding to a specific port, as defined in the bind security configuration parameters. In the message, function is the WLM function requested, port is the port number bound by the application, job name is the application STC/JOB/TSU name, and *ret_code* and *rsn_code* are the return code and reason code, respectively, from the WLM service.

Action: Use the return code and reason code to diagnose IWMSRSRG service failure. The IWMSRSRG return and reason codes are documented in the IBM publication titled *MVS Programming: Workload Management Services*. If unable to resolve the problem, contact Customer Support.

T01AP041I *server_name* waiting for OMVS Socket function to complete

Reason — The server identified by *server_name* attempted to open a socket to Open Edition/USS but received a temporary error condition. The server will attempt to open the socket every few seconds until the condition is resolved.

Action — If this message persists, save the message log and contact Customer Support.

T01AP043E *server_name* OMVS Socket function failed; Retval=*return_code*, Errno=*err_code* , ErrnoJr=*diag_code*, sd=-1

Reason — The server identified by *server_name* cannot open a socket to Open Edition/USS. The return code, error code and junior diagnostic codes are given. Refer to *Unicenter TCPaccess Communications Server Unprefixed Messages and Codes* for an explanation of these codes. The most likely causes of the failure are security or configuration errors.

Action — If you cannot determine the cause of the error after reviewing the codes returned, contact Customer Support. Save the message log with the server name and all error codes.

T01AP045W 3270-mode message formatted for NVT session on port *port*; client is *ipaddr:port*

Syntax Description: *port* is the port to which the client is connected

ipaddr:port is the IP address and port of the client

Reason — A message was generated containing 3270 data streams for an NVT session.

Action: — Insure that an NVT mode USSTAB is available.

T01AP100I Server *server_name* now listening on port *port_num*

Reason: — The server configured as *server_name* successfully opened its endpoint and is listening on port *port_num.*

Action: — None. This is an Informational message.

T01AP101I Session *sess_num* created on port *port_num* and passed to *module_name*

Reason: — The listener accepted a new connection on port *port_num*, and a new session identified by *sess_num* was created. Control is passed to module *module_name*, as configured in the MODULE() parameter of the SERVICE statement.

Action: — None. This is an Informational message.

T01AP102E Server *server_name* on port *port_num* unable to obtain context block for session *sess_num* on port *port_num*

Reason: The listener accepted a new connection on port *port_num*, and a new session identified by *sess_num* was created, but the listener was not able to get a context block for the new session. The session is terminated.

Action: This is probably due to a virtual storage constraint. Try increasing the region size for the job. If the problem persists, contact Customer Support.

T01AP103E Server *server_name* on port *port_num* call to *service_name* failed with return code *ret_code* and reason code *rsn_code*

Reason: The server identified by *server_name* on port *port_num* is attempting to establish communication with the data space that holds performance and tuning metrics. The service request identified by *service_name* failed. The return code and reason code are listed. Processing continues, but some metrics will not be measured.

Action: Contact Customer Support. Save the message log with the service name, return code, and reason code.

T01AP104I Server *server_name* on port *port_num* shutting down because the transport provider is ending.

Reason: The transport provider for the application is ending, so the server is quiesced. New connections on that port will not be accepted by the server. After a few seconds, the server will reattempt to listen for new connections.

Action: None. This is an Informational message.

T01AP105I Server *server_name* on port *port_num* restarting

Reason: The server identified by *server_name* and *port_num* was quiesced because the transport provider stopped. The server is restarting, and will now attempt to listen for new connections.

Action: None. This is an Informational message.

T01AP106I Server *server_name* on port *port_num* waiting for transport provider

Reason: The server identified by *server_name* and *port_num* cannot open its endpoint. The most likely cause is that the transport provider is unavailable. The server will attempt to open the endpoint every few seconds, until the transport provider becomes available.

Action: None. This is an Informational message.

T01AP107W Server *server_name* on port *port_num request* failed with RC=*ret_code*, ERRC=*error_code*, DGNC=*dgn_code*

Reason: The server identified by *server_name* and *port_num* failed to complete the TLI request identified by *request*. The TLI return code, error code, and diagnostic code are given. The most likely cause of the failure is that the transport provider is unavailable. The server will try to open the endpoint every few seconds, until the transport provider becomes available.

Action: If the message persists after the transport provider becomes available, contact Customer Support. Save the message log with the request name and error codes.

Chapter

13

T01CF*nnn* Configuration Messages

This chapter describes messages for configuration. These include messages T01CF000 through T01CF999.

T01CF*nnn*

T01CF001I Parm stmt: *message*

Reason: This message follows another T01CF*xxx* message and displays the contents of statement identified by the previous message.

Action: None. This is an Informational message.

T01CF002E LU keyword required, member=*mem_name*, command number=*rec_num*, command=*cmn_name*

mem_name Configuration member name.

rec_num Statement location in configuration member.

cmnd_name Failing command statement.

Reason: The LU keyword is required for this command statement. One or more VTAM LUs must be specified with the LU keyword.

Action: Add the LU keyword with valid VTAM LUs to the command statement.

T01CF004I No associated printer defined in TN3270E Pool=*pool_name*, member=*mem_name*, number=*rec_num*, command=*cmnd_name*

pool_name	Name of LU pool.
mem_name	Configuration member name.
rec_num	Statement location in configuration member.
cmnd_name	Command statement.

Reason: A TN3270 LU pool was defined without associated printer LUs.

Action: None. This is an Informational message.

T01CF006E LU syntax for Pool-pool_name and Pool TYPE is invalid, member=*mem_name*, command number=*rec_num*, command=*cmnd_name*

mem_name	Configuration member name.
rec_num	Statement location in configuration member.
cmnd_name	Failing command statement.

Reason: LU keyword syntax specified subparameter inconsistent with the LU pool type. The most likely cause is the specification of an associated printer for a pool TYPE other than TN3270E.

Action: Correct the LU parameter syntax for LU pool TYPE.

T01CF007I LUPOOL initialization successful

Reason: Informational message indicating that the LU pool configuration member was processed without error and that the LU pool facility has been activated.

Action: None. This is an Informational message.

T01CF008E LUPOOL initialization failed

Reason: A problem was encountered while processing the APPLU*xx* LU pool configuration member. Message(s) display at the operator console prior to this message describing the problem(s). The LU pool facility is not activated.

Action: Correct the problem with LU pool configuration member APPLUP*xx* and issue the REFRESH LUPARM(*mem_name*) command to initialize the LU pool facility.

T01CF009E Parm validation error - member = *mem_name*, command number = *cmnd_num* command = *cmnd_name*, keyword = *kywd*

mem_name	Member name.
cmnd_num	Logical number of the statement.
cmnd_name	Statement containing the syntax error.
kywd	Keyword in error.

Reason: An APPLUP*xx* command statement contains a syntax error. The parameter for the keyword specified is invalid. The LU pool facility is not activated.

Action: Correct the problem with the command statement and issue the REFRESH LUPARM(*mem_name*) command to initialize the LU pool facility.

T01CF010E Unknown pool name=*pool_name* for LURULE - member = *mem_name*, command number = *cmnd_num*

pool_name	Pool name.
mem_name	Member name.
cmnd_num	Logical number of the statement.

Reason: The APPLUP*xx* member contains an LURULE statement whose POOL keyword specifies a LU pool that was not defined by an LUPOOL statement. The LU pool facility is not activated.

Action: Include an LUPOOL statement defining the pool and issue the REFRESH LUPARM(*mem_name*) command to initialize the LU pool facility.

T01CF011E Duplicate LUPOOL=*pool_name* - member = *mem_name*, command number = *cmnd_num*

pool_name	Pool name.
mem_name	Member name.
cmnd_num	Logical number of the statement

Reason: The configuration member being parsed contains multiple LUPOOL statements that specify the same pool name. The LU pool facility is not activated.

Action: Correct the problem with the LUPOOL command statement and issue the REFRESH LUPARM(*mem_name*) command to reinitialize the LU pool facility.

T01CF012E Duplicate LUNAME *lu_name* for LUPOOL *pool_name* - member = *mem_name*, command number = *cmnd_num*

lu_name	LU name.
pool_name	Pool name.
mem_name	Member name.
cmnd_num	Specified command.

Reason: The same LU name was specified multiple times for the LU keyword in either an LUPOOL or LURULE statement. The duplicate LU names were ignored and the LU pool facility was activated.

Action: Correct the problem with the command statement and issue the REFRESH LUPARM(*mem_name*) command to reinitialize the LU pool facility.

T01CF013E No LURULE(S) specified - member = *mem_name*

Reason: No LURULE statements were specified in the member (APPLUP*xx*). At least one rule is required. The LU pool facility is not activated.

Action: Add LURULE statement(s) to the LU pool configuration member and issue the REFRESH LUPARM(*mem_name*) command to reinitialize the LU pool facility.

T01CF014W Append keyword obsolete, LUnames ignored - member=*mem_name*, command number=*rec_num*, command=*cmnd_name*

mem_name	Member name.
rec_num	Statement location in configuration member.
cmnd_num	Specified command.

Reason: The APPEND keyword is no longer supported. LUs are not usable.

Action: Change append pool to LURULE pool chaining.

T01CF015E Invalid value found in keyword field.

Reason: The value coded in the specified keyword contained invalid characters. Some require hexadecimal characters (A-F, zero-nine). Others require decimal characters (zero-nine).

Action: Correct the value in the configuration parameters and restart the APP task group.

T01CF016E Terminal LU count and Printer LU count not equal, member=*mem_name*, command number=*rec_num*, command=*cmnd_name*

mem_name	Member name.
rec_num	Statement location in configuration member.
cmnd_num	Failing command statement.

Reason: A TN3270E LU pool was defined with some terminal LUs without associated printer LUs. If more than one associated printer is specified, all terminals must have an associated printer defined.

Action: Specify an associated printer LU for each terminal LU, or specify only one printer LU for all terminals.

T01CF017E Command security requested with global security off

Reason: During configuration parsing of the HOST statement, the XSEC operand list resulted in turning off global security and still requesting command security. This is an invalid combination. If you want command security, enable global security. If you do not want global security, turn command security off.

Action: Correct the value in the configuration parameters and restart the APP task group.

T01CF018E Keyword=keyword1 value=kywd1_value incompatible with keyword=keyword2 value=kywd2_value, member=mem_name command=cmnd_name number=rec_num

keyword1	First keyword containing the incompatible parameter value.
kywd1_value	Value specified for the first keyword.
keyword2	Second keyword containing the incompatible parameter value.
kywd2_value	Value specified for the second keyword.
mem_name	Configuration member name.
cmnd_name	Statement containing the incompatible keyword values.
rec_num	Logical record number of the statement containing the incompatible keyword values.

Reason: Incompatible values were specified for two keywords of a statement.

Action: Review the documentation for the configuration statement and specify compatible values for the keywords.

T01CF019W PORTUSE keyword obsolete, all ports available - member=mem_name, command=cmmd_name

Reason: The PORTUSE keyword for the TCP and UDP commands in the TCPCF*xx* configuration is obsolete. This message appears if a configration from a previous release of TCPaccess is used. The Bind Security facility by default allows all TCP and UDP ports to be available to network server applications for listening. PORTRULES configured in the BINDSEC configuration parameter file can be used to control access to specific ports.

mem_name Name of configuration member.

cmnd_name Command is either TCP or UDP in configuration member.

Reason: Remove the PORTUSE keyword from the TCP and UDP commands in the TCPCFG*xx* file eliminated the message from the T01LOG

T01CF021E Missing required GAT statements. flag = *flag*

Reason: One or more required GAT statements were not found or were not successfully parsed.

The flag byte (*xx*) represents the composite of the required GAT statements that were successfully parsed. To determine which GAT statement is missing, subtract the flag (*xx*) from hexadecimal x'FC' – the difference is the composite hexadecimal values of the missing statement(s). Use the following table to determine which statement or statements are missing:

Flag Byte	GAT Statement
x'80	SOURCE
x'40'	CARDS
x'20'	FORTRAN
x'10'	OBJECT
x'08'	LOADLIB
x'04'	PRINT

Action: Correct the value in the configuration parameters and restart the APP task group.

T01CF022E Unresolved *type* name *name* in member *mem_name*.

Reason: A name field is referenced that is not resolved elsewhere in the configuration.

type	Type of name that was not resolved.
name	Name that was not resolved.
mem_name	Member name of the configuration being parsed.

Action: Correct the value in the configuration parameters and restart the APP task group.

T01CF023E *stmt* Statement, *#stmt_num* out of sequence in member *mem_name*.

stmt	Statement that is out of place.
stmt_num	Statement number.
mem_name	Member name of the configuration being parsed.

Reason: A statement appears out of place.

Action: see the *Customization Guide* for a reason of statement sequence rules. Correct the value in the configuration parameters and restart the APP task group.

T01CF024I MEDIA *med_name* has been built and anchored

Reason: MEDIA *med_name* is available.

Action: None. This is an Informational message.

T01CF026E Too many operands in *kywd* keyword.

Reason: The configuration statement was coded with more operands than the keyword *kywd* defines.

Action: Correct the value in the configuration parameters and restart the APP task group.

T01CF027E Duplicate *kywd name* found in *stmnt* statements.

kywd Keyword.

name Preexisting name defined by the keyword.

stmnt Command statements with the duplicate keywords.

Reason: A keyword defined a preexisting name.

Action: Correct the value in the configuration parameters and restart the APP task group.

T01CF028E *kywd* keyword is required.

Reason: A statement was encountered with keyword *kywd* that must be referenced by a previous statement.

Action: Read the your customization guide for an Reason of statement sequence rules. Correct the value in the configuration parameters and restart the APP task group.

T01CF029W The *service optional* port assignment variables will not be updated

Reason: The service optional port assignment variables will not be updated by the UPDATE operation command.

T01CF030E Too many *stmnt*'s in configuration

Reason: The maximum number of *stmnt* statements has been exceeded.

Action: Reduce the number of *stmnt* statements and restart.

T01CF031E STORAGE obtain failure RC in *kywd* keyword of statement

Reason: A STORAGE OBTAIN failed with RC=TC processing keyword of statement.

Action: Increase available storage.

T01CF032W The IP statement GATED variable will not be updated

Reason: The update will not be performed when the result of an Operator command.

T01CF040E DEFAULT keyword found in more than one APPL/TELNET statement.

Reason: More than one APPL entry has been specified as the default service entry.

Action: Correct the value in the configuration parameters and restart the APP task group.

T01CF041E Improper placement of TELNET statement.

Reason: The TELNET statement is out of sequence. This statement must precede all other APPL statements.

Action: Insert the TELNET statement in the configuration deck and restart the APP task group.

T01CF042E *parm* not permitted on the DEFAULT APPL statement.

Reason: The specified parameter is invalid on an APPL statement that has been designated as the default service.

Action: Correct the error and restart the APP task group.

T01CF043E *parm* on *stmnt* statement for service *service* not defined on any TERMPROF statement.

Reason: The variable parameter identifies the group name of the undefined TERMPROF statement and verb identifies the statement (APPL or TELNET) on which the error occurred. The TERMPROF value in an APPL or TELNET statement must match the GROUP value in a TERMPROF statement.

Action: Correct the error and restart the APP task group.

T01CF044E *parm* parameter is required on the TERMPROF statement.

Reason: A required parameter was omitted on the TERMPROF statement. The variable parameter identifies the name of the required TERMPROF parameter.

Action: Correct the error and restart the APP task group.

T01CF045E BUFFERSIZE parameter on TERMPROF statement is invalid.

Reason: The BUFFERSIZE parameter is either zero or greater than 8K.

Action: Correct the error and restart the APP task group.

T01CF047E Syntax error on LU parameter *luparm* in *stmnt* statement, in member *mem_name*

luparm LU subparameter name.

statement Specified statement.

mem_name Configuration member.

Reason: A syntax error was detected while processing an LU subparameter. This message is produced only when the LU parameter in an LUPOOL or LURULE statement exceeds 32,767 and one of these conditions exists:

- The LU name exceeds eight characters
- The LU name contains an illegal character

The first character in the LU name is not alphabetic or $, #, @.

Action: Correct the value in the configuration parameters and restart the APP task group.

T01CF048EI Maximum number of LU names exceeded in *stmnt* statement, member *mem_name*.

stmnt Specified statement.

mem_name Configuration member.

Reason: The maximum number of LUNAMES has been exceeded in statement of member.

T01CF049EI Media *media_name* already contains the ROUTE entry for *ipaddr1* via router *ipaddr2*

media_name The media name, identified by the NAME( ) keyword in the MEDIA statement.

ipaddr1 The DEST IP address, in dotted notation.

ipaddr2 The ROUTER IP address, in dotted notation.

Reason: A duplicate ROUTE definition is encountered during the execution of the UPDATE command.

T01CF051EI Parsing module *mod_name* not Found.

Reason: A required parsing module (*mod_name*) cannot be loaded. APP task group initialization does not complete successfully.

Action: Report this error to Customer Support.

T01CF052EI *stmnt* statement not processed - parsing routine *routine* missing.

Reason: A statement (*stmnt*) could not be processed because its parsing *routine* could not be located. The statement is ignored and APP task group initialization proceeds normally.

Action: Report this error to Customer Support.

T01CF053WI Redundant TERMPROF statement(s) detected for *appl* application

Reason: For 3767 linemode applications, only one TERMPROF statement is permitted. All TERMPROF statements except the first one on the application's TERMPROF group are ignored. *appl* is the name of the application on which the error occurred.

This is a warning only. APP task group initialization proceeds normally.

T01CF054E USSTAB option not processed - *tbl_name* USS table is not available

Reason: The USSTAB specified on the SERVICE statement could not be loaded or the default IBM USS Table (ISTINCDT) could not be loaded. In either case, configuration processing continues as if no USSTAB option was specified on any SERVICE statement.

Action: Make the assembled and linked USS Tables available to the job.

To do this:

- Add the data set for the tables to a library referenced in the JOBSTEP/STEPLIB DD in the start-up JCL
- Alternatively, you can put the DSN for the USS Tables in the linklist

T01CF055E USSTAB option not processed - format for *tbl_name* USS table is invalid

Reason: Either the USS Table specified on the SERVICE statement or the default IBM USS Table (ISTINCDT) that was loaded is not in a recognized format for USS Tables.

The macros to create USS Tables generate a hexadecimal X'BD' in the first byte of the table. The named table that was loaded did not contain X'BD' in the first byte.

Action: Make sure that the named USS Table load module was in fact created with the correct macros. For details, refer to the current version of the *VTAM Resource Definition Reference*, published by IBM.

T01CF058EI Missing POOLDEF statement for pool *pool_name* in *member*

Reason: Pool header for stated *pool_name* not found.

Action: Add appropriate POOLDEF and restart.

T01CF059EII *stmnt* statement not valid with MEDIA type *med_type*

Reason: *stmnt* statement with MEDIA type *med_type* is an invalid combination.

Action: Correct configuration parameters and restart.

T01CF065WI The default USS table *tbl_name* is not available or is invalid

Reason: The default USS table *tbl_name* not found or invalid.

Action: Start the APP task group.

T01CF069EI Operator command parameter parse error - command = *cmnd_name*

Reason: An APP operator command (*cmnd_name*) contains a syntax error. The command was most likely the REFRESH LUPARM(*mem_name*) command to refresh the LU pool facility with a new configuration member. The old LU pool configuration is not updated and remains operational.

Action: Reissue the APP command with correct syntax.

T01CF070E Operator command required keyword missing - command = *cmnd_name*, missing keyword = *kywd*.

cmnd_name	Specified operator command
kywd	Missing keyword

Reason: The specified APP operator command is missing the specified keyword. The command was most likely the REFRESH LUPARM(*mem_name*) command to refresh the LU pool facility with a new configuration member. The old LU pool configuration is not updated and remains operational.

Action: Reissue the APP command with correct syntax.

T01CF071EI SMTP configuration definition statement missing; port = *port_num*

Reason: A parameter list SERVICE statement defining a well-known port (*port_num*) for server or client SMTP was established, but, no SMTP configuration statement for this port was found.

Action: Verify configuration statements within APPCFG*xx* parameter member:

- Remove the SERVICE statement
- Alternatively, define an SMTP statement for the preceding port

T01CF072WI Append ignored, *pool_name* not previously defined - member = *mem_name*, command number = *cmnd_num*

Reason: The append was ignored because the LUPOOL *pool_name* is undefined.

T01CF080RI USSTAB <*tbl_name*> successfully refreshed for APP task <*task#*>

Reason: The REFRESH USSTAB (*tbl_name*) TASK (*task#*) operator command was successfully executed.

Action: None required

T01CF081WI Refresh of USSTAB failed, APP task <*task#*> inactive.

Reason: Negative response to REFRESH USSTAB (*tbl_name*) TASK (*task#*) operator command. The APP task <*task#*> was not fully initialized or is in the process of stopping.

Action: Wait for the APP task to complete initialization or restart the APP task <*task#*> if it was stopped

T01CF082W Refresh of USSTAB failed, <*tbl_name*> not found

Reason: Negative response to REFRESH USSTAB (*tbl_name*) TASK (*task*#) operator command. <*tbl_name*> was not found in the JOBLIB/STEPLIB or linklisted library

Action: Research required to determine what happened to the USS table.

T01CF083WI Refresh of USSTAB failed, invalid format in <*tbl_name*>

Reason: Negative response to REFRESH USSTAB (*tbl_name*) TASK (*task*#) operator command. <*tbl_name*> has invalid format (should start with X'BD').

Action: Research required to determine what happened to the USS table.

Source: APPSUSST

T01CF084WI Refresh of USSTAB failed. <*tbl_name*> not used in APP task <*task#*>

Reason: Negative response to REFRESH USSTAB (*tbl_name*) TASK (*task*#) operator command. <*tbl_name*> for APP task <*task*#> was not specified in active APPCFG*xx* SERVICE segment.

Action: Check for proper spelling of <*tbl_name*> and APP task <*task*#> was entered correctly.

T01CF101E No matching *stmnt1* statement found for *parm name* specified on *stmnt2 stmnt*

Reason: The configuration statement *stmt2* specified a *parm* parameter naming the *stmt1* statement named *name*, but the *stmt1* statement could not be found.

Action: Correct the configuration and restart.

T01CF102E Too many LINK statements specified for LCSNAME *lcs_name*.

Reason: A maximum of four link-level network adapters can be associated with a device by the LINK statement. More than four LINK statements were found for the same LCS statement.

Action: Correct the configuration and restart.

T01CF103I Network *ip_addr* added to MEDIA *med_name*.

Reason: A NETWORK statement was successfully parsed, and the network identified by *ip_addr* was added to the MEDIA identified by *med_name*.

Action: None. This is an Informational message.

T01CF104I LNI device *unit_addr* added to MEDIA *med_name*.

Reason: A configuration statement defining a network interface device was successfully parsed, and the device identified by *unit_addr* was added to the MEDIA identified by *med_name*.

Action: None. This is an Informational message.

T01CF105E Media name *med_name has no* associated NETWORK

Reason: The MEDIA statement identified by *med_name* does not have an associated NETWORK statement following it. Each MEDIA statement must be followed by at least one associated NETWORK statement. A ROUTE statement must follow the necessary MEDIA and NETWORK statements.

Action: Correct the configuration and restart.

T01CF106E A statement *stmnt* was not preceded by a MEDIA statement

Reason: A configuration statement identified by *statement* was encountered in the configuration member that was not preceded by a MEDIA statement. Each NETWORK, driver, and ROUTE statement must be preceded by a MEDIA statement with which it is associated.

Action: Correct the configuration and restart.

T01CF107E An ARP statement was not preceded by a driver statement

Reason: An ARP statement was encountered in the configuration member that was not preceded by a driver statement. Each ARP statement must be preceded by the MEDIA, NETWORK, and driver statements with which it is associated.

Action: Correct the configuration and restart.

T01CF108E A *stmnt* statement follows a virtual MEDIA

Reason: A driver statement identified by *stmnt* follows a MEDIA that specified TYPE=VIRTUAL. Driver statements are not permitted if the MEDIA type is VIRTUAL.

Action: Correct the configuration and restart.

T01CF109E Adapter number *number* is invalid.

Reason: Adapter numbers range from 0 to 15. Adapter numbers greater than 3 are valid only for 2216 devices.

Action: Correct the configuration and restart.

T01CF110E Media name *med_name* on *stmnt* statement does not exist

Reason: A ROUTE statement was found that specified a MEDIANAME parameter *med_name*, but no MEDIA statement was specified with that name.

Action: Correct the configuration and restart.

T01CF111W Invalid postprocessing entry detected in *mod_name*

Reason: A logic error occurred in module *mod_name*.

Action: Contact Customer Support.

T01CF113E DEST (*ip_addr*) not on same SUBNET (*ip_addr*) as IPADDRESS (*ip_addr*)

Reason: DEST and IPADDRESS must be on the same SUBNET.

Action: Correct the initialization parameters and restart.

T01CF114E ROUTER *rtr_name* is not on an attached subnet

Reason: The ROUTER keyword specified an IP address that was not on any existing subnet.

Action: Correct the initialization parameters and restart.

T01CF115E Media name *med_name* has already been used

Reason: The media name specified has already been used.

Action: Correct the input specification.

T01CF116E A MEDIANAME is required to connect the *stmnt* statement

Reason: The MEDIANAME is necessary to link this statement to the MEDIA statement issued previously.

Action: Correct the *stmnt* statement and restart.

T01CF117E IP address *ip_addr* is already in use

Reason: The IP address specified is already in use.

Action: Correct the initialization parameters.

T01CF118E LNI *lni_name* is still active

Reason: LNI *lni_name* must be inactive to be deleted.

Action: Deactivate the LNI and retry.

T01CF119E Syntax error in statement *stmnt*

Reason: A syntax error was detected in the statement specified.

Action: Correct the syntax and reissue.

T01CF120E The LOOPBACK configuration cannot be manipulated

Reason: The LOOPBACK configuration cannot be altered.

T01CF121E *kywd object* does not exist

Reason: The *kywd object* does not exist in the system.

T01CF122I Request complete

Reason: The operator-requested function is complete.

Action: None. This is an Informational message.

T01CF123E *product* component configuration is not yet complete

Reason: Initialization has not been completed for the named *product*.

Action: Try the command again later.

T01CF124E Network IP address *ip_addr* is still associated with Media *med_name*

Reason: The Media *med_name* cannot be deleted until all associations are deleted.

Action: Delete all associations before deleting media.

T01CF125E LNI *lni_name* is still associated with Media *med_name*

Reason: Media *med_name* cannot be deleted until all associations are deleted.

Action: Delete all associations before deleting media.

T01CF126E RC *ret_code* attempting to get the *latch_name* latch

Reason: An attempt to obtain the *latch_name* latch failed.

Action: If this happens regularly, please contact Customer Support.

T01CF127E A *cmnd_name* identifier is required

Reason: A *cmnd_name* identifier is required.

Action: Ensure the identifier exists in the configuration.

T01CF128E ROUTE entry not found

Reason: The ROUTE to be deleted was not found.

T01CF129E Media *med_name* has no ARP detail information.

Reason: ARP cannot be altered.

Action: Ensure Media *med_name* is properly configured.

T01CF130E Both DEST and GATEWAY IP addresses must be specified

Reason: Both DEST and GATEWAY addresses are required.

Action: Specify DEST and GATEWAY addresses and restart.

T01CF131E Both IP and MAC addresses must be specified

Reason: Both IP and MAC addresses are required.

Action: Specify IP and MAC addresses and restart.

T01CF132E Media *med_name* has no ARP information

Reason: Media *med_name's* ARP information not found.

Action: Ensure Media *med_name's* is properly configured.

T01CF133 *kywd object* deleted

Reason: The *kywd object* was deleted from the system.

Action: None. This is an Informational message.

T01CF134E Device *dev_name* is in *state_code* state

Reason: LNDC *dev_name* is in *state_code* state and cannot be altered.

T01CF150E APP configuration error: invalid SUBSYS Name *subsys_name*

Reason: The subsystem name specified is invalid.

Action: Correct the subsystem name and restart.

T01CF152E APP configuration error: Pload failed for *ptask*

Reason: The attempted PLOAD of *ptask* failed.

Action: Verify that the linklist/STEPLIB is properly configured.

T01CF153E Unable to *action* DCGDSN dataset *data_set_name*

Reason: The Locate, Recall, or Obtain of *data_set_name* failed.

Action: If the Locate failed, ensure the volume containing the data set is available.

- If the Recall failed, the data set was migrated and could not be recalled
- If the Obtain failed, the format-1 DSCB could not be obtained.

T01CF154E DCBDSN Dataset *data_set_name* is not on a DASD device

Reason: The DCBDSN data set *data_set_name* was not found.

Action: Ensure the DCBDSN data set is available.

T01CF155E DCBDSN Dataset *data_set_name* is of an unsupported dataset organization

Reason: The data set name specified is not a supported data set organization.

Action: Ensure data set *data_set_name* not found.

T01CF156I Telnet greeting initialization successful from member *mem_name*

Reason: Telnet greeting from member *mem_name* available for use.

Action: None. This is an Informational message.

T01CF157E Telnet greeting member *mem_name* does not exist in ARPAHELP dataset

Reason: Telenet greeting member *mem_name* not found in the ARPAHELP data set.

T01CF158E An I/O error occurred reading telnet greeting member *mem_name*

Reason: An I/O error occurred while reading the Telnet greeting member *mem_name*.

T01CF159E *diagnostic_information*

Reason: Further diagnostic information obtained via SYNADAF.

T01CF160E The open failed for telnet greeting member *mem_name*

Reason: The ARPAHELP data set could not be opened.

T01CF161E Refresh failed due to a storage obtain error; return code = *ret_code*

Reason: Storage obtain failed for greeting storage area.

Action: None. This is an Informational message.

T01CF162E A SERVICE statement is required in APP configuration member *mem_name*

Reason: A SERVICE statement was not found in member *mem_name*. This statement is required.

T01CF165I OBTAINLU (BEFORE) is ignored for *server_name* server on port *port_number*.

Reason: The Telnet server identified by *server_name* and *port_number* is configured to delay TN3270E negotiation until after presentation of the USS message 10 screen (STARTNVT is used on the SERVICE statement), but to select the LU name before message 10 (PRELU). These are contradictory parameters, since LU name selection is part of TN3270E negotiation. The PRELU parameter is ignored, and the LU name will be selected during TN3270E negotiation.

Action: Remove PRELU or use START3270.

T01CF166E *if_name*: A multiplexed configuration is not supported with *media_type* type media.

Reason: A multiplexed configuration was specified for a type of network medium that does not support such a configuration. In the message, *if_name* is the name of the interface in error and *media_type* is the type of network medium to which the interface was defined.

Action: Correct the configuration and restart Unicenter TCPaccess.

T01CF167E VIPA subnet *subnet_number* is already defined.

Reason: A VIPANET statement specified an IP address and subnet mask for a subnet that is already defined as a dynamic VIPA subnet. The message text provides the subnet number, which is the IP address logically ANDed with the subnet mask.

Action: Correct the VIPANET statements in the TCPCFG*xx* configuration member.

T01CF168E T01CF168E Media name *med_name* is reserved

Reason: The media name specified is a reserved name that has been given a special meaning within Unicenter TCPaccess. Therefore, it cannot be used as an installation defined media name.

Action: Correct the TCPCFG*xx* configuration member statement or command.

T01CF169E Bind security parameter validation failed due to a STORAGE OBTAIN failure, R15=*ret_code*, CB=*cb_name*

ret_code Return code (GPR 15 value) from the STORAGE OBTAIN macro.

cb_name Control block name.

Reason: Validation of the bind security configuration parameters failed. A STORAGE OBTAIN for the control block named in the message failed with the return code shown in the message. This is usually caused by the private area virtual storage requirements of the Unicenter TCPaccess address space exceeding the region size of the address space.

Action: Determine the cause of the STORAGE OBTAIN macro failure and correct it. If necessary, increase the region size for the Unicenter TCPaccess address space.

T01CF200E Required keyword *kywd* omitted on *cmnd* statement in configuration member *mbr*(#*stmt*).

kywd Required keyword.

cmnd Statement containing the error.

mbr APPCFG*xx* member.

stmt Logical statement number.

Reason: Self-explanatory

Action: APP task group initialization fails. Correct the error and restart the APP task group.

T01CF201E Invalid *kywd* value on *cmnd* statement in configuration member *mbr*(#*stmt*).

kywd Keyword in error.

cmnd Statement containing the error.

mbr APPCFG*xx* member*stmt* logical statement number.

Reason: Incorrect keyword value specified

Action: APP task group initialization fails. Correct the error and restart the APP task group.

T01CF202E Invalid *kywd* value on *cmnd*/*port* statement in configuration member *mbr*(#*stmt*).

kywd Keyword in error.

cmnd Statement containing the error.

port Port number.

mbr APPCFG*xx* member.

stmt Logical statement number.

Reason: Incorrect keyword value specified in a TNSSL statement

Action: APP task group initialization fails. Correct the error and restart the APP task group.

T01CF203E Multiple *cmnd* statements detected in configuration member *mbr*.

cmnd Statement containing the error.

mbr APPCFG*xx* member.

Reason: Self-explanatory

Action: APP task group initialization fails. Correct the error and restart the APP task group.

T01CF204E Duplicate PORT *port* on *cmnd* statement in configuration member *mbr*(#*stmt*).

port Port number..

cmnd Statement containing the error.

mbr APPCFG*xx* member.

stmt Logical statement number.

Reason: Two or more TNSSL statements were specified with the same port number.

Action: APP task group initialization fails. Correct the error and restart the APP task group.

T01CF205E Mutually exclusive *kywd* keywords on *cmnd* statement in configuration member *mbr*(#*stmt*).

kywd Keyword containing the error.

cmnd Statement containing the error.

mbr APPCFG*xx* member.

stmt Logical statement number.

Reason: If *kywd* is *KeyringDB*, SAFNAME and HFS elements were both specified on the same KEYRING statement. Otherwise, *kywd* is in conflict with another keyword on the statement.

Action: APP task group initialization fails. Correct the error and restart the APP task group.

T01CF206E *kywd* out of range on *cmnd* statement in configurationmember *mbr*(#*stmt*).

kywd Keyword in error.

cmnd Statement containing the error.

mbr APPCFG*xx* member.

stmt Logical statement number.

Reason: keyword value specified was not in the range of acceptable values.

Action: APP task group initialization fails. Correct the error and restart the APP task group.

T01CF207E Unpaired HFS kdb/stash files on *cmnd* statement in configuration member *mbr*(#*stmt*).

cmnd Statement containing the error.

mbr APPCFG*xx* member.

stmt Logical statement number.

Reason: HFSKDB was specified without HFSSTASH or vice-versa.

Action: APP task group initialization fails. Correct the error and restart the APP task group.

T01CF208E Duplicate KEYRING statements in configuration member *mbr*(#*stmt*).

mbr APPCFG*xx* member.

stmt Logical statement number.

Reason: Two or more KEYRING statements were specified with the same name.

Action: APP task group initialization fails. Correct the error and restart the APP task group.

T01CF209E Unable to locate keyring profile on *cmnd* statement for port *port* in configuration *mbr*.

cmnd Statement containing the error.

port Port number of the TNSSL statement causing the error.

mbr APPCFG*xx* member.

Reason: A TNSSL statement contained the name of an undefined keyring profile.

Action: APP task group initialization fails. Correct the error and restart the APP task group.

T01CF210W KEYRING(s) specified with no secure ports in configuration member *mbr*.

mbr APPCFG*xx* member.

Reason: One or more keyring profiles were defined but no TNSSL statements were detected.

Action: This is a warning message even though it is unlikely this condition was intended. APP task group initialization continues.

T01CF211E TNSSL statements detected but no KEYRINGs defined in configuration member *mbr*.

mbr APPCFG*xx* member.

Reason: Self-explanatory

Action: APP task group initialization fails. Correct the error and restart the APP task group.

T01CF212E No matching SERVICE statement for TNSSL port *port* in configuration member *mbr*.

port Port number.

mbr APPCFG*xx* member.

Reason: A port on a TNSSL statement could not be matched with a corresponding SERVICE statement.

Action: APP task group initialization fails. Correct the error and restart the APP task group.

Chapter 14

T01CO*nnn* Commutator Service Routine Messages

This chapter describes messages issued by the commutator service routine. These messages include T01CO000 through T01CO999.

T01CO*nnn*

T01CO001I Shutdown in progress for APP task

Reason: This message confirms that the shutdown request was received. The task group initiates orderly shutdown procedures.

Action: None. This is an Informational message.

T01CO002I APP task terminating

Reason: The task group is ready to terminate. The shutdown procedures either completed in an orderly manner or were forced by another shutdown command.

Action: None. This is an Informational message.

T01CO003E Permanent PTask *name* PAttach failure

Reason: Failure to activate Ptask *name*, normally APILOGTA.

Action: This is an internal problem. Notify Customer Support.

T01CO004E Permanent PTask *name* terminated with completion code *code*

Reason: Ptask *name* terminated abnormally. This may be a transient condition that resulted in an ABEND, in which case recovery will be successful.

Action: If the condition persists after a recovery attempt, notify Customer Support.

T01CO005I Permanent PTask *name* PAttached

This message is a confirmation of a Ptask activation.

Action: None. This is an Informational message.

T01CO006I Permanent PTask *name* terminated normally

Reason: This message is a confirmation of a Ptask termination.

Action: None. This is an Informational message.

T01CO007I Waiting for TCP task group initialization to complete

Reason: During startup, synchronizing various activity levels resulted in a controlled wait loop for another component to reach a required state. No action is necessary and message T01CO008I displays when the startup process is resumed.

Action: None. This is an Informational message.

T01CO008I Commutator initialization proceeding

Reason: This message indicates that commutator startup is resumed after waiting for some other event. This message follows T01CO007I.

Action: None. This is an Informational message.

T01CO009I Mount *vol_ser* on *unit_addr* for *job_name step_name* or reply 'NO'

Reason: A mount is required for a data set on magnetic tape. This allows the operator to cancel a tape mount request before the mount time expires. It can be used, for example, when a requested tape volume cannot be located.

Action: Mount the tape on the indicated device or reply NO to cancel the request if the tape cannot be mounted.

T01CO010I Waiting for available unit *unit_name* Reply 'U' to cancel.

Reason: This message displays at 30 second intervals if the tape mount request from message T01CO009I times out. A cancellation option is made available if the requested tape cannot be mounted.

Action: None. This is an Informational message.

T01CO011I MOUNT volser ON *unit* FOR *job_name step_name* OR REPLY 'NO'

Reason: A mount is required for a data set on magnetic tape. This allows the operator to cancel a tape mount request before the mount time expires. It can be used, for example, when a requested tape volume cannot be located.

Action: Mount the tape on the indicated device or reply NO to cancel the request if the tape cannot be mounted.

T01CO012I NO ACSSCB POINTER IN ACCOM

Reason: The ACSSCB pointer in control block ACCOM does not exist. The pointer was overlaid with zeros. The subsystem interface cannot be used and the product ABENDs with a user 2007.

Action: Save all output from the job. Contact Customer Support. Restart the job.

T01CO013I ACSSCB PTR NOT AN ASCB

Reason: The ACSSCB pointer in control block ACCOM does not point to a valid ACSB. The subsystem interface cannot be used and the product ABENDs with a user 2107.

Action: Save all output from the job. Contact Customer Support. Restart the job.

T01CO014I LOAD FAILED FOR CHAR/DBCS TRANSLATION MODULE *mod*

Reason: The attempt to load the specified module, *mod*, as single or double byte character set translation table failed.

Action: Verify that the character set name was created and is a valid table.

T01CO015I VALIDATION FAILED FOR CHAR/DBCS TRANSLATION MODULE *mod*

Reason: An invalid character set translation table (*mod*) was specified or defined. This table may have been created incorrectly or was not created for the specified *transopt* (single or double).

Action: Verify character set matches the translation mode.

T01CO016E ACLOAD FAILED - MODULE *module*

Reason: This message appears immediately before an ABEND issued by the ACLOAD routine in ACCOM when a PLOAD macro specifies a module (*module*) not usable for subsequent processing. Unless local modifications are made to the product distribution libraries, this message should only appear during testing, not in a product environment.

The code of the subsequent ABEND identifies the reason:

706 The module is linked as non-executable.

C06 The module is not found in the correct library, or is linked to be either overlay or scatter.

D06 An error occurred during the BLDL process.

Action: The product LOAD library must be in the STEPLIB DD statement of the startup JCL. If the ABEND code is a D06 and the module name is ARPAINIT, probably the LOAD library was not placed into (or was removed from) the STEPLIB DD statement in the startup JCL.

In most cases a relink with the correct attributes corrects this condition. The D06 ABEND is triggered by system action, and may require the assistance of a system programmer to resolve.

T01CO017I RACROUTE-RACINIT RC GREATER THAN 8

Reason: A user attempts to sign on to external security and receives a return code higher than eight from a RACROUTE-RACINIT sign-on attempt. The only valid return codes from RACROUTE-RACINIT are zero, four, and eight. The bad return code could be generated from an external security exit or module ACSECPC. The product ABENDs the ptask with an 0C1.

Action: Save all output from the job. Contact Customer Support for assistance to display RACROUTE-RACINIT return codes in the JOB log.

T01CO018I ACEE FOR VALIDATED USER DID NOT OCCUR

Reason: A ptask is attempting to perform a RACROUTE-RACINIT TYPE=DELETE function and does not have a valid ACEE address in its SECACEE field. The ptask ABENDs with an X'307'.

Action: Save all output from the job. Check for any SVC dumps that may have been issued. Check the JOB log and WTO logs for related messages. If you are using P-services to create an application, check the program logic of the failing ptask. Check the parameter lists passed to the P-service. If you are using the default applications, contact Customer Support. They can tell you how to display RACROUTE-RACINIT return codes in the JOB logs.

Chapter 15

T01DN*nnn* DNR Messages

This chapter describes messages issued by the Domain Name Resolver (DNR) task group. These messages are T01DN000 through T01DN999.

T01DN*nnn*

T01DN001I NDS/DNR initialization successful

Reason: The DNR task group was successfully started and was able to process the primary and secondary configuration members without error.

Action: None. This is an Informational message.

T01DN002E NDS/DNR initialization failed - reason code = *dec_ret_code* (*hex_ret_code*)

Reason: The DNR task group failed to initialize due to at least one error.

The following lists the valid values for dec X'hex':

4 X'0004'	Processing of primary configuration member NRCFG*xx* failed.
8 X'0008'	Getmain of NDSA storage (Subpool 241) failed.
12 X'000C'	GPOL for TIMER or SEND thread DSRB failed.
16 X'0010'	SCHED failed for TIMER or SEND thread.
20 X'0014'	GPOL for RECEIVER thread DSRB failed.
24 X'0018'	SCHED failed for RECEIVE thread.
28 X'001C	One or more DNR*xxxxx* modules could not be loaded.

Action: Correct the problem that caused the task group initialization to fail. Then restart the DNR task group.

T01DN003E Configuration member DNRCFGxx not found or in error

Reason: Module DNRICNFG could either not locate or successfully parse the primary configuration member DNRCFG*xx* (where *xx* is the primary configuration member name suffix).

Action: Make sure that the primary configuration member exists and that the control statements have the proper syntax and valid values. This message should be preceded by a message from DNRICNFG (T01DN015E - T01DN021E), which explains the error in more detail.

T01DN004I NDS/DNR shutdown successfully completed.

Reason: The DNR task group completed processing of directory requests and stopped execution.

Action: None. This is an Informational message.

T01DN005I NDS/DNR shutdown - freemain of NDSA failed, R15=*register_contents*

Reason: During DNR task group shutdown, the FREEMAIN of the NDSA failed. Register 15 following the FREEMAIN request is displayed in the message. The NDSA storage, which is in Subpool 241 (CSA), was not freed and cannot be recovered. The single parameter, *x*, indicates the contents of R15 after the FREEMAIN request

Action: Contact Customer Support for assistance in resolving this problem.

T01DN006I NDS/DNR shutdown - freemain of ASVT failed, R15=*ret_code*

Reason: During DNR task group shutdown, the FREEMAIN of the DNR ASVT failed. Register 15 following the FREEMAIN request is displayed in the message. The DNR ASVT storage, if not successfully freed, is freed by the operating system at task termination.

The single parameter, *x*, indicates the contents of register 15 after the FREEMAIN request.

Action: Contact Customer Support for assistance in resolving this problem.

T01DN007I NDS/DNR timer services initialized

Reason: The timer thread successfully initialized.

Action: None. This is an Informational message.

T01DN008I NDS/DNR timer services terminated

Reason: The timer thread terminated normally.

Action: None. This is an Informational message.

T01DN009W NDS/DNR timer services terminated - damaged clock

Reason: A STIMER/TTIMER request failed due to a damaged CPU clock. Timing services for the task group cannot continue. The directory requests may or may not complete following this error.

Action: Correct the problem with the CPU clock on the CPU that caused the problem. Then stop and restart the DNR task group.

T01DN010I NDS/DNR network receiver initialized

Reason: The receiver thread successfully initialized.

Action: None. This is an Informational message.

T01DN011I NDS/DNR network receiver terminated

Reason: The receiver thread terminated normally.

Action: None. This is an Informational message.

T01DN012W NDS/DNR receiver thread abended and has recovered. Unit of work being processed may not have completed!

Reason: An ABEND occurred while the receive thread was in control. The unit of work being processed may not have completed and the associated directory request may not complete normally. This problem may cause other DNR requests not to complete normally. An IFS formatted dump should accompany this message. An MVS formatted dump should accompany this message if the execution JCL included a SYSUDUMP DD statement.

Action: Contact Customer Support. Have ready for them the IFS formatted dump and the SYSUDUMP, if possible. If SYSLOG message IEA995I indicates that the active load module is DNRSCALL, query SMP/E for the PTF level of DNRSCALL before calling for support.

T01DN013I Orderly shutdown of DNR task group requested

Reason: The MVS operator requested that the DNR task group be shutdown normally by issuing either a %STOP DNR or %P command for the first time.

Action: None. This is an Informational message.

T01DN014W Fast shutdown of DNR task group requested

Reason: The MVS operator requested that the DNR task group be shutdown immediately by issuing either a %STOP DNR or %P command for the second time.

Action: Any Directory service requests that were not completed before the second operator command have been flushed. Any application programs waiting for the associated directory request to complete may not complete and may have to be cancelled.

T01DN015E Configuration command '*cmnd_name*' is invalid

Reason: While parsing the contents of the primary DNR configuration member, a non-valid DNR configuration command was encountered.

Action: Correct the configuration statements in the DNR configuration member. Then restart the DNR task group.

T01DN016E Only one *cmnd_name* command allowed in configuration member *mem_name*

Reason: While parsing the contents of the primary DNR configuration member, a command was encountered more than once. The *mem_name* variable is the primary DNR configuration member.

Action: Correct the primary DNR configuration member *mem_name* so that it contains only one of each DNR command.

T01DN017E Required keyword *kywd* missing on *stmnt* statement in configuration member *mem_name*

Reason: A required keyword was not specified as an operand on the specified command.

Action: Correct the DNR primary configuration member *mem_name*.Add the required keyword operand to the appropriate command.

T01DN018E Parsing of configuration member *mem_name* failed, reason code = *dec_ret_code* (*hex_ret_code*)

Reason: The IPARSE command could not successfully parse a command in the DNR primary configuration member.

Action: Verify that each keyword on every DNR configuration statement is spelled correctly and that the format of each command is correct. Then restart the DNR task group.

T01DN019E Configuration member *mem_name* did not contain a DNR options statement

Reason: DNR primary configuration member (*mem_name*) did not contain a statement that started with the command DNR.

Action: Correct the primary configuration member, adding a statement that starts with the command DNR.

T01DN020E Configuration member *mem_name* did not contain a POOLDEF options statement for 'NAME(DSRB)'

Reason: The primary DNR configuration member (*mem_name*) did not include a valid POOLDEF statement for the pool with the name DSRB.

Action: Correct or add a POOLDEF statement for the pool with the name DSRB in the primary DNR configuration member.

T01DN021E Configuration member *mem_name* does not exist. DNR initialization failed

Reason: The DNR primary configuration member (*mem_name*) could not be located in the data set concatenation referenced by the SYSPARM DD statement.

Action: Make sure that the DNR primary configuration member does exist in one of the data sets in the SYSPARM DD statement. If it does exist, then verify that the CNFG(*xx*) argument of the START DNR command contains the valid suffix.

T01DN022E Command *cmnd_name* failed in member *mem_name* on statement *stmnt*

Reason: A parsing error occurred while processing the command shown in the message. *stmnt* is the failing parameter and parameter value, up to 40 characters.

Action: Correct the error and restart the DNR task group.

T01DN023E Undefined keyword *kywd* on command *cmnd_name*, member *mem_name*

Reason: The TSO parser detected an unidentified keyword in the command statement.

Action: Correct the value in the configuration parameters and restart the DNR task group.

T01DN024E Return code *ret_code* received from configuration file I/O handler

Reason: An unexpected error was detected in the configuration file I/O handler. *ret_code* is the error/return code. This message is issued whenever an unexpected return code is received from ACPIPRMS and is preceded by another message that provides more information.

ACPIPRMS passes the following return codes:

00 OK.

04 EOF on member.

08 Member not found or unreadable.

12 DCB not open.

16 Invalid parameter list.

20 Assembly buffer too small .

Action: Refer to preceding message for information to help resolve the problem.

T01DN025T NDS/DNR processing directory request # *request*

Reason: This message is issued each time a new directory request is received and processing begins. .

T01DN026T NDS/DNR directory request # *request* completed normally

Reason: This message is issued each time a directory request completes execution unless the request timed out, the requesting task or address space terminated, or the request terminated due to an ABEND.

T01DN027I NDS/DNR directory request # *request* timed out

Reason: This message is issued each time a directory request completes execution due to the time limit being exceeded for that request.

Action: None. This is an Informational message.

T01DN028I NDS/DNR directory request # *request* cannot be completed. Requesting task or address space terminated

Reason: This message is issued each time a DNR request cannot be completed as the issuing task or address space terminated before the request could complete normally.

Action: None. This is an Informational message.

T01DN029I NDS/DNR directory request # *request* terminated

Reason: This message is issued for each DNR request that cannot be completed normally due to the operator requesting that the DNR task group shutdown.

Action: None. This is an Informational message.

T01DN030E NDS/DNR abend User 3000, broken DSRB chain, request # *request*

Reason: A DSRB, a pool element that represents one directory request, could not be found on one of the two internal DSRB chains. Since the DNR chains are maintained under the control of the local lock, this error represents either a serious DNR logic error or a storage overlay.

Action: Contact Customer Support. Have ready for them the IFS formatted dump and the SYSUDUMP, if possible.

T01DN031W NDS/DNR directory request # *request* cannot be completed - abend detected during request processing!

Reason: An ABEND occurred while a Request thread was in control. The unit of work being processed was ABENDed and the directory request terminated. This problem may or may not cause other DNR requests to terminate abnormally. An IFS formatted dump should accompany this message. An MVS formatted dump should accompany this message if the execution JCL included a SYSUDUMP DD statement.

Action: Contact Customer Support. Have ready for them the IFS formatted dump and the SYSUDUMP, if possible. If SYSLOG message IEA995I indicates that the active load module is DNRSCALL, query SMP/E for the PTF level of MOD DNRSCALL before calling for support.

T01DN032I NDS/DNR network sender initialized

Reason: The Send thread was able to successfully initialize.

Action: None. This is an Informational message.

T01DN033I NDS/DNR network sender terminated

Reason: The Send thread was able to terminate normally.

Action: None. This is an Informational message.

T01DN034W NDS/DNR sender thread abended and has recovered. Subsequent global requests may not complete normally

Reason: An ABEND occurred while the Send thread was in control. The unit of work being processed may not have completed and the associated directory request may not complete normally, if at all. This problem may cause other DNR requests not to complete. An IFS formatted dump should accompany this message. An MVS formatted dump should accompany this message if the execution JCL included a SYSUDUMP DD statement.

Action: Contact Customer Support. Have ready for them the IFS formatted dump and the SYSUDUMP, if possible. If SYSLOG message IEA995I indicates that the active load module is DNRSCALL, query SMP/E for the PTF level of DNRSCALL before calling for support.

T01DN035I NDS/DNR processing dump command, request = *req_type*

Reason: Acknowledges that the DNR is processing a dump command for the request type returned (*req_type*).

Action: None. This is an Informational message.

T01DN036W NDS/DNR invalid dump command entered - ignored

Reason: The DNR dump command that was entered was invalid. Either it could not be correctly parsed or a required suboperand was missing.

Action: Determine why the DUMP command was invalid and reissue the dump command. Verify that each keyword was entered correctly and if the CACHE DATA option was used that a data field was included (that is, DNR DUMP CACHE(DATA(SATURN.ACC.COM.)).

T01DN037I NDS/DNR completed processing dump command, completion code = *dec_comp_code* (*hex_comp_code*)

Reason: The dump command has completed execution.

Action: None. This is an Informational message.

T01DN038E NDS/DNR An abend occurred while processing dump command. Processing terminated.

Reason: An unexpected ABEND occurred while trying to process the dump command. This is most likely a software error.

Action: Make sure that both an IFS formatted dump and SYSUDUMP were produced, then contact Customer Support.

T01DN039E NDS/DNR unable to obtain DSRB for dump processing, IGPOL return code = *dec_ret_code* (*hex_ret_code*).

Reason: The dump command requires a Direct Services Request Block (DSRB) to process a dump request. However, the IGPOL service returned with a non-zero return code when it was called to allocate a DSRB. This problem can be caused by having either a too small region size or incorrect POOLDEF statements in the DNRCFG00 customization member.

Action: Verify that the POOLDEF statements in the DNRCFG00 TCPPARM member are correct and that a sufficiently large enough region size was specified on either the job or execute statement in the startup JCL.

T01DN040I NDS/DNR Processing purge command

Reason: Acknowledges that the DNR is processing the PURGE command.

Action: None. This is an Informational message.

T01DN042I NDS/DRN Completed processing purge command, Completion code = *dec_comp_code hex_ret_code*

Reason: The PURGE command has completed execution.

The decimal code is represented by *dec_comp_code*; the hexadecimal completion code is represented by *hex_ret_code*.

Action: None. This is an Informational message.

T01DN043E NDSF/DNR An abend occurred while processing purge command. Purge command processing terminated.

Reason: An unexpected ABEND occurred while trying to process the PURGE command. This is most likely a software error.

Action: Make sure that both an IFS formatted dump and SYSUDUMP were produced, then contact Customer Support.

T01DN044E NDS/DNR Unable to obtain DSRB for purge processing, IGPOL return code = *dec_comp_code hex_ret_code*

Reason: The PURGE command requires a Direct Services Request Block (DSRB) to process a dump request. However, the IGPOL service returned with a non-zero return code when it was called to allocate a DSRB. This problem can be caused by having either a too small region size or incorrect POOLDEF statements in the DNRCFG00 customization member.

The decimal code is represented by *dec_comp_code*; the hexadecimal completion code is represented by *hex_ret_code*.

Action: Verify that the POOLDEF statements in the DNRCFG00 TCPPARM member are correct and that a sufficiently large region size was specified on either job or execute statement in the startup JCL.

Chapter 16

T01RT*nnn* TelnetRTM Messages

This chapter describes the messages issued when using TelnetRTM. These include messages T01RT000 through T01RT999.

T01RT*nnn*

T01RT001I TelnetRTM initialization successful

Reason: TelnetRTM initialization completed successfully.

Action: None. This is an informational message.

T01RT002I TelnetRTM termination complete

Reason: TelnetRTM termination completed.

Action: None. This is an informational message.

T01RT003E TelnetRTM initialization failed, reason code = *rsn_code*

Reason: Initialization of the TelnetRTM task group failed.

Values for *rsn_code* are:

00 Success.

08 Configuration parameter validation failed.

12 RTVT STORAGE OBTAIN failed.

16 ASVT STORAGE OBTAIN failed.

20 Socket hash table STORAGE OBTAIN failed.

24 SLU name hash table STORAGE OBTAIN failed.

28 DSCA STORAGE OBTAIN failed.

32 DSPSERV CREATE failed.

36 ALESERV ADD failed.

40 ITIME ALLOC failed.

44 ILATCH ALLOC failed.

48 Pool header ILOAD failed.

52 Data space storage exhausted.

Action: Contact Customer Support.

T01RT004E DSPSERV *function* error - R15=*ret_code* R00=*rsn_code*

Reason: TelnetRTM data space could not be created, deleted, or extended. In the message, *function* will be either CREATE, DELETE, or EXTEND and *ret_ code* and *rsn_ code* will be the DSPERV return code and reason code.

Action: Contact Customer Support.

T01RT005E ALESERV *function* error - R15=*ret_code* R00=*rsn_code*

Reason: Addressability to the TelnetRTM data space could not be established or dropped. In the message, *function* is either ADD or DELETE and *ret_code* and *rsn_code* is the ALESERV return code and reason code.

Action: Contact Customer Support.

Chapter

17 T01F1*nnn* FTP Messages

This chapter describes the messages issued by the FTP program. These include messages T01F1001 through T01F1999.

T01F1*nnn*

T01F1001I VLT invocation option specified, but file SYSVLT is not allocated

Reason: Client program FTP or FTP2 was running in batch mode. The VLT invocation option was specified. File SYSVLT does not exist.

Action: Add a file in the JCL setup for the batch FTP job:

```
//SYSVLT  DD   SYSOUT=*,DCB=BLKSIZE=133
```

T01F1002E GETMAIN FAILED FOR DSA IN *program*

Reason: This message appears when the client software attempts to extract an encrypted password. Not enough core was available to perform a GETMAIN. The client cannot extract an encrypted password. The program parameter identifies the program where the failure occurred. If the client program is running interactively the client prompts the user for a password.

Action: The user should allocate a larger region size to their TSO session. If the problem persists, contact Customer Support.

T01F1003E FREEMAIN FAILED FOR DSA IN *program*

Reason: The GETTRPW subroutine issued a FREEMAIN and it failed. The GETTRPW routine performs password extraction. The client cannot extract an encrypted password. The program parameter identifies the program where the failure occurred. If the client program is running interactively, the client prompts the user for a password.

Action: Contact Customer Support.

T01F1004E GETTRPW RETURN CODE = *rc*

Reason: A user invoked client program FTP. An unexpected error occurred in initialization routine GETTRPW and returned the specified return code (*rc*).

Action: Contact Customer Support. Save the *rc* return code.

T01F1005I APP option specified, but no application name entered

Reason: A user invoked one of the client programs (TCPEEP, FTP, FTP2, or Telnet) utilizing the APP invocation option but no VTAM application name was entered. Client programs TCPEEP, FTP, or FTP2 then terminate. Client program Telnet uses the default VTAM application name.

Action: Refer to the *User Guide* for the syntax of the invocation option on the FTP and FTP2 commands; see the *Administrator Guide* or *System Management Guide* for information about the TCPEEP command. A user can issue the APPLID command under Client Telnet to change the VTAM application name.

T01F1006I APP name is longer than 8 characters *vtam_app_name*

Reason: A user invoked one of the client programs (TCPEEP, FTP, FTP2, or TELNET) utilizing the APP invocation option, but the VTAM application name (*vtam_app_name*) was longer than eight characters. Client programs TCPEEP, FTP, or FTP2 then terminate. Client program Telnet uses the default VTAM application name.

Action: Refer to the *User Guide* for the syntax of the invocation option on the FTP and FTP2 commands; see the *Administrator Guide* or *System Management Guide* for information about the TCPEEP command. A user can use the APPLID command under Client Telnet to change the VTAM application name.

T01F1007I APP and SYS options specified, SYS option ignored

Reason: A user invoked a client program (TCPEEP, FTP, FTP2, or TELNET) using both the APP and SYS invocation options. The client program ignored the SYS option and used the APP option.

Action: Refer to the *User Guide* for the syntax of the invocation option on the FTP and FTP2 commands; see the *Administrator Guide* or *System Management Guide* for information about the TCPEEP command.

T01F1008I VLT log follows:

Reason: Client program FTP or FTP2 was running in batch mode using VLT tracing. This should be the first line in the SYSVLT DD file.

Action: None. This is an Informational message.

T01F1009I GTF logging invoked

Reason: Client program FTP or FTP2 was running interactively utilizing GTF tracing.

Action: None. This is an Informational message.

T01F1010I Unable to allocate VLTLOGWK, return code from VLTGLGE=*rc*

Reason: An attempt to invoke client program FTP or FTP2 with the VLT invocation option. The clients call to VLTGLGE failed to allocate a VLTLOGWK work structure. VLT cannot trace without a VLTLOGWK work structure. The return code (*rc*) is the GETMAIN failure return code. VLT tracing stopped; FTP continues.

Action: Save the output of the FTP job. Contact Customer Support.

T01F1011I Line number *n* stripped off end of line

Reason: Client program FTP or FTP2 was running in batch mode. Line number *n* was found in columns 73 to 80 of the SYSGET input file line. The line number is ignored from the command stream to be presented to the FTP servers.

Action: The message disappears if the line numbers are removed from the batch FTP SYSGET DD input file.

T01F1012E Unexpected Reply –
***** VTELNET: ATOPEN fail - Storage Shortage**
***** VTELNET: ATOPEN fail - Invalid retcode**
***** VTELNET: ATOPEN fail - No core for TCCB**
***** VTELNET: ATOPEN fail - Close occurred**
***** VTELNET: ATOPEN fail - No APCB**
***** VTELNET: ATOPEN fail - API req failed**
***** VTELNET: ATOPEN fail - Timeout**
***** VTELNET: ATOPEN fail - Apend/Tpend**
***** VTELNET: ATOPEN fail - Invalid parms**
***** VTELNET: ATOPEN fail - Storage shortage**

Reason: This message displays on the user's screen when a Telnet or FTP user attempts to connect while an SOS condition exists. The specific message indicates the cause of the failure and can be one of the following:

Storage Shortage	Insufficient space for the proper operation of the product. Increase the region size for the job.
Invalid retcode	Invalid return code. Review the return code.
No core for TCCB	Insufficient storage to allocate a TCCB; insufficient core space for the proper operation of the product. Increase the region size for the job.
Close occurred	Release/disconnect from remote; indicates a network problem. A TCPEEP of the connection might help determine the cause.
No APCB	Bad parameter to ATOPN (no EPCB or APCB); indicates an internal error. Check for any SVC dumps that may have been issued and contact Customer Support.
API req failed	API error; indicates an internal error. Check the log file for additional information and contact Customer Support.
Timeout	Timeout opening the connection; normally indicates a problem communicating with the remote host. A TCPEEP of the connection might help determine the cause.

Apend/Tpend	API or transport provider termination; indicates API or transport provider termination.
Invalid parms	Bad parameter to ATOPN; indicates an internal error. Check the log file for additional information and contact Customer Support.

Action: Check the action required for the specific message.

T01F1013I TSO ENVIRONMENT UNABLE TO SUPPORT PUTGET, GETPASS RC=*rc*

Reason: A TSO command called the GETPASS subroutine to prompt for a non-display password. The PUTGET return code is given (*rc*). The return code determines the environmental problem experienced.

Action: Usually this error occurs when the TSO command is invoked improperly.

The following list provides some of the reasons the TSO environment may not support PUTGET:

- Running client programs in batch mode (PUTGET does not function in batch mode)
- The TSO profile is set to NOPROMPT (the PROFILE PROMPT command allows client programs to prompt for a password in non-display mode)
- Running client programs under the TSO XDC debugger (PUTGET does not function in TSO XDC debugger mode)

T01F1014I VLTLOGWK pointer corrupted VLT logging turned off, LOGPTR=*address*

Reason: Client program FTP or FTP2 was running utilizing VLT tracing. FTP discovered that the VLTLOGWK control block was corrupted. VLT tracing stopped; FTP continues.

Action: Save the output of the FTP job. Contact Customer Support.

T01F1015I ACB OPEN FAILED FOR VLT *code1*, ACB ERROR RC=*code2* VLT RC=*code3* R15=*code4* R0=*code5*

Reason: The attempt to open a server TELNET session with a VTAM application failed to complete successfully. Register 15 and 0 values are options fields.

These are the possible error codes:

code1 VLT applid.

code2 ACB error code. (continued on next page)

code3 VLT error code (in hexadecimal):

- 00 Operation completed successfully.
- 01 Normal with chained buffer.
- 02 Buffer purged by another entry.
- 03 I/O error in PUT operation.
- 04 I/O error in GET operation.
- 05 There are no applids left or passed applid is bad.
- 06 Error in REQSESS request.
- 07 Bad BIND image.
- 08 Error in SIGNA.
- 09 Error in SETLOGON.
- 0A Error in OPENSEC.
- 0B Error in RESET.
- 0C Cannot OPEN ACB: Unknown reason.
- 0D Cannot OPEN ACB: Network seems misdefined.
- 0E Cannot OPEN ACB: Storage shortage.
- 0F Cannot OPEN ACB: VTAM shutting down.
- 10 Error in TERMSES.
- 11 VQE queueing error.
- 12 Contention resolution failed for GET.
- 14 OPEN timeout error (two minute limit).
- 17 Data was received for an LU0 device that exceeds the buffer. LU0 does not support RU chaining. Increase the BUFFERSIZE on the TERMPROF statement for the TERMTYPE.

code4 Register 15 after OPEN request.

code5 Register 0 after OPEN request.

Action: If the ACB error code is nonzero, refer to the IBM manual *VTAM Reference Summary*.

T01F1016I RPL ERROR: REQ=*code1* R15=*code2* R0=*code3* RTNCD=*code4* SENSE=*code5* DEST=*code6* VLT RC=*code7*

Reason: A VLT attempted an RPL based VTAM request but the request failed. Destination applid (*code6*), register 15 (*code2),* and 0 (*code3*) values are option fields.

code1	RPL request type.
code2	Register 15 after RPL request.
code3	Register 0 after RPL request.
code4	RPL return code.
code5	RPL sense.
code6	Destination applid.
code7	VLT return code (refer to VLT codes defined in message ACC406I).

Action: Verify that VTAM is functioning properly. Use the *VTAM Reference Summary* manual for a Reason for the RPL codes.

T01F1017I VLT BIND ERROR: bind VLT RC=*vlt*

Reason: A VTAM BIND was received but failed VLT's bind validity checking. The user session is terminated.

Action: Verify that VTAM is functioning properly. Use the *SNA Reference Summary* manual for a description of a bind format. Refer to message T01F1071I for an explanation of the VLT return codes.

T01F1018I VLT ERROR: VLT RC=*vlt*

Reason: A VLT error was detected.

Action: Verify that VTAM is functioning properly. See message T01F1071I for a reason for the VLT return codes.

T01F1019E FTP RETURNS CC=*rc*

Reason: FTP returns the specified return code (*rc*).

Programs FTP and FTP2 can terminate and set these return codes:

0 FTP successful.

8 An error occurred in processing the FTP commands, or a remote host disconnected, or a file transfer failed.

Check the SYSPUT data output for the problem.

16 A file allocation error occurred.

Check that the SYSGET and SYSPUT files are in the batch JCL. Also, check that the FIOS option was entered in the invocation option parameters passed to FTP.

When running FTP2 in batch, unless the NOA option is specified, a return code of 16 can be returned when there is no NETRC file.

Action: If the user is executing FTP or FTP2 in batch, the user must check the SYSPUT and take the appropriate action.

Chapter 18

T01F2*nnn* FTP2 Messages

This chapter describes the messages issued by the FTP2 program. These include messages T01F2001 through T01F2999.

T01F2*nnn*

T01F2001I VLT invocation option specified, but file SYSVLT is not allocated

Reason: Client program FTP or FTP2 was running in batch mode. The VLT invocation option was specified. File SYSVLT does not exist.

Action: Add a file in the JCL setup for the batch FTP job:

```
//SYSVLT  DD   SYSOUT=*,DCB=BLKSIZE=133
```

T01F2002E NO NETRC FILE

Reason: FTP2 tried to allocate a NETRC file but failed.

Action: Check to make sure that the NETRC file for the user is in the catalog but not on DASD.

T01F2003I VLT log follows:

Reason: Client program FTP or FTP2 was running in batch mode utilizing VLT tracing. This should be the first line in the SYSVLT DD file.

T01F2004I GTF logging invoked

Reason: Client program FTP or FTP2 was running interactively utilizing GTF tracing.

Action: None. This is an Informational message.

T01F2005I Unable to allocate VLTLOGWK, return code from VLTGLGE=*rc*

Reason: An attempt to invoke client program FTP or FTP2 with the VLT invocation option. The clien'ts call to VLTGLGE failed to allocate a VLTLOGWK work structure. VLT cannot trace without a VLTLOGWK work structure. The return code (*rc*) is the GETMAIN failure return code. VLT tracing stopped; FTP continues.

Action: Save the output of the FTP job. Contact Customer Support.

T01F2006I APP option specified, but no application name entered

Reason: A user invoked one of the client programs (TCPEEP, FTP, FTP2, or TELNET) utilizing the APP invocation option but no VTAM application name was entered. Client programs TCPEEP, FTP, or FTP2 then terminate. Client program TELNET uses the default VTAM application name.

Action: Read your *User Guide* for the syntax of the invocation option on the FTP and FTP2 commands; refer to the *Administrator Guide* or *System Management Guide* for information about the TCPEEP command. A user can issue the APPLID command under Client Telnet to change the VTAM application name.

T01F2007I APP name is longer than 8 characters *vtam_app_name*

Reason: A user invoked one of the client programs (TCPEEP, FTP, FTP2, or TELNET) using the APP invocation option, but the VTAM application name (*vtam_app_name*) was longer than eight characters. Client programs TCPEEP, FTP, or FTP2 then terminate. Client program TELNET uses the default VTAM application name.

Action: Read your *User Guide* for the syntax of the invocation option on the FTP and FTP2 commands; refer to the *Administrator Guide* or *System Management Guide* for information about the TCPEEP command. A user can use the APPLID command under Client Telnet to change the VTAM application name.

T01F2008I APP and SYS options specified, SYS option ignored

Reason: A user invoked a client program (TCPEEP, FTP, FTP2, or TELNET) using both the APP and SYS invocation options. The client program ignored the SYS option and used the APP option.

Action: Read your *User Guide* for the syntax of the invocation option on the FTP and FTP2 commands; refer to the *Administrator Guide* or *System Management Guide* for information about the TCPEEP command.

T01F2009E GETMAIN FAILED FOR DSA IN *program*

Reason: This message appears when the client software wants to attempt to extract an encrypted password. Not enough core was available to perform a GETMAIN. The client cannot extract an encrypted password. The program parameter identifies the program where the failure occurred. If the client program is running interactively the client prompts the user for a password.

Action: The user should allocate a larger region size to their TSO session. If the problem persists, contact Customer Support.

T01F2010E ACF2 IS NOT ACTIVE ON THE SYSTEM

Reason: This message appears when the client software attempts to extract an encrypted password. Module GETTRPW determined that the external security system eTrust CA-ACF2 is inactive; the client cannot extract an encrypted password. If the client program is running interactively the client prompts for a password.

Action: If eTrust CA-ACF2 is not up, contact the local eTrust CA-ACF2 systems people to work on this problem. If eTrust CA-ACF2 is active or the site is not running it, contact Customer Support.

T01F2011E TOP SECRET IS NOT ACTIVE ON THE SYSTEM

Reason: This message appears when the client software attempts to extract an encrypted password. Module GETTRPW determined that the external security system eTrust CA-TopSecret is inactive. The client cannot extract an encrypted password. If the client program is running interactively, the client prompts the user for a password.

Action: If eTrust Top Secret is not up, contact the local eTrust CA-TopSecret systems people to work on their eTrust CA-TopSecret problem. If eTrust CA-TopSecret is active or the site is not running it, contact Customer Support.

T01F2012E NO EXTERNAL SECURITY IN THE SYSTEM

Reason: This message appears when the client software wants to attempt to extract an encrypted password. Module GETTRPW determined that no external security system is active. The client cannot extract an encrypted password. If the client program is running interactively, the client prompts the user for a password.

Action: If the external security system is not active, contact your local security administrator. If there is no external security system (RACF, eTrust CA-ACF2, eTrust CA-TopSecret), no action is required. If there is an external security system up and active, contact Customer Support.

T01F2013E FREEMAIN FAILED FOR DSA IN *program*

Reason: The GETTRPW subroutine issued a FREEMAIN and it failed. The GETTRPW routine performs password extraction. The client cannot extract an encrypted password. The program parameter identifies the program where the failure occurred. If the client program is running interactively, the client prompts the user for a password.

Action: Contact Customer Support.

T01F2014I UNABLE TO ESTABLISH ESTAE IN GETTRPW

Reason: The GETTRPW subroutine issued an ESTAE and it failed. The GETTRPW routine performs password extraction. The client cannot extract an encrypted password. If the client program is running interactively, the client prompts the user for a password.

Action: Contact Customer Support.

T01F2015I *command* IS NOT RUNNING AUTHORIZED

Reason: The TSO command (*command*) invoked the GETTRPW subroutine to perform MVS authorized services. The GETTRPW subroutine tried to use an MVS authorized service. The specified command or alias is not being accessed in an authorized manner. The GETTRPW routine performs password extraction. The command cannot extract an encrypted password. If the command is running interactively, the command prompts the user for a password.

Action: If the installation wants to run the command in an authorized manner, they must follow their local procedures for authorizing a TSO command. If the command is an authorized TSO command, check to make sure that the command was invoked in an authorized environment and/or manner.

Review the sections about authorizing client commands and APF authorizing of the common load data set in your *Getting Started* or *Planning Guide* for information about authorizing the client commands TCPEEP and FTP2.

Some reasons a client command might not be running authorized are:

- LINK library is not APF authorized. Your MVS system PARMLIB member IEAAPF*xx* may not contain a correct combination of library name and volser for the product LINK library.
- Current STEPLIB concatenation may contain the LINK library and a non-APF authorized library.
- Calling programs TCPEEP or FTP2 from a non-authorized environment.
- Programs TCPEEP or FTP2 are not in the AUTHCMD, AUTHPGM, and AUTHTSF sections in MVS system PARMLIB member IKJTSO*xx*.

T01F2016E RACROUTE REQUEST=EXTRACT ABEND IN GETTRPW

Reason: The GETTRPW subroutine issued a RACROUTE REQUEST=EXTRACT and the request ABENDed. The GETTRPW routine performs password extraction. The command cannot extract an encrypted password. If the TSO command is running interactively, the command prompts the user for a password.

Action: Save and format all dumps taken during the ABEND. Contact your local external security support personnel (RACF, eTrust CA-ACF2, eTrust CA-TopSecret) and provide all relevant information about the problem. Customer Support can provide information on the SAF call if needed.

T01F2017E ESTAE RECURSION OR ESTAE ERROR IN GETTRPW

Reason: The GETTRPWE ESTAE subroutine for GETTRPW was entered twice or ABENDed. The GETTRPW routine performs password extraction. The TSO command cannot extract an encrypted password. If the TSO command is running interactively the command prompts the user for a password.

Action: Format and save all dumps taken. Contact Customer Support.

T01F2018E RACROUTE RACXTRT RETURN CODE=00*wwxxyy*

Reason: The GETTRPW subroutine issued a RACROUTE REQUEST=EXTRACT and the request failed.

The return code is of the form:

ww The RACXTRT reason code.

xx The RACXTRT return code.

yy The RACROUTE return code.

The GETTRPW routine cannot perform encrypted password extraction. If the TSO command is running interactively, the command prompts the user for a password.

Action: Contact your local external security support personnel (RACF, eTrust CA-ACF2, eTrust CA-TopSecret) and provide all relevant information about the problem. The RACXTRT return and reason codes along with the RACOUTE return code should direct your local system support action. Customer Support can provide information on the SAF call and return codes if needed.

T01F2019E FTP.NETRC LOCATE RETURN CODE =*rc*

Reason: FTP2 issued a CAMLST name locate for the NETRC file. The locate failed; the return code is rc.

Action: Use the IBM publication *Catalog Administration Guide* to look up the return code. The documented return codes should direct your action.

T01F2020I Line number *n* stripped off end of line

Reason: Client program FTP or FTP2 was running in batch mode. Line number *n* was found in columns 73 to 80 of the SYSGET input file line. The line number is ignored from the command stream to be presented to the FTP servers.

Action: The message disappears if the line numbers are removed from the batch FTP SYSGET DD input file.

T01F2021E Unexpected Reply - * VTELNET: ATOPEN fail - Storage Shortage**
***** VTELNET: ATOPEN fail - Invalid retcode**
***** VTELNET: ATOPEN fail - No core for TCCB**
***** VTELNET: ATOPEN fail - Close occurred**
***** VTELNET: ATOPEN fail - No APCB**
***** VTELNET: ATOPEN fail - API req failed**
***** VTELNET: ATOPEN fail - Timeout**
***** VTELNET: ATOPEN fail - Apend/Tpend**
***** VTELNET: ATOPEN fail - Invalid parms**
***** VTELNET: ATOPEN fail - Storage shortage**

Reason: This message displays on the user's screen when a Telnet or FTP user attempts to connect while an SOS condition exists. The specific message indicates the cause of the failure and can be one of the following:

Storage Shortage	Insufficient space for the proper operation of the product. Increase the region size for the job.
Invalid retcode	Invalid return code. Review the return code.
No core for TCCB	Insufficient storage to allocate a TCCB; insufficient core space for the proper operation of the product. Increase the region size for the job.
Close occurred	Release/disconnect from remote; indicates a network problem. A TCPEEP of the connection might help determine the cause.
No APCB	Bad parameter to ATOPN (no EPCB or APCB); indicates an internal error. Check for any SVC dumps that may have been issued and contact Customer Support.
API req failed	API error; indicates an internal error. Check the log file for additional information and contact Customer Support.
Timeout	Timeout opening the connection; normally indicates a problem communicating with the remote host. A TCPEEP of the connection might help determine the cause.

Apend/Tpend	API or transport provider termination; indicates API or transport provider termination.
Invalid parms	Bad parameter to ATOPN; indicates an internal error. Check the log file for additional information and contact Customer Support.

Action: Check the action required for the specific message.

T01F2022I TSO ENVIRONMENT UNABLE TO SUPPORT PUTGET, GETPASS RC=*rc*

Reason: A TSO command called the GETPASS subroutine to prompt for a non-display password. The PUTGET return code is given (*rc*). The return code determines the environmental problem experienced.

Action: Usually this error occurs when the TSO command is invoked improperly.

Some of the reasons the TSO environment may not support PUTGET:

- Running client programs in batch mode (PUTGET does not function in batch mode)
- The TSO profile is set to NOPROMPT (the PROFILE PROMPT command allows client programs to prompt for a password in non-display mode.
- Running client programs under the TSO XDC debugger (PUTGET does not function in TSO XDC debugger mode)

T01F2023I Store Unique has been *state*.

Reason: This is an informative message issued after the SUNIQUE command has been received.

Action: Each time the command is issued, it turns STORE UNIQUE either on or off.

T01F2024I Firewall has been *state*

Reason: This message is a response to the FTP2 Client command FIREWALL. The *state* variable indicates enabled or disabled. The FIREWALL command toggles the implementation of Firewall-Friendly FTP RFC 1579 on and off. RFC 1579 specifies that for FTP data connection establishment, the client sends the PASV command to the remote and the PORT command to the local host. This is the default.

When the FIREWALL command is entered to disable this implementation, the FTP2 client sends the PASV command to the local host and the PORT command to the remote host.

Action: If your MVS host is behind a firewall and you are experiencing problems with your FTP connection timing out, toggle the firewall implementation with the FIREWALL FTP2 command.

T01F2025I VLTLOGWK pointer corrupted VLT logging turned off, LOGPTR=*address*

Reason: Client program FTP or FTP2 was running using VLT tracing. FTP discovered that the VLTLOGWK control block was corrupted. VLT tracing stopped; FTP continues.

Action: Save the output of the FTP job. Contact Customer Support.

T01F2026I ACB OPEN FAILED FOR VLT *code1*, ACB ERROR RC=*code2* VLT RC=*code3* R15=*code4* R0=*code5*

Reason: The attempt to open a server Telnet session with a VTAM application failed to complete successfully. Register 15 and 0 values are options fields.

These are the possible error codes:

code1 VLT applid.

code2 ACB error code. (continued on next page)

code3 VLT error code (in hexadecimal):

- 00 Operation completed successfully.
- 01 Normal with chained buffer.
- 02 Buffer purged by another entry.
- 03 I/O error in PUT operation.
- 04 I/O error in GET operation.
- 05 There are no applids left or passed applid is bad. This is a common error if the number of current FTP jobs exceeds half the total VTAM LUs defined in ACCPOOL. 2 LUs are needed for each job.
- 06 Error in REQSESS request.
- 07 Bad BIND image.
- 08 Error in SIGNA.
- 09 Error in SETLOGON.
- 0A Error in OPENSEC.
- 0B Error in RESET.
- 0C Cnnot OPEN ACB: Unknown reason.
- 0D Cannot OPEN ACB: Network seems misdefined.
- 0E Cannot OPEN ACB: Storage shortage.
- 0F Cannot OPEN ACB: VTAM shutting down.
- 10 Error in TERMSES.
- 11 VQE queueing error.
- 12 Contention resolution failed for GET.
- 14 OPEN timeout error (2 minutes limit).
- 17 Data was received for an LU0 device that exceeds the buffer. LU0 does not support RU chaining. Increase the BUFFERSIZE on the TERMPROF statement for the TERMTYPE.

code4 Register 15 after OPEN request.

code5 Register 0 after OPEN request.

Action: If the ACB error code is nonzero, read the IBM manual *VTAM Reference Summary*.

T01F2027I RPL ERROR: REQ=*code1* R15=*code2* R0=*code3* RTNCD=*code4* SENSE=*code5* DEST=*code6* VLT RC=*code7*

Reason: A VLT attempted an RPL based VTAM request but the request failed. Destination applid (*code6*), register 15 (*code2*), and 0 (*code3*) values are option fields.

code1 RPL request type.

code2 Register 15 after RPL request.

code3 Register 0 after RPL request.

code4 RPL return code.

code5 RPL sense.

code6 Destination applid.

code7 VLT return code (refer to VLT codes defined in message T01F2026I).

Action: Verify that VTAM is functioning properly. Use the *VTAM Reference Summary* manual for an explanation of RPL codes.

T01F2028I VLT BIND ERROR: bind VLT RC=*vlt*

Reason: A VTAM BIND was received but failed VLT's bind validity checking. The user session is terminated.

Action: Verify that VTAM is functioning properly. Use the *SNA Reference Summary* manual for a description of a bind format. Refer to message T01F2026I for an explanation of VLT return codes.

T01F2029I VLT ERROR: VLT RC=*vlt*

Reason: A VLT error was detected.

Action: Verify that VTAM is functioning properly. Refer to message T01F2026I for an explanation of VLT return codes.

T01F2030E FTP RETURNS CC=*rc*

Reason: FTP returns the specified return code (*rc*).

Programs FTP and FTP2 can terminate and set these return codes:

0 FTP successful.

8 An error occurred in processing the FTP commands or a remote host disconnected, or a file transfer failed.

Check the SYSPUT data output for the problem.

16 A file allocation error occurred. Check that the SYSGET and SYSPUT files are in the batch JCL. Also, check to make sure that the FIOS option was entered in the invocation option parameters passed to FTP.

When running FTP2 in batch, unless the NOA option is specified, a return code of 16 can be returned when there is no NETRC file.

Action: If the user is executing FTP or FTP2 in batch, the user must check the SYSPUT and take the appropriate action.

Chapter 19

T01F3*nnn* FTP3 Messages

This chapter describes the messages issued by the FTP3 program. These include messages T01F3001 through T01F3999.

T01F3*nnn*

T01F3000I FTP *product_name version*

Reason: This is the first message displayed after the user logon is processed. It indicates that FTP is running.

Action: None. This is an Informational message.

T01F3001E DD:*ddname* file open request failed.

Reason: Client FTP3 could not open the specified *ddname* indicated in the message text.

Action: Allocate the DD in TSO or specify the DD in the batch JCL stream.

T01F3002E ACF2 is installed but is not active.

Reason: eTrust CA-ACF2 is defined as a subsystem in MVS, but eTrust CA-ACF2 is not active or might not be completely initialized.

Action: Contact your system administrator to start eTrust CA-ACF2.

T01F3003E TOP SECRET is installed, but is not active.

Reason: eTrust CA-TopSecret is defined to MVS, but is not active or might not be completely initialized.

Action: Ask your system administrator to start eTrust CA-TopSecret.

T01F3004E System security is not active.

Reason: Client FTP3 expects to find eTrust CA-ACF2, RACF or eTrust CA-TopSecret. None of these security systems is found active in the system. Without an active security system, Client FTP3 is not able to get user attributes, such as encrypted passwords for logging in to the local host.

Action: Ask your system administrator to start the system's security function.

T01F3005E FTP is not authorized to access user information.

Reason: Client FTP3 does not have access to user information through system security.

Action: Contact your system administrator to verify that Client FTP3 is implemented as an authorized TSO command.

T01F3006E RACROUTE REQUEST=EXTRACT abend; error code: 00*wwxxyy*

Reason: The GetUserInfo function failed when issuing RACROUTE REQUEST=EXTRACT system call. The error code uses the following form:

ww The RACROUTE reason code.

xx The RACROUTE return code.

yy The SAF return code.

The GetUserinfo routine cannot perform encrypted password extraction. If the command is running interactively, the user is prompted for a password

Action: Contact your local external security support personnel (RACF, CA-ACF2, CA-TOP SECRET) and provide all relevant information. Refer to the IBM document *RACF Macro Reference* for information regarding the above return and reason codes.

The error code is the same as the RACROUTE return code and reason code.

T01F3007E Client FTP is not able to establish its error recovery environment during user validation.

Reason: The GetUserinfo function failed to establish ESTAE, or set up its error recovery environment.

Action: Gather your MVS and TSO release numbers and maintenance levels, and contact Customer Support.

T01F3008E RACROUTE ESTAE recursion or ESTAE error.

Reason: The GetUserinfo function failed to establish ESTAE, or set up a recovery environment.

Action: Report this problem to your local MVS and security support personnel.

T01F3009E RACROUTE REQUEST=EXTRACT failed; error code: 00*wwxxyy*

Reason: The GetUserinfo function failed issuing the RACROUTE REQUEST=EXTRACT system call. The error code uses the following form:

ww The RACROUTE reason code.

xx The RACROUTE return code.

yy The SAF return code.

The GetUserinfo routine cannot perform encrypted password extraction. If the command is running interactively, the user is prompted for a password.

Action: Contact your local external security support personnel (RACF, eTrust CA-ACF2, eTrust CA-TopSecret) and provide all relevant information. Refer to the IBM document *RACF Macro Reference* for information regarding the above return and reason codes.

T01F3010E Not able to open connection to the local host; error code: xx *err_msg*

Reason: Possible logic error or insufficient buffer space. A socket call failed returning error code: *xx err_msg*

Action: Use the error message provided (*err_msg*) to determine the cause of the failure.

T01F3011E Not able to connect to the FTP server; error code: xx *err_msg*

Reason: The Client FTP3 connect function failed so no connection was established to the remote host.

Action: Use the error message provided (*err_msg*) to determine the cause of the failure.

T01F3012E Call to mvsselect() failed; error code: xx *err_msg*

Reason: Possible logic error. Client FTP3 is waiting for a response that has not arrived.

Action: Use the error message provided (*err_msg*) to determine the cause of the failure.

T01F3013E Attempt to connect to server failed - Interrupted by user.

Reason: PA1 was pressed during local host connect.

Action: Retry the connection.

T01F3014E Attempt to connect to server timed out.

Reason: There was no response from the local server within MYOPENTIME.

Action: Verify that MYOPENTIME is set to a non-zero value in the FTP.DATA file. Check T01LOG for relevant messages.

T01F3015E Local Host OPEN failed with return code: *rrrr*.

Reason: The 220 reply was not received from the local server.

Action: Check T01LOG under the TCP started task for relevant messages.

T01F3016E Unable to log in to the local host: error code: xx *err_msg*

Reason: A C/socket API error prevented Client FTP3 from processing the user logon.

Action: Use the error message provided (*err_msg*) to determine the cause of the failure.

T01F3017E Unable to log in to the local host FTP Return Code = *rrrr*.

Reason: The expected reply code was not received from the local server in response to the user logon request.

Action: Verify that the user ID and password are valid for your security system.

T01F3018E TCPIP subsystem is not currently available.

Reason: The TCP transport provider specified with the TCPJOBNAME parameter is not currently available.

Action: Specify a value for TCPJOBNAME that corresponds to a currently available TCP subsystem.

T01F3021E Client FTP is not able to get the local host network address; error code: xx *err_msg*

Reason: The ioctl SIOCGIFCONF failed. Client FTP3 is not able to get local host ioconfig information.

Action: Use the error message provided (*err_msg*) to determine the cause of the failure.

T01F3022E Client FTP is not able to get the local host network address; error code: xx *err_msg*

Reason: The m_alloc function call failed. Client FTP3 is not able to allocate storage for ioconfig or for the second ioctl call.

Action: Use the error message provided (*err_msg*) to determine the cause of the failure.

T01F3023E Client FTP is not able to get the local host network address; error code: xx *err_msg*

Reason: The ioctl function is not able to get the network address from the ioconfig.

Action: Use the error message provided (*err_msg*) to determine the cause of the failure.

T01F3024I Command terminated at user's request.

Reason: The PA1 key or ATTENTION key was pressed to terminate the previous request or command.

T01F3026I FTP statement rejected by the local FTP server: *Statement*

Reason: An incorrect FTP.DATA statement was rejected by the FTP server.

Action: Correct the statement in the FTP.DATA file and resubmit the request. If you need more information about the statement, rerun the Client FTP3 request with the trace option.

T01F3027I The file *filename* is not allocated.

Reason: Client FTP3 is not able to allocate the specified file.

Action: Verify that the specified file exists and resubmit the Client FTP3 request.

T01F3028I An incorrect statement is detected in the TCPIP.DATA file: *Statement*

Reason: An incorrect statement was found in the TCPIP.DATA file.

Action: Correct the statement and resubmit the Client FTP3 command.

T01F3029I An incorrect statement is detected in the FTP.DATA file: *Statement*

Reason: An incorrect statement was found in the FTP.DATA file.

Action: Correct the statement and resubmit the Client FTP3 command.

T01F3030E Client FTP is not able to open *filename*.

Reason: Client FTP3 failed to open the specified file.

Action: Verify the data set attributes of the specified file.

T01F3031I An FTP.DATA file statement is not allowed as entered: *Statement*

Reason: An incorrect statement was detected in the FTP.DATA file.

Action: Correct the statement and reenter the Client FTP3 command.

T01F3032E Command *cmnd* is not supported.

Reason: The user entered an unsupported command.

Action: Enter HELP for a list of commands supported by Client FTP3.

T01F3034E *cmnd* command unknown

Reason: User has entered a command not recognized by Client FTP.

Action: Correct the entry and resubmit the Client FTP3 request. Enter HELP for a list of commands supported by Client FTP3.

T01F3035E Client FTP was not able to process the USER command; error number: *xx err_msg*.

Reason: Client FTP3 is not able to log the user into the remote host.

Action: Use the error message provided (*err_msg*) to determine the cause of the failure.

T01F3036I Call to mvsselect() socket function failed; error code: xx *err_msg*

Reason: Possible logic error. Client FTP3 is waiting for a response that has not arrived.

Action: Use the error message provided (*err_msg*) to determine the cause of the failure.

T01F3037I Remote host closed control connection unexpectedly.

Reason: The remote host FTP server closed the control connection after data transfer was completed. This might indicate that the control connection timed out or the remote host has crashed.

Action: Retry; if the problem persists, contact your system administrator.

T01F3039E No response from the remote FTP server in *nn* seconds.

Reason: The remote FTP server is not responding after a time-out.

Action: Retry; if the problem persists, contact your network administrator. If the error is not caused by implementation restrictions, contact Customer Support.

T01F3040E The remote host is not connected.

Reason: A Client FTP3 CLOSE command was issued, but the remote host is not connected and, therefore, not able to respond.

Action: Connect to the remote host by issuing an open command or reenter the Client FTP3 command.

T01F3041E No HELP information found.

Reason: There is no help information for the specified command.

Action: Enter HELP for a list of commands supported by Client FTP3.

T01F3042E Client FTP is not able to open a connection to the remote host; error code: xx *err_msg*

Reason: A socket function failed. Client FTP3 is not able to open a data connection.

Action: Use the error message provided (*err_msg*) to determine the cause of the failure.

T01F3043E Client FTP is not able to open a connection to the remote host; error code: xx *err_msg*

Reason: Possible logic error. A BIND function failed trying to open a data connection.

Action: Use the error message provided (*err_msg*) to determine the cause of the failure.

T01F3045E FTP listen() failed; error code: xx *err_msg*

Reason: The listen function call failed in NLST services; error code: *xx err_msg*

Action: Use the error message provided (*err_msg*) to determine the cause of the failure.

T01F3046E Unable to connect to remote host in *nn* seconds.

Reason: Connection to remote host FTP server data port timed out.

Action: Contact your network administrator and restart Client FTP3.

If the error is not caused by implementation restrictions, contact Customer Support.

T01F3047E accept() failed; cannot create remote data socket; error code: xx *err_msg*

Reason: Possible logic error. The socket accept() function failed. Client FTP3 is not able to open the data connection socket, error code: *xx err_msg*

Action: Use the error message provided (*err_msg*) to determine the cause of the failure.

T01F3049E Already connected to host *host_name*.

Reason: A session with the specified remote host was already active when an open command was entered.

Action: Continue operation or close the current connected remote host and open another remote host.

T01F3050E Request to open a connection to remote host *host_name* failed.

Reason: Possible user error. Client FTP3 failed to resolve the host name.

Action: Verify that the correct host name or domain name server specification was used. Also verify that the domain name resolver is functioning properly.

T01F3051E Client FTP was not able to send the PASS command to the remote host; error code: xx *err_msg*

Reason: The PASS command failed.

Action: Use the error message provided (*err_msg*) to determine the cause of the failure.

T01F3052E Client FTP failed to receive a reply to the PASS command; error code: xx *err_msg*

Reason: Client FTP3 is unable to receive a response from the remote host while processing the PASS command.

Action: Use the error message provided (*err_msg*) to determine the cause of the failure.

T01F3054E Transmission mode does not allow restart.

Reason: Data transfer restart is allowed only if the data transfer mode is block mode.

Action: Use the mode B command to switch the data transfer mode before the restart.

T01F3055E Unable to allocate CHKPOINT data set.

Reason: Client FTP3 is not able to allocate the *userid*.FTP.CHKPOINT file.

Action: Contact your system administrator for help.

T01F3056E Unable to open CHKPOINT data set.

Reason: Client FTP3 is not able to open the *userid*.FTP.CHKPOINT file.

Action: Contact your system administrator for help.

T01F3057I TSO returned : return code xx.

Reason: The TSO command submitted to TSO returned *xx*. If *xx* was returned, a TSO error message describes the error.

If the user entered an unauthorized TSO command the TSO service routine rejects the command with return code of 20 and reason code of 60 because Client FTP3 is running as an authorized TSO command and TSO/E IKJEFTSR specifically allows you to invoke authorized functions from an unauthorized application program. It maintains system security by running an invoked authorized function in its own isolated environment. However, to maintain system security, an authorized application program can use the TSO/E service facility to invoke only authorized programs or commands, or CLISTs consisting of only authorized programs and commands.

Action: None. This is an Informational message.

T01F3059E recv() error code: xx *err_msg;*

Reason: Socket receiver error; error code listed in T01F3001E.

Action: Use the error message provided (*err_msg*) to determine the cause of the failure.

T01F3303E TSO is required when executing a TSO command.

Reason: The user attempted to issue a TSO command when not running under TSO.

Action: Retry the command under TSO.

T01F3310I FTP Return Code = *rrrr*

Reason: The Client FTP3 command failed with return code *rrrrr*.

Action: None. This is an Informational message.

T01F3502E Too many positional parameters entered.

Reason: Only host and remote port number can be specified as command line arguments.

Action: Reissue the FTP command specifying the appropriate command line arguments.

T01F3503E No Timeout value provided.

Reason: The TIMEOUT keyword was specified but no time-out value was provided.

Action: Specify a value for TIMEOUT.

T01F3504E No TCP Jobname provided

Reason: The TCPJOBNAME keyword was specified but no value was provided.

Action: Specify a value for TCPJOBNAME.

T01F3505E Incorrect EXIT parameter in the FTP command: EXIT=*nn*

Reason: The EXIT keyword was provided but the value was invalid.

Action: Specify a correct value for EXIT.

T01F3506E No Translate Table Dataset name provided.

Reason: The TRANSLATE keyword was specified but no table name was provided.

Action: Specify a valid translate table name.

T01F3507E No subsystem ID provided.

Reason: The SSID keyword was specified but no value was provided.

Action: Specify a value for SSID.

T01F3508E Unknown parameter: cccccccc

Reason: A keyword was specified that is not valid.

Action: Remove the invalid keyword from the command line.

T01F3602E Call to mvsselect() failed; error code: xx *err_msg*

Reason: Possible logic error. Client FTP3 is waiting for a response.

Action: Use the error message provided (*err_msg*) to determine the cause of the failure.

T01F3603E Attempt to establish data connection failed - Interrupted by user.

Reason: The PA1 or ATTENTION key was pressed while waiting for data connection to open.

Action: Retry the connection.

T01F3604E Attempt to establish data connection timed out.

Reason: There was no response to the connect request in *dconntime*.

Action: Verify that *dconntime* is set to a non-zero value in the FTP.DATA file. Check the T01LOG for relevant messages.

T01F3605E Not able to establish data connection: error code: xx *err_msg*

Reason: The attempt to open the data connection failed.

Action: Use the error message provided (*err_msg*) to determine the cause of the failure.

T01F3606E Not able to establish data connection: error code: xx *err_msg*

Reason: The connect call failed for the data connection.

Action: Use the error message provided (*err_msg*) to determine the cause of the failure.

T01F3616E Command *command* requires a remote host connection: use OPEN first.

Reason: The user attempted to invoke a command that requires remote host without issuing OPEN first.

Action: Open the remote host and retry the command.

Chapter 20

T01FB*nnn* Batch FTP Messages

This chapter describes the messages issued by the Batch FTP program. These include messages T01FB001 through T01FB999.

T01FB*nnn*

T01FB001E User call parm list error

Reason: The user passed incorrect parameters.

Action: Pass the proper parameters to FTPBATCH.

T01FB002E Invalid user command request (not O,S,R,C)

Reason: The user command was not O (Open), S (Send), R (Receive), or C (Close).

Action: Pass one of the listed commands to FTPBATCH.

T01FB003E Error reading SYSOPT option record

Reason: The SYSOPT record does not start with FTP or FTP2.

Action: Begin the SYSOPT record with FTP or FTP2, to indicate the requested FTP client.

T01FB004E Batch open request to FTP/FTP2 application has failed

Reason: An Open command failed.

T01FB005E Invalid message length specified

Reason: In a Send command, the specified message length was not in the range 1-79.

Action: Specify a message length in the range 1-79.

T01FB006E User and FTP/FTP2 out of sync

Reason: FTP/FTP2 was not expecting a Send command.

Action: Issue the Send command in the proper sequence.

T01FB007E Invalid command request from FTP/FTP2

Reason: FTP/FTP2 issued a command other than SYSGET or SYSPUT.

T01FB008E FTP/FTP2 batch open error

Reason: FTP/FTP2 was unable to open the requested file.

T01FB009E FTP/FTP2 and user out of sync

Reason: FTP/FTP2 returned a SYSGET command, but the last user command was not a Send.

T01FB010E GETMAIN for message queue area failed

Reason: Available storage is insufficient for getting a new message queue area.

Chapter 21

T01GD*nnn* GateD Messages

This chapter describes messages issued by the GateD task group. These messages include T01GD001 through T01GD999.

T01GD*nnn*

T01GD001I Gated subtask attached successfully

Reason: The attempt to attach the GateD subtask was successful.

Action: None. This is an Informational message.

T01GD002E Gated subtask attach failed, Rc=*ret_code*

Reason: This error is generally encountered because the GateD module was not found or there was insufficient memory to complete the task.

T01GD003E Gated subtask ended; restarting in 30 seconds

Reason: The GateD subtask ended prematurely.

T01GD005I A GateD *cmd_name* command has been successfully processed.

Reason: A GateD operator command *cmd_name* was successfully processed by the operator interface, and the request has been posted to the GateD subtask.

Action: None. This is an Informational message.

T01GD006E A GateD *cmd_name* is received but the GateD subtask is not running | is already running

Reason: Either a GATED START command was issued when the GateD subtask was already running, or a GATED STOP or other command was issued when the GateD subtask was not running.

Action: Stop or start the GateD subtask as appropriate.

T01GD007E The GateD ISRB could not be scheduled. Request failed

Reason: A GATED START command was received, but the GateD subtask ISRB cold not be scheduled.

Action: This can be caused by a storage shortage condition. If no storage shortage exists, retry the request. If it still fails, contact Customer Support.

T01GD008I GateD subtask ended

Reason: The GateD subtask ended, either in response to an operator request, or because the TCP task group is shutting down.

Action: None. This is an Informational message.

Chapter 22

T01IP*nnn* Internet Layer Messages

This chapter describes the messages issued by the Internet layer (IP, ICMP) routine. These include messages T01IP001 through T01IP999.

T01IP*nnn*

T01IP001D Protocol number not supported, protocol = *proto*

Reason: An IP datagram arrived for protocol *proto*, which is not found in the RAW IP protocol table.

Action: Determine if the protocol number should be supported. If so, examine the supporting application for problems.

T01IP002D UDP port number not supported, port = *port_num* from *remoteIP*:*rport*.

Reason: A UDP datagram arrived that is not found in the UDP Port table.

port_num	Port from which unfound UDP datagram arrived.
rport	Remote port.
remoteIP	IP address of remote host.

Action: Determine if the port number should be supported. If so, examine the supporting application for problems.

T01IP003E TCP port number not supported, port = *port_num* from *remoteIP:rport*.

Reason: A TCP datagram arrived that is not found in the TCP Port table.

port_num	Port from which unfound TCP datagram arrived.
rport	Remote port.
remoteIP	IP address of remote host.

Action: Determine if the port number should be supported. If so, examine the supporting application for problems.

T01IP010E ICMP checksum error From *remoteIP*.

Reason: An ICMP datagram arrived with a checksum error. The remote host IP address is represented by *remoteIP*.

Action: Examine the remote host for problems.

T01IP011E ICMP message too short (*numb*), from *remoteip*.

Reason: An ICMP datagram arrived that is too short. The length is represented by *num*: the remote host IP address is represented by *remoteIP*.

Action: Examine the remote host for problems.

T01IP012D ICMP message session not found, from *remoteIP*.

Reason: An ICMP datagram arrived for which the corresponding endpoint cannot be found. The remote host IP address is represented by *remoteIP*.

Action: Examine the remote host for problems.

T01IP013I *proto* session *localIP:lport remoteIP:rport* rerouted from *oldgwIP* to *newgwIP*.

Reason: An ICMP datagram arrived that resulted in re-routing a session.

proto	Protocol number of the session.
lport	Source port number.
rport	Destination port number.
localIP	Source IP address.
remoteIP	Destination IP address.
oldgwIP	Old gateway IP address
newgwIP	New gateway IP address.

Action: Examine the network for routing problems.

T01IP014E *proto* session *localIP:lport remoteIP:rport,* destination unreachable and unable to reroute.

Reason: An ICMP datagram arrived which indicated the destination was unreachable.

proto	Protocol number of the session.
lport	Source port number.
rport	Destination port number.
localIP	Source IP address.
remoteIP	Destination IP address.

Action: The protocol of the session is represented by *proto*; local and remote port numbers are represented by *lport* and *rport*; local and remote IP addresses are represented by *localIP* and *remoteIP*. Examine the network for routing problems.

T01IP015T ICMP *type* received for protocol *proto, srcIP:srcPORT dstIP:dstPORT* from *senderIP*.

Reason: An ICMP datagram arrived.

type	Type of ICMP datagram that arrived.
proto	Protocol number of the header.
srcIP	Source IP address.
srcPORT	Source port number.
dstIP	Destination IP address.
dstPORT	Destination port number.
senderIP	Sender IP address.

Action: Examine the network for routing problems.

T01IP016E *proto* session *localIP:lport remoteIP:rport,* time-to-live exceeded from *senderIP*.

Reason: An ICMP datagram arrived indicating the IP time-to-live field was exceeded.

proto	Protocol number of the session.
lport	Source port number.
rport	Destination port number.
localIP	Source IP address.
remoteIP	Destination IP address.
senderIP	Sender IP address.

Action: Examine the network for routing problems. Possibly increase the default time-to-live.

T01IP017I Redirect received for *destIP* to *newrouterIP* from *oldrouterIP* type = *redirect_type* code = *redirect_code*

Reason: An ICMP arrived specifying a redirect.

destIP	IP address of destination gateway.
newrouterIP	IP address of new gateway.
oldrouterIP	IP address of sending gateway.
redirect_type	Type of network or host redirect received.
redirect_code	Redirect code (zero - three) .

Action: None. This is an Informational message.

T01IP018E ICMP parm problem xxx received for *localIP*, to *remoteIP*, from *senderIP*.

Reason: An ICMP datagram arrived specifying a parameter problem.

xxx	Offset into the datagram in error.
localIP	Source address of the datagram in error.
remoteIP	Destination address of the datagram in error.
senderIP	Sending host IP address.

Action: Correct the field in error.

T01IP019E ICMP *type* in error. from *senderIP*.

Reason: An ICMP datagram arrived which was in error. The type of the ICMP datagram in error is represented by *type*; the IP address of the sending host is represented by *senderIP*.

Action: Examine the remote host to correct the field in error.

T01IP030E Time to live exceeded on forwarded datagram. from *localIP* to *remoteIP*.

Reason: A datagram arrived that was to be forwarded through with a time-to-live field of zero or one. The source and destination host IP addresses are represented by *localIP* and *remoteIP*.

Action: Examine the remote host to increase its time-to-live field.

T01IP031E *proto* datagram *localIP:lport remoteIP:rport,* Unable To Route To Destination.

Reason: A datagram arrived that was to be forwarded but a route could not be obtained.

proto	Protocol number of the session.
lport	Source port number.
rport	Destination port number.
localIP	Source IP address.
remoteIP	Destination IP address.

Action: Examine the network for routing problems.

T01IP032I Redirect sent to *localIP,* for *remoteIP,* new router *newrouterIP*.

Reason: A datagram arrived which was to be forwarded and the route was for the same subnet.

localIP	Source IP address.
remoteIP	Destination IP address.
newrouterIP	Gateway described in the redirect.

Action: Examine the network for routing problems.

T01IP040T *proto* datagram *ldestIP:destPort srcIP:srcPort* unable to forward to destination.

Reason: A datagram arrived that was to be forwarded but forwarding was disabled.

proto	Protocol number of the session.
destIP	Destination IP address.
destPort	Destination port number.
srcIP	Source IP address.
srcPort	Source port number.

Action: Examine the network for routing problems.

T01IP041D *proto* datagram *localIP:lport remoteIP:rport,* protocol not supported.

Reason: A datagram arrived specifying a protocol that is not supported (no endpoint found).

proto	Protocol number of the session.
lport	Source port number.
rport	Destination port number.
localIP	Source IP address.
remoteIP	Destination IP address.

Action: Examine the application that supports this protocol.

T01IP042D UDP datagram *localIP:lport remoteIP:rport, port* not found.

Reason: A UDP datagram arrived for a port that has no session.

lport	Source port number.
rport	Destination port number.
localIP	Source IP address.
remoteIP	Destination IP address.

Action: Examine the application that supports this port.

T01IP043I TCP datagram *localIP:lport remoteIP:rport* session not found.

Reason: A TCP datagram arrived for which no session could be found.

lport	Source port number.
rport	Destination port number.
localIP	Source IP address.
remoteIP	Destination IP address.

Action: Examine the application that supports this port.

T01IP044E IP checksum error from *remoteIP*.

Reason: An IP datagram arrived with a checksum error. The remote host IP address is represented by *remoteIP*.

Action: Examine the remote host for problems.

T01IP045E IP datagram truncated for *proto, IPid, localIP:lport remoteIP:rport*.

Reason: An IP datagram was received whose header indicated a length exceeding the size of the packet. Part of the data portion is apparently lost. The datagram is discarded.

proto	Protocol number.
IPid	IP identifier number.
lport	Source port number.
rport	Destination port number.
localIP	Source IP address.
remoteIP	Destination IP address.

Action: Examine the remote host.

T01IP046E IP datagram too short for *proto, IPid, localIP:lport remoteIP:rport*.

Reason: An IP datagram was received whose length would not hold a complete IP header. The datagram was truncated by the network or badly formed by the remote host. The datagram is discarded.

proto	Protocol number.
IPid	IP identifier number.
lport	Source port number.
rport	Destination port number.
localIP	Source IP address.
remoteIP	Destination IP address.

Action: Examine the remote host.

T01IP047E IP datagram reassembly error for *proto, IPid, localIP:lport remoteIP:rport*.

Reason: An IP datagram was received whose reassembly offset value would cause the total datagram length to exceed 65535, the maximum size for an IP datagram. The datagram is discarded.

proto	Protocol number.
IPid	IP identifier number.
lport	Source port number.
rport	Destination port number.
localIP	Source IP address.
remoteIP	Destination IP address.

Action: Examine the remote host.

T01IP048E IP datagram resource shortage for *type, localIP:lport remoteIP:rport*.

Reason: An IP datagram was received that could not be processed due to a shortage of internal resources. The datagram is discarded.

type	Resource type.
lport	Source port number.
rport	Destination port number.
localIP	Source IP address.
remoteIP	Destination IP address.

Action: Examine the resource type. It may be necessary to expand the buffer pool(s) with the POOLDEF statement in the TCPCFG*xx* configuration member, or to increase the region size for the product.

T01IP049E IP datagram invalid source route for *proto, IPid, localIP:lport remoteIP:rport*

Reason: An IP datagram was received whose IP header contained an invalid source route option. The datagram is discarded.

proto	Protocol number of the session.
IPid	IP identifier number.
lport	Source port number.
rport	Destination port number.
localIP	Source IP address.
remoteIP	Destination IP address.

Action: Examine the remote host.

T01IP050I IP datagram reassembly timeout for *proto, IPid, localIP:lport remoteIP:rport*.

Reason: A datagram timed out during reassembly.

proto	Protocol number.
IPid	IP identifier number.
lport	Source port number.
rport	Destination port number.
localIP	Source IP address.
remoteIP	Destination IP address.

Action: Examine the network for dropped packets.

T01IP051E IP output no LNI segment for *proto*, *IPid*, *localIP:lport remoteIP:rport*.

Reason: IP is attempting to build an IP header for a datagram, but found errors in the options passed from the higher layer.

proto	Protocol number.
IPid	IP identifier number.
localIP	Source IP address.
lport	Source port number.
remoteIP	Destination IP address.
rport	Destination port number.

Action: This may indicate an internal error. Save all output and contact Customer Support.

T01IP052D IP datagram follows:

Reason: An error was encountered in an inbound IP datagram. The datagram is dumped following the message.

Action: This diagnostic message is preceded by an error or warning message. Follow the action for that message.

T01IP053E IP output header will not fit for *proto*, *IPid*, *localIP:lport remoteIP:rport*.

Reason: IP is attempting to build an IP header for a datagram, but found an error in the data or options passed from the higher layer.

proto	Protocol number.
IPid	IP identifier number.
localIP	Source IP address.
lport	Source port number.
remoteIP	Destination IP address.
rport	Destination port number.

Action: This may indicate an internal error. Save all output and contact Customer Support.

T01IP054E IP output options will not fit for *proto*, *IPid*, *locapIP:lport remoteIP:rport*.

Reason: IP is attempting to build an IP header for a datagram, but found an error in the data or options passed from the higher layer.

proto	Protocol number.
IPid	IP identifier number.
localIP	Source IP address.
lport	Source port number.
remoteIP	Destination IP address.
rport	Destination port number.

Action: This may indicate an internal error. Save all output and contact Customer Support.

T01IP055E IP output bad source route list for *proto*, *IPid*, *locapIP:lport remoteIP:rport*.

Reason: IP is attempting to build an IP header for a datagram, but found an error in the data or options passed from the higher layer.

proto	Protocol number.
IPid	IP identifier number.
localIP	Source IP address.
lport	Source port number.
remoteIP	Destination IP address.
rport	Destination port number.

Action: This may indicate an internal error. Save all output and contact Customer Support.

T01IP056E IP output option length is zero for *proto*, *IPid*, *locapIP:lport remoteIP:rport*.

Reason: IP is attempting to build an IP header for a datagram, but found an error in the data or options passed from the higher layer.

proto	Protocol number.
IPid	IP identifier number.
localIP	Source IP address.
lport	Source port number.
remoteIP	Destination IP address.
rport	Destination port number.

Action: This may indicate an internal error. Save all output and contact Customer Support.

T01IP057E IP output MBUF shortage for *proto*, *IPid*, *locapIP:lport remoteIP:rport*.

Reason: IP is unable to obtain MBUF storage to construct an output IP datagram.

proto	Protocol number.
IPid	IP identifier number.
localIP	Source IP address.
lport	Source port number.
remoteIP	Destination IP address.
rport	Destination port number.

Action: Check the logs for other evidence of a storage shortage. It may be necessary to increase the region size for the TCP/IP job.

T01IP059E Invalid IP header from interface *interface_name*. MBUF length *dgram_length* (decimal).

Reason: A datagram was received by IP with an invalid length field. One of the following was found to be in error:

- The IP header length in the IP header was less than the minimum (20 bytes)
- The packet length in the IP header was less than the IP header length
- The buffer (MBUF) length of the datagram was less than the minimum length for an IP packet (20 bytes)
- The buffer (MBUF) length of the datagram was less than the packet length in the IP header

The first 64 bytes of the datagram are dumped following the message. An ICMP parameter error is returned to the sending host. The packet is discarded.

interface_name	Interface name of the interface that presented the datagram to IP.
dgram_length	Length, in decimal, of the datagram

Action: Examine the remote host for errors.

T01IP060W IP host unreachable for *proto*, *IPid*, *localIP:lport remoteIP:rport*.

Reason: A datagram could not be delivered by an LNI. ARP time out is the probable cause.

proto	Protocol number.
IPid	IP identifier number.
lport	Source port number.
rport	Destination port number.
localIP	Source IP address.
remoteIP	Destination IP address.

Action: Examine the remote host.

T01IP070W ICMP datagram *localIP:lport remoteIP:rport*, unable to route.

Reason: An ICMP datagram was sent for which a route could not be obtained.

lport	Source port number.
rport	Destination port number.
localIP	Source IP address.
remoteIP	Destination IP address.

Action: Examine the network for routing problems.

T01IP071E IP header follows: *hex_dump_format*

Reason: This message contains a dump of an input IP datagram header. It follows another IP error message, such as T01IP010E , T01IP011E , T01IP018E , or T01IP019E .

Action: See the accompanying error message description for appropriate action.

Chapter 23

T01IU*nnn* IUCV Messages

This section describes messages issued by the IUCV routine. These messages include T01IU000 through T01IU099.

T01IU*nnn*

T01IU001I Connection to IUCV established - subsystem name *subsys_name*

Reason: Connection to IUCV was established successfully.

Action: None. This is an Informational message.

T02IU002I Connection to IUCV shut down normally

Reason: The IUCV connection was shut down normally.

Action: None. This is an Informational message.

T01IU003I Connection to IUCV terminated

Reason: The connection to IUCV was terminated.

Action: None. This is an Informational message.

T01IU004E Connection to IUCV deferred: subsystem name *subsys_name* not compatible

Reason: An unrecognized IUCV is already running.

Action: Bring down the other IUCV and start the IUCV for this product.

T01IU005E Connection to IUCV deferred

Reason: The IUCV address space has not yet been started.

Action: Start the IUCV address space.

T01IU006W An abend occurred processing SAPI CB *cb_name* in module *mod_name* at address *addr*

Reason: An ABEND occurred while the SAPI control block was being processed.

cb_name	Name of control block name that was processing.
mod_name	Module name.
addr	Address of module.

Action: Retain the dump and contact Customer Support.

T01IU007I IUCV address space has shutdown - subsystem name: *subsysid*

Reason: The IUCV subsystem *subsysid* has terminated. An attempt will be made to reconnect with a new IUCV address space every 30 seconds. *ssss* is the four-character VMCF subsystem name.

Action: None. This is an Informational message.

Chapter 24

T01LL*nnn* Link Layer Messages

This section describes messages issued by the Link Layer routine. These messages include T01LL000 through T01LL999.

T01LL*nnn*

In the following series of messages, device generally refers to the physical channel interface used to connect a network interface to an MVS host. The term interface generally refers to the logical endpoint within a particular network interface.

For instance, a 3172-type device has a single channel interlace and form one to four LAN statements. The device in this case represents the channel interface, and each LAN statement represents an interface. Deactivation of the channel path renders all interfaces useless, whereas, deactivating of a single interface does not affect the device or the other interfaces that share the same device.

For devices such as 3745 running the CDLC protocol or a 3762 running CETI, the distinction between the device and the interface is less clear.

From an operational viewpoint, the LNI STOP NAME(*if_name*) command attempts to halt communications with an interface by sending to it the proper shutdown commands. If the device is experiencing problems, the shutdown commands may not be sent and the LNI STOP command fails to complete. The DEVSTOP NAME(*dev_name*), however, deals at the subchannel level, and results in a Halt Subchannel being issued to the targeted device, ensuring that all communications with the device are stopped.

T01LL001I Activation of virtual interface complete

Reason: The virtual driver has completed its initialization sequence successfully. IP packets may now be sent from one application on the host to another application on the same host.

Action: None. This is an Informational message.

T01LL002I Deactivation of virtual interface complete

Reason: The virtual driver completed its termination sequence. IP packets can no longer be sent between an application on the host to another application on the same host.

Action: None. This is an Informational message.

T01LL100E Interface *if_name*: program *prog_name* encountered an ILATCH *call_type* error for latch *latch_name*, reason code = *rsn_code*

Reason: Program *prog_name* attempted to perform an ILATCH operation for latch name *latch_name*, but the call failed with reason code *rsn_code*. As a result, interface *if_name* may no longer operate correctly.

if_name Name of the interface.

prog_name Name of the program to be interfaced to.

call_type Type of latch call is one of the following:

- GET
- GETSHR
- FREE
- FREESHR

latch_name Name of latch calling.

rsn_code Reason code.

Action: If an SVCDUMP was produced, save the dump and contact Customer Support.

It may be necessary to stop, LNI STOP NAME(*if_name*), and restart the interface, LNI START NAME(*if_name*) to make the it operational again. If the LNI STOP command does not stop the interface, use the device stop command instead, DEV STOP NAME(*if_name*).

T01LL101E Media *med_name*: program *prog_name* encountered an ILATCH *call_type* error for latch *latch_name*, reason code - *rsn_code*

Reason: Program *prog_name* attempted to perform an ILATCH operation, for latch *latch_name*. The ILATCH call failed with reason code *rsn_code*. As a result, one or more interfaces associated with media *med_name* may no longer operate correctly.

med_name Media name.

prog_name Program name.

call_type Type of latch call is one of the following:

- GET
- GETSHR
- FREE
- FREESHR

latch_name Latch name.

rsn_code Reason code.

Action: If an SVCDUJP was produced, save the dump and contact Customer Support.

It may be necessary to stop, LNI STOP NAME(*if_name*), and restart each interface, LNI START NAME(*if_name*), associated with the media to make the media fully operational. If the LNI STOP command does not stop an interface, use the device stop command instead, DEV STOP NAME(*if_name*).

T01LL102E Device *dev_name*: program *prog_name* encountered an ILATCH *call_type* error for latch *latch_name*, reason code - *rsn_code*

Reason: Program *prog_name* attempted to perform an ILATCH operation for latch name *latch_name*, but the call failed with reason code *rsn_code*. As a result, interface *if_name* may no longer operate correctly.

dev_name Device name.

prog_name Name of the program to be interfaced to.

call_type Type of latch call, which will be one of the following:

- GET
- GETSHR
- FREE
- FREESHR

latch_name Name of latch calling.

rsn_code Reason code.

Action: If an SVCDUMP was produced, save the dump and contact Customer Support.

It may be necessary to stop, DEV STOP NAME(*if_name*), and restart the device, DEV START NAME(*dev_name*) to make the it operational again. If the device in question is a 3172 or its equivalent, the DEV STOP command may affect more than one interface.

T01LL103W Media *med_name*: ARP conflict: IP address *ip_addr* is being used by host with hardware address *hw_addr*

Reason: During the initialization sequence of an interface, an Address Request Protocol (ARP) message is sent on the local network to see if any other host is using the same IP address as the MVS host. If another host responds to this message, indicating the MVS host's IP address is its IP address, this message is issued.

This message implies that two hosts are using the same IP address. Depending on timing and the routing of IP datagrams, all sessions may be lost or experience severe degradation of service while both hosts are active.

Action: An IP address can only be assigned to a single active host at any point in time. Locate the remote host and change the IP address on either the remote host or the local MVS host.

T01LL104I Device *dev_name*: automatic device restart has been initiated: Restart timer expiration

Reason: The named device is being restarted because the restart timer associated with the device expired. The restart time is controlled by the RESTART(*time*) keyword on the configuration statement for the device in configuration member TCPCFG*xx*.

Action: If an automatic restart is desired, no action is necessary. If the restart interval is too long or too short, change the related device statement in TCPCFG*xx*. To prevent the device from automatically restarting, issue command LNI STOP NAME(*if_name*) for each interface associated with this device (3172s may have up to four interfaces associated with a device).

T01LL105E Device *dev_name*: automatic device restart failed: ISRB IGPOL call failed, reason code - *rsn_code*

Reason: An attempt was made to automatically restart a device, but insufficient virtual storage was available, preventing the auto restart.

Action: Reduce the virtual storage utilization in the region and issue the DEV START NAME(*dev_name*) command manually to restart the device.

T01LL106E Device *dev_name*: initialization sequence failed to complete (timeout). Device shutdown initiated

Reason: The named device failed to complete its initialization sequence in a timely manner. In other words, 30 seconds passed and the device ailed to complete the full exchange of startup messages.

Action: A problem may exist with the device in question. Resetting the device may clear the problem. If the problem continues after resetting the device, use the TCPEEP utility to trace the exchange of startup messages.

T01LL107E Device *dev_name*: automatic device shutdown failed: ISRB IGPOL call failed, reason code - *rsn_code*

Reason: An attempt was made to automatically shut down a device, but insufficient virtual storage was available, preventing the auto shutdown. The lack of virtual storage may explain why the device was shut down.

Action: Reduce the virtual storage utilization in the region and issue the DEV STOP NAME(*dev_name*) command manually to shut down the device.

T01LL108E Device *dev_name*: I/O interrupt missing on device *unit_addr*, device shutdown initiated

Reason: An I/O operation was started to the named device on subchannel address *unit_addr*. The I/O operation was expected to complete but did not. Shutdown processing is being initiated for the device for this reason.

Action: Check the device to determine the reason it failed to complete the I/O operation in a timely manner. Restart the device with the DEV START NAME(*dev_name*) command when the problem is corrected.

T01LL109E Device *dev_name*: automatic device shutdown failed: ISRB IGPOL call failed, reason code = *rsn_code*

Reason: An attempt was made to automatically shut down a device, but insufficient virtual storage was available, preventing the auto shutdown. The lack of virtual storage in the region may explain why the device was experiencing problems and is now being shut down.

Action: Reduce the virtual storage utilization in the region and issue the DEV STOP NAME(*dev_name*) command manually to shut down the device.

T01LL110E Device *dev_name*: device is failing to respond to timing commands, device shutdown initiated

Reason: Certain types of devices, including the named device, support commands that are sent periodically from the host to the device, to determine if the device is still functioning properly. The device usually sends a response to these commands, and as a result, the host knows that at least the channel interface of the device is still fully functional. In this particular instance, the device failed to respond to the command sent by the host.

Action: Check the device to determine why it failed to respond to the command sent to it by the host. If there is a possibility that the device is responding but the host is failing to recognize the response, use a TCPEEP interface trace or a GTF trace to diagnose the error on the host.

T01LL111E Device *dev_name*: automatic device shutdown failed: ISRB IGPOL call failed, reason code = *rsn_code*

Reason: An attempt was made to automatically shut down a device, but insufficient virtual storage was available, preventing the auto shutdown. The lack of virtual storage in the region may explain why the device was experiencing problems and is now being shut down.

Action: Reduce the virtual storage utilization in the region and issue the DEV STOP NAME(*dev_name*) command manually to shut down the device.

T01LL112I Interface *if_name*: automatic LNI restart has been initiated: Restart time expiration

Reason: The named interface is being restarted because the restart timer associated with the interface has expired. The restart time is controlled by the RESTART(time) keyword on the configuration statement for the interface in configuration member TCPCFG*xx*.

Action: If automatic restart is desired, no action is necessary. If the restart interval is too long or too short, change the related interface statement in TCPCFG*xx*. To prevent the interface from automatically restarting, issue the command LNI STOP NAME(*if_name*) for the interface.

T01LL113I Interface *if_name*: automatic LNI restart failed: ISRB IGPOL call failed, reason code - *rsn_code*

Reason: An attempt was made to automatically restart an interface, but insufficient virtual storage was available, preventing the auto restart.

Action: Reduce the virtual storage utilization in the region and issue the LNI START NAME(*if_name*) command manually to restart the device.

T01LL114E Interface *if_name*: initialization sequence failed to complete (timeout). LNI shutdown initiated

Reason: The named interface failed to complete its initialization sequence in a timely manner. In other words, 60 seconds passed and the interface failed to complete the full exchange of startup messages.

Action: A problem may exist with the interface in question. Resetting the interface may clear this problem. If the problem continues after resetting the interface, the TCPEEP utility can be used to trace the exchange of startup messages.

T01LL115E Interface *if_name*: automatic LNI shutdown failed: ISRB IGPOL call failed, reason code = *rsn_code*

Reason: An attempt was made to automatically shut down an interface, but insufficient virtual storage was available, preventing the auto shutdown. The lack of virtual storage may explain why the interface was shut down.

Action: Reduce the virtual storage utilization in the region and issue the LNI STOP NAME(*dev_name*) command manually to shut down the device.

T01LL116W Interface *if_name*: interface appears to be idle, initiating polling of local routers / hosts

Reason: Too much time passed since the last packet was received on the named interface. At this point, the interface does not know if this is due to an idle network, or whether the interface is experiencing technical problems. To determine why no packets were received, the interface tries to actively solicit responses from other devices on the local network.

If no response can be solicited from any device on the local network, message T01LL117W is issued and the interface is shut down.

Action: The IDLENET keyword of the Media statement within configuration member TCPCFG*xx* controls the period an interface waits while idle before polling. Adjust this value to meet the traffic conditions of the local network.

If this message is issued when a network is not idle, check for interface problems and correct if necessary.

T01LL117W Interface *if_name*: Idle polling of local routers / hosts failed to solicit a response

Reason: After message T01LL116W is issued, the named interface tries to solicit a response from a device on the local network. After reaching the query limit set by the IDLENET keyword of the Media statement, the interface shuts down to see if resetting the interface corrects the lack of input messages. The interface will be restarted automatically.

Action: Determine if the lack of input messages to the interface was caused by a network with no other active devices or whether the lack of messages was caused by a malfunctioning controller. If the network is truly idle, the IDLENET keyword of the Media statement in TCPCFG*xx* can be set to an idle time of zero to disable this feature. If the network is active but the controller is malfunctioning, fix the controller.

T01LL118I Interface *if_name*: shutdown of idle LNI initiated

Reason: See Message T01LL117W .

Action: See Message T01LL117W .

T01LL119I Interface *if_name*: automatic LNI shutdown failed: ISRB IGPOL call failed, reason code = *rsn_code*

Reason: An attempt was made to automatically shut down an interface, but insufficient virtual storage was available, preventing the auto shutdown. The lack of virtual storage may explain why the interface was shut down.

Action: Reduce the virtual storage utilization in the region and issue the LNI STOP NAME(*dev_name*) command manually to shut down the device.

T01LL120E Device *dev_name*: Purge I/O SVC call failed, reason code = *rsn_code*. I/O may not have been halted

Reason: The termination sequence for the named device was initiated, but the Purge I/O SVC call failed with the specified reason code. The reason codes from purge are documented in IBM's *DFSMS Advanced Services*.

If I/O was still active after the Purge SVC failed, an ABEND may occur when the I/O completes due to freed control blocks being accessed.

Action: Using the Purge SVC reason code, determine why the Purge SVC failed. If the problem is caused by site-specific conditions, correct the problem. If the problem is due to a TCP/IP software defect, contact Customer Support.

T01LL121E Device *dev_name*: initialization failed: timer allocation error, reason code = *rsn_code*

Reason: During initialization of the named device, the Itime service failed to allocate a new timer element (ITIM) to be used by the device and its associated interfaces. Device initialization cannot continue.

Action: Use the reason code supplied by the Itime service to determine why the ITIM allocation request failed.

T01LL122E Device *dev_name*: initialization failed: Save area pool allocation error, reason code = *rsn_code*

Reason: During initialization of the named device, a Storage Obtain request for a save area pool failed. The reason code from Storage Obtain is documented in IBM's *Authorized Assembler Services Reference SET-WTO*. The save area pool is allocated from subpool 91 for a length of 4608 bytes.

Action: Use the reason code from Storage Obtain to determine why the request for private storage failed. If there is insufficient storage in the region, change the region size for the current step in the TCP/IP JCL and restart TCP/IP.

T01LL123E Device *dev_name*: initialization failed: LNIO pool allocation error, reason code = *rsn_code*

Reason: During initialization of the named device, a Storage Obtain request for the LNIO pool failed. The reason code from Storage Obtain is documented in IBM 's *Authorized Assembler Services Reference SET-WTO*. The LNIO pool is allocated from subpool 230. The requested length in bytes is equal to the count of input and output buffers for the device times 256.

Action: Use the reason code from Storage Obtain to determine why the request for private storage failed. If there is insufficient storage in the region, change the region size for the current step in the TCP/IP JCL and restart TCP/IP.

T01LL124E Device *dev_name*: initialization failed: Buffer pool allocation error, reason code = *rsn_code*

Reason: During initialization of the named device, a Storage Obtain request for the I/O buffer pool failed. The reason code from Storage Obtain is documented in IBM 's *Authorized Assembler Services Reference SET-WTO*. The I/O buffer pool is allocated from subpool 79. The requested length in bytes is equal to the count of input buffers times the input buffer size, plus the count of output buffers times the output buffer size. For CETI devices, add an additional 112 bytes to this total.

I/O Buffer sizes in bytes are:

CDLC	MTU size specified on related Media + 4.
CETI	1518.
Claw	Size defined by the BUFSIZE keyword on the CLAW statement.
Hyperchannel	MTU size specified on related Media + 64.
LCS	20480.

Action: Use the reason code from Storage Obtain to determine why the request for private storage failed. If there is insufficient storage in the region, change the region size for the current step in the TCP/IP JCL and restart TCP/IP. If the error occurred due to too many I/O buffers being requested, change the IBUF and OBUF values in TCPCFG*xx* to reflect more appropriate values.

T01LL125E Device *dev_name*: initialization failed: LNSI pool allocation error, reason code = *rsn_code*

Reason: During initialization of the named device, a Storage Obtain request for the LNSI pool failed. The reason code from Storage Obtain is documented in IBM's *Authorized Assembler Services Reference SET-WTO*. The LNSI pool is allocated from subpool 245, above the 16 megabyte line, for a length of 256 bytes times the sum of the input and output buffers.

Action: Use the reason code from Storage Obtain to determine why the request for global system storage failed. If there is insufficient storage in Extended SQA, change your system configuration to allocate more extended global storage or fix the application or system service which is exhausting extended global storage.

T01LL126E Device *dev_name*: initialization failed: Unable to build initial read channel program

Reason: An internal logic error occurred while trying to build the channel program for the read subchannel for the named device.

Action: Contact Customer Support.

T01LL127E Device *dev_name*: initialization failed: attention handler registration failed, reason code = *rsn_code*

Reason: The named device requires use of an unsolicited attention handler to operate correctly. The attention handler manager, T01SUIHR, was unable to register the named device. Initialization of the device cannot continue.

Action: Use the reason code returned from T01SUIHR to diagnose and correct the problem. These reason codes are:

0 Successful.

4 Error accessing IEANTRT (possible link-edit error).

8 Non-zero return code from IEANTRT was returned. See IBM's *Authorized Assembler Services Reference ENF-ITT* for the return codes returned from IEANTRT.

12 No empty slots in unsolicited interrupt handler table (UIHT).

16 Load of T01PATTN failed. See the text of message T01LL154E for the reason code returned from the ILOAD (not LOAD) service.

20 Storage Obtain for UIHT failed (subpool 245, length 784 bytes). See the text of message T01LL153E and IBM's *Authorized Assembler Services Reference SET-WTO*, for the reason code returned from Storage Obtain.

24 Non-zero return code from NUCLKUP for module IOSVATTN. See the text of message T01LL152E and IBM's *Authorized Assembler Services Reference LLA-SDU* for the reason codes returned from NUCLKUP.

28 MVS system attention table is full.

32 Non-zero return code from IEANTCR was returned. See the text of message T01LL150E and IBM's *Authorized Assembler Services Reference ENF-ITT* for the reason codes returned from IEANTCR.

T01LL128E Device *dev_name*: initialization failed: UCBLOOK for unit *unit_addr* failed, ret=*ret_code*, rsn=*rsn_code*

Reason: The initialization of the named device failed because the system service UCBLOOK returned a non-zero return code, *ret_code*.

Action: Use the return code and reason code to diagnose why the dynamic allocation failed. The return code and reason codes from UCBLOOK are documented in IBM's *Authorized Assembler Services Reference SET-WTO*.

T01LL129E Device *dev_name*: initialization failed: SVC99 deallocation for unit *unit_addr* failed, ret=*ret_code*, rsn=*rsn_code*

Reason: The initialization of the named device failed because the dynamic allocation system service, SVC 99, failed. At the time of the error, SVC 99 was invoked to deallocate the specified unit. SVC 99 return values are documented in IBM's *Authorized Assembler Services Guide*.

Action: Use the return code, *ret_code*, and the reason code, *rsn_code*, to diagnose why the dynamic allocation failed.

T01LL130E Device *dev_name*: initialization failed: SVC99 allocation for unit *unit_addr* failed, ret=*ret_code*, rsn=*rsn_code*

Reason: The initialization of the named device failed because the dynamic allocation system service, SVC 99, failed. At the time of the error, SVC 99 was invoked to allocate the specified unit. SVC 99 return values are documented in IBM's *Authorized Assembler Services Guide*.

Action: Use the return code, *ret_code*, and the reason code, *rsn_code*, to diagnose why the dynamic allocation failed.

T01LL131E Device *dev_name* initialization failed: Unit *unit_addr* is not online

Reason: The initialization of the named device failed because the specified unit is not online to the MVS I/O subsystem.

Action: Vary the device online if possible and issue the DEV START NAME(*dev_name*) to restart the affected device.

T01LL132W Device *dev_name*: message to *ip_addr* discarded, length *msg_len* exceeds device limitations

Reason: An IP datagram of length *msg_len* was given to the named device for transmission. The length of the datagram exceeded the maximum length supported by the device and the IP datagram was discarded.

Action: Validate that the Media statement in configuration member TCPCFG*xx* has a MTU value that is valid for the named device.

T01LL133W Device *dev_name*: parameter error on call to T01SARSR, Reason=*rsn_code*

Reason: An internal logic error occurred on the call from T01SLCWR to T01SARSR, resulting in the discard of an IP datagram.

The following reason codes can be returned:

0C00	Storage Obtain failed.
0C01	No unicast address was supplied.
0C02	ARP not supported by media (probable configuration error).
0C03	Buffer too small.

Action: Contact Customer Support.

T01LL134E Device *dev_name:* error building control channel program

Reason: An internal logic error occurred while trying to build the channel program for the control subchannel for the named device.

Action: Contact Customer Support.

T01LL135I Device *dev_name:* interface now operational

Reason: The named device has received the first initialization message from the network controller indicating that the device is operational. The initialization sequence for the device continues.

Action: None. This is an Informational message.

T01LL136E Device *dev_name*: send buffer mismatch, configured=*cnfg_size*, supported=*sup_size*

Reason: Initialization for the named CDLC device failed because the configured buffer size (Media statement MTU size) does not match the value defined in the CDLC device.

Action: Either correct the MTU size on the Media statement in configuration member TCPCFG*xx* or correct the value defined in the CDLC device.

T01LL137I Device *dev_name*: device now active

Reason: The named device has completed its device level initialization sequence. Although the device has completed initialization, no IP datagrams may be sent or received via the device until one or more associated interfaces completes its initialization sequence.

Action: None. This is an Informational message.

T01LL138E Device *dev_name*: CDLC type device not ready, sense=*sense_data*

Reason: The initialization sequence for the named CDLC device failed, because the CDLC device is in an invalid state. The sense data returned by the device indicates the current CDLC state. The sense data is documented in IBM's *NCP VxRx and EP Rx Reference*.

Action: Use the sense data to determine which state the CDLC device is currently in. Once the state is determined, perform the necessary actions on the CDLC device to place it in the proper state required for TCP/IP initialization.

T01LL139E Device *dev_name*: Unknown CCW command code (*hex_cmnd_code*) on control operation interrupt

Reason: The CDLC device driver for the named device was unable to recognize the channel command code used in the last I/O operation. This is most likely the result of an internal logic error.

Action: Contact Customer Support.

T01LL140E Device *dev_name*: I/O error, Unit *unit_addr*, IOSCOD=*post_code*, OSFCSW=*chnl_stat_word*, sense=*sense_data*

Reason:

An I/O error occurred on subchannel *unit_addr* for the named device. Sense data was supplied by the device. IOSCOD is the MVS/ESA IOS postcode recorded in the IOSB and documented in IBM's *MVS/ESA Data Areas, Vol 2 (DCCB-ITTCTE)*. The channel status word is the value presented by the MVS subchannel at the time of I/O completion and is also recorded in the IOSB. The channel status word is documented in IBM's *ESA/390 Principles of Operation*. The sense data returned is specific to the device and can usually be located with documentation supplied by the hardware vendor.

Action:

Use the channel status word, *post_code* and *sense_data* to determine why the I/O error occurred. If the problem is due to a hardware error, correct the error and restart the device with a DEV START NAME(*dev_name*) command.

If the problem is due to a software problem:

1. Issue command F TCPIP,LOGGING PRINT((LL,ALL))
2. Run GTF
3. Recreate the problem
4. Contact Customer Support

Use the following options when starting GTF:

```
TRACE=IOP,SSCHP,HSCH,CSCH,MSCH,CCWP,PCI

IO=SSCH=(unit_addr)
CCW=(CCWN=255,DATA=0,PCITAB=2)
END
```

T01LL141E Device *dev_name*: I/O error, Unit *unit_addr*, IOSCOD=*post_code*, IOSFCSW=*chnl_stat_word*

Reason: An I/O error occurred on *subchannel unit_addr* for the named device. Sense data was not supplied by the device. IOSCOD is the MVS/ESA IOS postcode recorded in the IOSB and documented in IBM's *MVS/ESA Data Areas, Vol 2 (DCCB-ITTCTE)*. The channel status word is the value presented by the MVS subchannel at the time of I/O completion and is also recorded in the IOSB. The channel status word is documented in IBM's *ESA/390 Principles of Operation*.

Action: Use the channel status word and postcode to determine why the I/O error occurred. If the problem is due to a hardware error, correct the error and restart the device with a DEV START NAME(*dev_name*) command.

If the problem is due to a software problem:

1. Issue command F TCPIP,LOGGING PRINT((LL,ALL)).
2. Run GTF.
3. Recreate the problem.
4. Contact Customer Support.

Use the following options when starting GTF:

```
TRACE=IOP,SSCHP,HSCH,CSCH,MSCH,CCWP,PCI

IO=SSCH=(unit_addr)
CCW=(CCWN=255,DATA=0,PCITAB=2)
END
```

T01LL142E Device *dev_name*: device shutdown initiated due to I/O error

Reason: An I/O error occurred on the named device and as a result, the affected device is being shutdown.

Action: Locate previous error messages for an Reason of the I/O error. Perform the recommended action for the previous messages and issue command DEV START NAME(*dev_name*) when the condition causing the error is eliminated.

T01LL143E Device *dev_name*: automatic device shutdown failed: ISRB IGPOL call failed, reason code = *rsn_code*

Reason: An attempt was made to automatically shutdown a device, but insufficient virtual storage was available, preventing the auto shutdown. The lack of virtual storage in the region may explain why the device was experiencing problems and is now being shut down.

Action: Reduce the virtual storage utilization in the region and issue the DEV STOP NAME(*dev_name*) command manually to shutdown the device.

T01LL144E Device *dev_name*: I/O error, Unit *unit_addr*, dieRC=*die_rsn_code*, (ErrMsgBody) IOSCOD=*post_code*, IOSFCSW=*chnl_stat_word*, Sense=*sense_data*

Reason: An I/O error occurred on subchannel *unit_addr* for the named device. Sense data was supplied by the device. IOSCOD is the MVS/ESA IOS postcode recorded in the IOSB and documented in IBM's *MVS/ESA Data Areas, Vol 2 (DCCB-ITTCTE)*. The channel status word is the value presented by the MVS subchannel at the time of I/O completion and is also recorded in the IOSB. The channel status word is documented in IBM's *ESA/390 Principles of Operation*. The sense data returned is specific to the device and can usually be located with documentation supplied by the hardware vendor.

The disable interrupt exit reason code (*die_rsn_code*) identifies the exact error test that detected the error condition.

The DIE reason codes are:

2 No CCW address in CSW (device possibly reset).

3 CCW address less than lowest LNIO address.

4 CCW address higher than highest LNIO address.

5 LNIO ID validation failed.

6 LCS input: Unrecognized / invalid ending status.

7 LCS input: LNIO with ending status not first active LNIO.

8 LCS output: Unrecognized / invalid ending status.

9 CLAW input: Unrecognized / invalid ending status.

10 Claw output: Unrecognized / invalid ending status.

11 CDLC input: Unrecognized / invalid ending status.

12 CDLC input: Unknown operation code.

13 CDLC input: Missing DE on read.

14 CDLC input: Busy on read.

15 CDLC input: DE on Read Start zero or one.

16 CDLC input: Unrecognized status on Read Start zero or one.

17 CDLC output: Unrecognized / invalid ending status.

18 CDLC output: Unknown operation code.

19 CDLC output: Chaining error (DE on write).

20 CDLC output: Unrecognized status on Write.

21 CDLC output: Unrecognized status on Write Break.

23 CDLC output: Chaining error (DE on Write Start zero or one).

24 CDLC output: Unrecognized status on Write Start zero or one.

40 Hyperchannel input: Unrecognized / invalid ending status.

41 Hyperchannel input: LNIO with ending status not first active LNIO.

42 Hyperchannel output: Unrecognized / invalid ending status.

50 CETI control: Any ending status.

54 CETI attention: Status other than ATTN or UC.

58 CETI input: Any ending status.

62 CETI output: Any ending status.

Action:

Use the channel status word, postcode, sense data, and DIE reason code to determine why the I/O error occurred. If the problem is due to a hardware error, correct the error and restart the device with a DEV START NAME(*dev_name*) command.

If the problem is due to a software problem:

1. Issue command F TCPIP,LOGGING PRINT((LL,ALL)).
2. Run GTF.
3. Recreate the problem.
4. Contact Customer Support.

Use the following options when starting GTF:

```
TRACE=IOP,SSCHP,HSCH,CSCH,MSCH,CCWP,PCI

IO=SSCH=(unit_addr)
CCW=(CCWN=255,DATA=0,PCITAB=2)
END
```

T01LL145E Device *dev_name*: I/O error, Unit *unit_addr*, dieRC=*die_rsn_code*, IOSCOD=*post_code*, IOSFCSW=*chnl_stat_word*

Reason: An I/O error occurred on subchannel *unit_addr* for the named device. Sense data was not supplied. IOSCOD is the MVS/ESA IOS postcode recorded in the IOSB and documented in IBM's *MVS/ESA Data Areas, Vol 2 (DCCB-ITTCTE)*. The channel status word is the value presented by the MVS subchannel at the time of I/O completion and is also recorded in the IOSB. The channel status word is documented in IBM's *ESA/390 Principles of Operation*. The disable interrupt exit reason code (*die_rsn_code*) identifies the exact error test that detected the error condition. These codes are documented in message T01LL144E.

Action: Use the channel status word, postcode and DIE reason code to determine why the I/O error occurred. If the problem is due to a hardware error, correct the error and restart the device with a DEV START NAME(*dev_name*) command.

If the problem is due to a software problem:

1. Issue command F TCPIP,LOGGING PRINT((LL,ALL)).
2. Run GTF.
3. Recreate the problem.
4. Contact Customer Support.

Use the following options when starting GTF:

```
TRACE=IOP,SSCHP,HSCH,CSCH,MSCH,CCWP,PCI

IO=SSCH=(unit_addr)
CCW=(CCWN=255,DATA=0,PCITAB=2)
END
```

T01LL146I Device *dev_name*: initialization in progress

Reason: Device initialization is in progress for *dev_name*.

Action: None. This is an Informational message.

T01LL147I interface *if_name*: initialization in progress

Reason: Interface initialization is in progress for the *if_name*.

Action: None. This is an Informational message.

T01LL148I Device *dev_name*: termination in progress

Reason: Device termination is in progress.

Action: None. This is an Informational message.

T01LL149E Device *dev_name*: termination error: SVC99 deallocation for unit *unit_addr* failed, ret=*ret_code*, rsn=*rsn_code*

Reason: The termination of the named device failed because the dynamic allocation system service, SVC 99, failed. At the time of the error, SVC 99 was invoked to deallocate the specified unit. SVC 99 return values are documented in IBM's *Authorized Assembler Services Guide*.

Action: Use the return code, *ret_code*, and the reason code, *rsn_code*, to diagnose why the dynamic de-allocation failed.

T01LL150E Attention table manager: named token create failed, reason code=*rsn_code*

Reason: The named device requires use of an unsolicited attention handler to operate correctly. The attention handler manager, T01SUIHR, was unable to register the named device due to an error returned by the MVS system service IEANTCR service. Initialization of the device cannot continue. The reason codes returned by IEANTCR are documented in IBM's *Authorized Assembler Services Reference ENF-ITT*.

Action: Use the reason code returned from IEANTCR to diagnose and correct the problem.

T01LL151E Attention table manager: system attention table Is Full

Reason: The named device requires use of an unsolicited attention handler to operate correctly. The attention handler manager, T01SUIHR, was unable to register the named device because all system wide attention table slots are in use. Initialization of the device cannot continue.

Action: If the attention table can be expanded via a user mod or PTF from IBM, expand the attention table. Otherwise, get a network device that does not require use of the attention table.

T01LL152E Attention table manager: nucleus lookup for IOSVATTN failed, reason code=*rsn_code*

Reason: The named device requires use of an unsolicited attention handler to operate correctly. The attention handler manager, T01SUIHR, was unable to register the named device because a call to MVS system service NUCLKUP failed. Initialization of the device cannot continue. The reason codes returned by the NUCLKUP service are document in IBM's *Authorized Assembler Services Reference LLA-SDU*.

Action: Use the reason code returned from NUCLKUP to diagnose and correct the problem.

T01LL153E Attention table manager: Unable to allocate attention Table, reason code=*rsn_code*

Reason: The named device requires use of an unsolicited attention handler to operate correctly. The attention handler manager, T01SUIHR, was unable to register the named device because a call to MVS system service NUCLKUP failed. Initialization of the device cannot continue. The reason codes returned by the NUCLKUP service are documented in IBM's *Authorized Assembler Services Reference LLA-SDU*.

Action: Use the reason code returned from NUCLKUP to diagnose and correct the problem.

T01LL154E Attention table manager: load of module T01PATTN failed, reason code=*rsn_code*

Reason: The named device requires use of an unsolicited attention handler to operate correctly. The attention handler manager, T01SUIHR, was unable to register the named device due to a non-zero return code being returned by the ILOAD service. Initialization of the device cannot continue. The ILOAD service was attempting to load module T01PATTN into system subpool 245 storage.

Action: Use the reason code returned from the ILOAD service to diagnose and correct the problem. This error may be caused by:

- Insufficient extended global storage and may require modifications to system parameters to increase the amount of global virtual storage available for application usage
- Module T01PATTN not existing in a data set that is part of the TCP/IP STEPLIB concatenation.

T01LL155E Attention table manager: attention table is full

Reason: The named device requires use of an unsolicited attention handler to operate correctly. The attention handler manager, T01SUIHR, was unable to register the named device because its private attention table is full. Initialization of the device cannot continue. The private attention table is capable of holding information for 32 separate devices.

Action: Contact Customer Support and ask them to provide a new version of module T01SUIHR that allocates a private attention table with a larger capacity.

T01LL156E Attention table manager: named token retrieve failed, reason code=*rsn_code*

Reason: The named device requires use of an unsolicited attention handler to operate correctly. The attention handler manager, T01SUIHR, was unable to register the named device due to a non-zero reason code being returned by the MVS system service IEANTRT. Initialization of the device cannot continue. The reason codes returned by IEANTCR are documented in IBM's *Authorized Assembler Services Reference ENF-ITT*.

Action: Use the reason code returned from IEANTRT to diagnose and correct the problem.

T01LL157E Attention table manager: product installation error, T01SUIHR not link edited with IEANTRT

Reason: The named device requires use of an unsolicited attention handler to operate correctly. The attention handler manager, T01SUIHR, was unable to register the named device because module T01SUIHR was not properly link edited with module IEANTRT at TCP/IP installation time. Initialization of the device cannot continue.

Action: Correct the SMP/E error condition that caused the original link edit to fail and then link edit T01SUIHR with module IEANTRT, which can typically be located in library SYS1.CSSLIB.

T01LL158I Device *dev_name*: termination complete

Reason: The termination sequence for the name device successfully completed. All resources used by the device have been freed.

Action: None. This is an Informational message.

T01LL159I Interface *if_name*: termination complete

Reason: The termination sequence for the name device successfully completed. All resources used by the interface have been freed. If multiple interfaces share the same device, the device may still be holding resources for the other interfaces. If this is the case, use the DEV STOP NAME(*dev_name*) command to stop the device.

Action: None. This is an Informational message.

T01LL161W Interface *if_name*: init request ignored, device *dev_name* has not completed shutting down

Reason: An LNI STOP command was issued, but the named device associated with the targeted interface is still in its termination phase. The command was ignored.

Action: Wait for the device to complete termination or investigate why the device is failing to terminate in a timely manner.

T01LL162W Interface *if_name*: init ignored, device *dev_name* has not completed initialization

Reason: An LNI START command was issued, but the named device associated with the targeted interface is still in its initialization phase. The command was ignored.

Action: Wait for the device to complete initialization or investigate why the device is failing to initialize in a timely manner.

T01LL163E Interface *if_name*: init request failed (Device *dev_name* init failed, reason code *rsn_code*)

Reason: A request to initialize the named interface required the initialization of the associated device. The device initialization failed.

Action: Look for other error messages for the named device that explain why the initialization of that device failed.

T01LL164W Interface *if_name*: init request ignored, interface has not completed shutting down

Reason: A request to initialize the named interface was ignored because the interface has not completely finished a previous shutdown sequence.

Action: The interface can be forced down by issuing the DEV STOP NAME(*dev_name*) command. Then the interface can be restarted by issuing the LNI START NAME(*if_name*) command.

Note: Be aware that some physical intervention with the network controller may be necessary if the controller is malfunctioning.

T01LL165W interface *if_name*: init ignored, prior init request has not completed

Reason: A request to initialize the named interface was ignored because the interface was already in the process of initializing.

Action: Wait for interface initialization to complete. If interface initialization fails to complete, issue the DEV STOP NAME(*dev_name*) command to shutdown the interface and device, correct any error with the network controller, and then issue the LNI START NAME(*if_name*) command to restart the interface.

T01LL166W interface *if_name*: init request ignored, interface is operational

Reason: A request to initialize the named interface was ignored because the interface was fully operational.

T01LL167W Interface *if_name*: init request failed, unable to build initial channel program

Reason: An internal logic error occurred while trying to build a channel program for the named interface.

Action: Contact Customer Support.

T01LL168W Device *dev_name*: unit *unit_addr* is not defined as a channel to channel adapter

Reason: All network controllers with the exception of CDLC and Hyperchannel network controllers must be defined to the MVS I/O subsystem as a Channel-To-Channel Adapter (CTCA). The named device is not defined to MVS as a CTCA.

Action: If this message is issued when the unit address specified on the device statement in configuration member TCPCFG*xx* is incorrect. Change the unit address on the statement and restart TCP. If the unit address is correct, change the unit definition within MVS, defining the network controller as a CTCA.

T01LL169I Device *dev_name*: Allocation conflict: unit *unit_addr* may be offline, or allocated to another application

Reason: During the initialization sequence of the named device, the dynamic allocation of unit_addr failed with SVC 99 error code x'0214' or x'0244', indicating that the device was either offline or allocated to another application.

Action: Validate that the correct unit address was specified on the device statement in configuration member TCPCFG*xx*. If the unit address is correct, use the MVS command D U,,ALLOC,*unit_addr* to determine the which application is currently using the named device. Once you make this determination, reconfigure either this TCP/IP stack or the other application, so that only one application uses the device at a time.

T01LL170W Interface *if_name*: shutdown request ignored, device *dev_name* is already down

Reason: A request to shutdown the named interface was ignored because the associated device was already shut down.

T01LL171W Interface *if_name*: shutdown request ignored, device *dev_name* is already shutting down

Reason: A request to shutdown the named interface was ignored because the associated device was already in the process of shutting down.

T01LL172W Interface *if_name*: shutdown request ignored, device *dev_name* has not completed initialization

Reason: A request to shutdown the named interface was ignored because the associated device has not completed its initialization sequence.

Action: If the device appears to be hung, issue the DEV STOP NAME(*dev_name*) command to shut down the device immediately.

T01LL173W Interface *if_name*: shutdown request ignored, interface is not active

Reason: A request to shutdown the named interface was ignored because the interface is not currently active.

T01LL174W Interface *if_name*: shutdown request ignored, interface is already shutting down

Reason: A request to shutdown the named interface was ignored because the interface already started shutdown processing.

Action: If shutdown processing of the interface appears to have stalled, issue the DEV STOP NAME(*dev_name*) command to immediately shutdown the device and all of its associated interfaces.

T01LL175I Interface *if_name*: no active links, device *dev_name* is being shut down

Reason: A request to shutdown a CLAW interface has found no active CLAW links. As a result, the underlying CLAW device is being shutdown also.

Action: None. This is an Informational message.

T01LL176I Interface *if_name*: shut down in progress

Reason: A request to shutdown the named interface was accepted and shutdown processing for the interface is now active.

Action: None. This is an Informational message.

T01LL177D Device *dev_name*: LNDC Storage dump *mem_dump*

Reason: This diagnostic message is produced following an I/O error. The storage being dumped is the Local Network Device Control (LNDC) block. The LNDC is used to keep track of the state of a particular device, and all its resources.

Action: This information is supplied as part of diagnostic data when an I/O error occurs. The intended audience for this data is Customer Support. Customers should make this data available to Customer Support when reporting I/O error problems.

T01LL178E Device *dev_name*: LNSI Storage dump *mem_dump*

Reason: This diagnostic message is produced following an I/O error. The storage being dumped is the LNI SRB / IOSB (LNSI) block. The IOSB is the control block used to communicate information about I/O requests between TCP/IP and the MVS I/O Subsystem. The IOSB is located at offset x'48' in the LNSI and its layout is documented in IBM's *MVS/ESA Data Areas, Vol 2 (DCCB-ITTCTE)*.

Action: This information is supplied as part of diagnostic data when an I/O error occurs. The intended audience for this data is Customer Support. Customers should make this data available to Customer Support when reporting I/O error problems.

T01LL179D Device *dev_name*: LNIO Storage dump *mem_dump*

Reason: This diagnostic message is produced following an I/O error. The storage being dumped is the LNI Input / Output (LNIO) block. The LNIO contains the Channel Command Words (CCWs) and Indirect Addressing Word (IDAW) lists.

Action: This information is supplied as part of diagnostic data when an I/O error occurs. The intended audience for this data is Customer Support. Customers should make this data available to Customer Support when reporting I/O error problems.

T01LL180I Device *dev_name*: interface *if_name* is now active

Reason: The named interface completed its initial hand shaking sequence. Depending on the device type, the initialization sequence may require more data exchanges to make the interface fully operational. At that point, IP datagrams can be sent and received.

Action: None. This is an Informational message.

T01LL181I Device *dev_name*: interface *if_name* is fully operational

Reason: All processing required to activate the named interface completed successfully. The interface is now capable of sending and receiving IP datagrams.

Action: None. This is an Informational message.

T01LL182I Device *dev_name*: device is now active

Reason: All processing required to activate the named device completed successfully. No IP datagrams may be sent or received, however, until one or more associated interfaces complete its initialization sequence and becomes fully operational.

Action: None. This is an Informational message.

T01LL183I Interface *if_name*: interface initialization in progress

Reason: The initialization process has begun for the named interface.

Action: None. This is an Informational message.

T01LL184W Device *dev_name*: device shutdown notification received

Reason: A 3172-type device received a shutdown notification from the network controller. The device and all of its interfaces are immediately shutdown. Automatic restart of the device will occur when the time interval specified by the RESTART keyword of the LCS statement in configuration member TCPCFG*xx* expires.

Action: If the shutdown message was received unexpectedly, check the network controller to determine why the shutdown message was sent to the host.

T01LL185W Device *dev_name*: interface *if_name* has been stopped

Reason: This message is issued following message T01LL184W or T01LL186E, to identify each interface that is being shutdown automatically.

Action: Perform the recommended action for the previously issued error or warning message.

T01LL186E Device *dev_name*: interface *if_name* command (*3172_cmnd*) to controller failed

Reason: A 3172 type interface or device was sent the command code *3172_cmnd*. The command, however, was returned by the network controller in error.

Action: Try to determine why the network controller rejected the command. Customers may need to contact Customer Support for assistance in identifying the meaning of the *3172_cmnd* code and the expected response from the device.

T01LL187I Device *dev_name*: device is now active

Reason: All processing required to activate the named device completed successfully. No IP datagrams may be sent or received, however, until one or more associated interfaces complete its initialization sequence and becomes fully operational.

Action: None. This is an Informational message.

T01LL188I Device *dev_name*: TCP/IP link has been established, link id = *link_id*

Reason: During the initialization of a CLAW device, the primary CLAW link used to transmit IP datagrams was established. The link ID is for use in identifying the link when running network traces.

Action: None. This is an Informational message.

T01LL189I Device *dev_name*: checksum assist link has been established, link id = *link_id*

Reason: During the initialization of a CLAW device, the secondary CLAW link used to transmit TCP/IP datagrams that have offloaded check summing processing performed on them was established. The link ID is for use in identifying the link when running network traces.

Action: None. This is an Informational message.

T01LL190I Device *dev_name*: packetizing link has been established, link id = *link_id*

Reason: During the initialization of a CLAW device, the packed CLAW link used to transmit multiple IP datagrams was established. The packed link cannot actually transmit or receive IP datagrams until the TCP/IP and checksum sub-links are established.

Action: None. This is an Informational message.

T01LL191W Device *dev_name*: Claw link *link_id* has been stopped

Reason: The named device stopped processing for the specified CLAW link. The type of link can be identified by locating the previous T01LL188I, T01LL189I or T01LL190I message.

Action: If the stopping of the link was unexpected, determine why the network control stopped the link.

T01LL192I Device *dev_name*: interface *if_name* is fully operational

Reason: All processing required to activate the named interface completed successfully. The interface is now capable of sending and receiving IP datagrams.

Action: None. This is an Informational message.

T01LL193E Device *dev_name*: checksum Assist feature is not supported

Reason: The Media statement in the configuration member TCPCFG*xx*, was coded with keyword Assist, NoCheckSum, or OffLoadCkSum. The named device, however, is not capable of performing offloaded checksum processing.

Action: If the network controller is incapable of performing offloaded checksum processing, change the Media statement in TCPCFG*xx* to include option NoAssist, CheckSum or HostCkSum. If the network controller does support offloaded checksum processing, verify that the network controller is properly configured to perform this service on the subchannel addresses being used by the device.

T01LL194E Device *dev_name*: packetizing feature is not supported

Reason: The CLAW statement in the configuration member TCPCFG*xx*, was coded with keyword Packed. The named device, however, is not capable of packing multiple datagrams into a single channel I/O buffer.

Action: If the network controller is incapable of performing packed I/O buffering, change the CLAW statement in TCPCFG*xx*, removing the Packed keyword. If the network controller does support packed I/O buffering, verify that the network controller is properly configured to perform this service on the subchannel addresses being used by the device.

T01LL195E Device *dev_name*: Claw error: command *claw_cmnd*, error *claw_err*

Reason: The named CLAW device received a CLAW protocol error for the specified CLAW command code. CLAW command codes and CLAW error codes are documented in IBM's *Block Multiplexer Channel Adapter: User's Guide and Service Information* or *Enterprise Systems Connection Adapter: User's Guide and Service Information*.

Action: Use the supplied information to determine why the CLAW command was rejected by the network controller. If the problem appears to be TCP/IP software related, contact Customer Support.

T01LL196E Device *dev_name*: automatic device shutdown failed: ISRB IGPOL call failed, reason code = *rsn_code*

Reason: An attempt was made to automatically shutdown a device, but insufficient virtual storage was available, preventing the auto shutdown. The lack of virtual storage in the region may explain why the device was experiencing problems and is now being shutdown.

Action: Reduce the virtual storage utilization in the region and issue the DEV STOP NAME(*dev_name*) command manually to shutdown the device.

T01LL198I Device *NAME*:the CLAW statement does not reflect the controller configuration PACKED option

Reason: The controller is configured for the PACKED option, but the CLAW statement in the TCPCFG*xx* member does not contain the PACKED option keyword. Both configuration options must match. The controller shuts down.

Action: Correct either the startup configuration member, or reconfigure the controller to match.

T01LL199I Device *dev-name*: Awaiting contact from partner CTC device

Reason: Normal start-up condition when one side is activated before the other. Message repeats at 90-second intervals.

Action: None. If you'd like to eliminate this message, bring up the other side.

T01LL200E Device *dev-name* received invalid command.

Reason: The device received a read, write, or prepare command out of sequence. This is most likely caused by a configuration mismatch. The read-to-write reversal required for the ends of the CTC connection may not be correct.

Action: Correct the configuration member and restart.

T01LL201I Media *med_name* is now operational with 1 or more active interfaces

Reason: The named Media was defined with one or more interfaces. At least one of these interfaces is now fully functional. As a result, the IP addresses associated with this media can now be accessed from remote hosts.

Action: None. This is an Informational message.

T01LL202W Media *med_name* is now inoperable due to all attached interfaces being inoperational

Reason: The named media was defined with one or more interfaces. All of these interfaces are now shutdown or in the process of shutting down. As a result, the IP addresses associated with this media most likely cannot be accessed from remote hosts unless some other form of IP routing via a different media is occurring.

Action: If the desired state of the interfaces owned by this media is not down, then issue the LNI START NAME(*if_name*) command to start one or more of the interfaces owned by this media.

T01LL203E Device *dev_name* Level 2 error *level2_err_hex - level2_err_english*

Reason: The named CETI device reported a level two error. The error code is printed in hexadecimal, along with its English translation. IBM's *IEEE 802.3 Local Area Network Reference Manual* documents the CETI protocol. The error message is received and documented, but no further action is performed by the device driver.

Action: If the CETI interface is malfunctioning, this error message may help explain the cause of the malfunction. Otherwise, no action is required.

T01LL204W Device *dev_name* Excessive collisions

Reason: The named CETI device detected excessive collisions on the attached Ethernet. This may be the cause of TCP/IP network performance problems.

Action: If excessive collisions are affecting network performance, then reconfiguration of the local area network may be necessary to reduce the collision rate.

T01LL205W Device *dev_name* Out of window collisions

Reason: The named CETI device detected excessive out of window collisions on the attached Ethernet. This may be the cause of TCP/IP network performance problems.

Action: If excessive out of window collisions are affecting network performance, then reconfiguration of the local area network may be necessary to reduce the collision rate.

T01LL206W Device *dev_name* Alignment errors

Reason: The named CETI device detected excessive alignment errors on the attached Ethernet. This may be the cause of TCP/IP network performance problems.

Action: If excessive alignment errors are affecting network performance, then locate and correct the device on the Ethernet causing the alignment errors.

T01LL207W Device *dev_name* Lost Ethernet carrier

Reason: The named CETI device detected that the Ethernet carrier was lost. This is most likely caused by a loose cable or a cut cable.

Action: Locate and fix the break in the physical medium.

T01LL208E Device *device*: Input buffer contains bad next offset in LNIH at offset *curr_offset*. Next = *next_offset*.

Reason: An input buffer was received from a network interface device with an invalid offset.

One of the following was found:

1. The offset of the next message in the buffer exceeds the maximum buffer size for the device.
2. The offset of the next message is less than the offset of the current message.

The remainder of the buffer is discarded. This message is accompanied by message T01LL209D, which dumps the contents of the remainder of the buffer.

device	Name of device that presented the buffer.
curr_offset	Offset of current message.
next_offset	Offset of next message.

Action: Examine the device for errors.

T01LL209D Device *device*: Buffer Storage Dump

Reason: This message accompanies message T01LL0208E. See the description of that message for details.

T01LL210W Device *dev_name* MAC address 0 not allowed

Reason: The named LCS device returned a hardware address of all zeros during the initialization sequence. Since this represents a broadcast address, it is not allowed.

Action: Check the *Customization Guide* LINK Usage Notes for known device issues and follow those recommendations. If there are no recommendations for the failing device, capture a TCPEEP GROUPS('NETIF') FORMAT(DATA(96)) trace and contact Customer Support.

T01LL211W WLM Service: *WLM_groupname* WLM Registration | Deregistration Failed; *ret_code*, reason=*reason_code*

Reason: An attempt to WLM register or deregister the TCP/IP stack or a service application with the MVS Workload Manager. An error was encountered when either the WLM service IWMSRSRG to register or IWMSRDRS to deregister was invoked. Refer to OS/390 V2R5.0 MVS Workload Management Service for Reason of error and reason codes.

Action: Call Customer Support and report the problem.

WLM_groupname	The WLM Groupname.
Registration | Deregistration	The message specifies registration or deregistration.
ret_code	WLM general return code.
reason_code	WLM reason code.

T01LL212W Device *dev_name* Unit *unit_addr* Dynamic allocation information field contains unexpected value. Value *info_field*

Reason: Dynamic allocation discovered a non-zero value in the S99INFO field. This value is reported in the *info_field*. Device allocation proceeds.

Action: No action is required. If problems occur, this value may provide diagnostic assistance.

T01LL213E Device *dev_name* Unit *unit_addr* Dynamic allocation, UCBASID not allocated to our ASID. Value *asid_value*

Reason: Dynamic allocation discovered the ASID to which the device was allocated was not the ASID of the TCP stack. Allocation fails.

Action: This problem is usually associated with SMS. Make sure that SMS excludes the device.

T01LL214E *Interface* if_name*: initialization failed: IXCJOIN for member* member_name *failed, ret*=ret_code*, rsn*=rsn_code

Reason: The initialization of the named interface failed because the system service IXCJOIN returned a non-zero return code. In the message, *if_name* is the name of the interface that failed initialization, *member_name* is the XCF member name, and *ret_code* and *rsn_code* are the return code and reason code, respectively, from the IXCJOIN system service. Note that *if_name* also identifies the XCF group name.

Action: Use the return code and reason code to diagnose the IXCJOIN system service failure. The IXCJOIN return and reason codes are documented in the IBM publication *MVS Programming: Sysplex Services Reference*.

T01LL215E Interface *if_name*: initialization failed: IXCQUERY failed, ret=*ret_code*, rsn=*rsn_code*

Reason: The initialization of the named interface failed because the system service IXCQUERY returned a non-zero return code. In the message, *if_name* is the name of the interface that failed initialization and *ret_code* and *rsn_code* are the return code and reason code, respectively, from the IXCQUERY system service. Note that *if_name* also identifies the XCF group name.

Action: Use the return code and reason code to diagnose the IXCQUERY system service failure. The IXCQUERY return and reason codes are documented in the IBM publication *MVS Programming: Sysplex Services Reference*.

T01LL216E Interface *if_name*: initialization failed: IXCSETUS failed, ret=*ret_code*, rsn=*rsn_code*

Reason: The initialization of the named interface failed because the system service IXCSETUS returned a non-zero return code. In the message, *if_name* is the name of the interface that failed initialization and *ret_code* and *rsn_code* are the return code and reason code, respectively, from the IXCSETUS system service. Note that *if_name* also identifies the XCF group name.

Action: Use the return code and reason code to diagnose the IXCSETUS system service failure. The IXCSETUS return and reason codes are documented in the IBM publication *MVS Programming: Sysplex Services Reference*.

T01LL217I Interface *if_name*: initialization complete

Reason: The initialization of the named interface has completed.

Action: None. This is an Informational message.

T01LL218E Interface *if_name*: termination error: IXCSETUS failed, ret=*ret_code*, rsn=*rsn_code*

Reason: The termination of the named interface failed because the system service IXCSETUS returned a non-zero return code. In the message, *if_name* is the name of the interface that failed termination and *ret_code* and *rsn_code* are the return code and reason code, respectively, from the IXCSETUS system service. Note that *if_name* also identifies the XCF group name.

Action: Use the return code and reason code to diagnose the IXCSETUS system service failure. The IXCSETUS return and reason codes are documented in the IBM publication *MVS Programming: Sysplex Services Reference*.

T01LL219E Interface *if_name*: termination error: IXCQUIES failed, ret=*ret_code*, rsn=*rsn_code*

Reason: The termination of the named interface failed because the system service IXCQUIES returned a non-zero return code. In the message, *if_name* is the name of the interface that failed termination and *ret_code* and *rsn_code* are the return code and reason code, respectively, from the IXCQUIES system service. Note that *if_name* also identifies the XCF group name.

Action: Use the return code and reason code to diagnose the IXCQUIES system service failure. The IXCQUIES return and reason codes are documented in the IBM publication *MVS Programming: Sysplex Services Reference*.

T01LL220E Interface *if_name*: termination error: IXCLEAVE failed, ret=*ret_code*, rsn=*rsn_code*

Reason: The termination of the named interface failed because the system service IXCLEAVE returned a non-zero return code. In the message, *if_name* is the name of the interface that failed termination and *ret_code* and *rsn_code* are the return code and reason code, respectively, from the IXCLEAVE system service. Note that *if_name* also identifies the XCF group name.

Action: Use the return code and reason code to diagnose the IXCLEAVE system service failure. The IXCLEAVE return and reason codes are documented in the IBM publication *MVS Programming: Sysplex Services Reference*.

T01LL221W Interface *if_name*: parameter error on call to T01SARSR, reason = *rsn_code*

Reason: An internal logic error occurred on the call from T01SLCFW to T01SARSR, resulting in the discard of an IP datagram. The following reason codes can be returned:

0C00—Storage Obtain failed.

0C01—No unicast address was supplied.

0C02—ARP not supported by media (probable configuration error).

0C03—Buffer too small.

Action: Contact Customer Support.

T01LL222E Interface *if_name*: data transfer error: IXCMSGO failed, ret=*ret_code*, rsn=*rsn_code*

Reason: A data transfer error occurred on the named interface because the system SERVICE IXCMSGO returned a non-zero return code. In the message, *if_name* is the name of the interface on which the error occurred and *ret_code* and *rsn_code* are the return code and reason code, respectively, from the IXCMSGO system service. Note that *if_name* also identifies the XCF group name.

Action: Use the return code and reason code to diagnose the IXCMSGO system service failure. The IXCMSGO return and reason codes are documented in the IBM publication *MVS Programming: Sysplex Services Reference*.

Note: It may be necessary to stop, LNI STOP NAME(*if_name*), and restart the interface, LNI START NAME(*if_name*), to make it fully operational again.

T01LL223E Interface *if_name*: data transfer error: IXCMSGI failed, ret=*ret_code*, rsn=*rsn_code*

Reason: A data transfer error occurred on the named interface because the system SERVICE IXCMSGI returned a non-zero return code. In the message, *if_name* is the name of the interface on which the error occurred and *ret_code* and *rsn_code* are the return code and reason code, respectively, from the IXCMSGI system service. Note that *if_name* also identifies the XCF group name.

Action: Use the return code and reason code to diagnose the IXCMSGI system service failure. The IXCMSGI return and reason codes are documented in the IBM publication *MVS Programming: Sysplex Services Reference*.

Note: It may be necessary to stop, LNI STOP NAME(*if_name*), and restart the interface, LNI START NAME(*if_name*), to make it fully operational again.

T01LL224E Unable to obtain storage for ISRB scheduling service – ret=*ret_code*.

Reason: Internal error. The XCF group exit was unable to obtain storage required for scheduling the IXCQUERY supplemental support service. *ret_code* is the failing return code.

Action: Contact Customer Support.

T01LL225E Unable to schedule module *module_name* – ret=*ret_code*.

Reason: Internal error. The XCF group exit was unable to schedule required service routine *module_name*. *ret_code* is the failing return code.

Action: Contact Customer Support.

T01LL226E IXCQUERYservice request failed – ret=*ret_code*, rsn=*rsn_code*.

Reason: Internal error. The IXCQUERY supplemental support service failed. *ret_code* and rsn=*rsn_code* are the failing return/reason codes. These codes are documented in IBM publication *MVS Programming:Sysplex Services Reference* under "IXCQUERY return and reason codes."

Action: Contact Customer Support.

Chapter

25 T01LU*nnn* LU Pool Messages

This chapter describes messages concerning LU Pool facilities. These messages include T01LU000 through T01LU999.

T01LU*nnn*

T01LU001E No LU available in LUPOOL:*lupool_name* for user:*user_id* IP=*remoteIP* LURULE command number=*rec_num*

Reason: The LU pool facility attempted to acquire an LU name from an LU pool (*lupool_name*) for a user (*user_id*) from a remote host (*remoteIP*). All LU names identified in the LURULE statement (*rec_num*) are already allocated to other users. The request to access the VTAM application is denied and the user is returned to the Telnet prompt.

Action: User can retry the Telnet command to acquire any deallocated LU name. The system administrator can add additional LU names to the LU pool identified in the message.

T01LU002E No applicable LURULE for LU assignment for user:*user_id* IP=*remoteIP* resource = *resource_name*

Reason: The LU pool facility was attempting to acquire an LU name for a user (*user_id*) from a remote host (*remoteIP*) for resource (*resource_name*). The remote user could not match the criteria for any LURULE rule. The request to access the VTAM application is denied and the user is returned to the Telnet prompt.

Action: Contact the system administrator and request a rule to access host applications.

T01LU003E Lupool facility not active due to initialization error

Reason: During initialization of the APP task group, an error was encountered during LU pool processing of the LU pool configuration file APPLUP*xx*. No users are allowed access to host applications until the problem with the LU pool configuration is resolved. Other messages are displayed prior to this message describing the errors encountered while processing the configuration file.

Action: Contact the system administrator to request correction of the problem. The system administrator must review the operator console log for messages describing the reason for initialization failure. After the problem is resolved, from the operator console issue APP REFRESH LUPARM(*mem_name).*

T01LU004E Associated Printer LU for terminal LU:*term_lu* IP:*ip_addr* not available.

Reason: An associated printer (*ip_addr*) was not specified for terminal LU *term_lu*.

Action: Specify an associated printer for the LU in question in a TN3270E pool.

T01LU005E LU:*term_lu* not available from LUPOOL:*pool_name* IP:*ip_addr*

Reason: A remote TN3270E user requested a specific terminal LU (*term_lu*) but the LU was not available. The LU name is either not defined in an LU pool (*pool_name*) or the LU is already in session. The remote terminal user is given by *ip_addr*.

Action: Verify that the LU is defined in an LU pool. If it is defined, the remote can try to get the LU again later when it may become available.

Chapter 26

T01NT*nnn* NETSTAT Messages

This section describes messages issued by NETSTAT. These messages include T01NT000 through T01NT099.

T01NT*nnn*

T01NT000I Request complete

Reason: This message is displayed at the end of a multi-line response to indicate that the request completed.

Action: None. This is an Informational message.

T01NT001E PC call lookup for T01CCMND failed

Reason: Internal error.

Action: Repeat the command. If the error is repeated, contact Customer Support.

T01NT002E Invalid syntax. Enter NETSTAT HELP for command description.

Reason: The command entered was not recognized.

Action: Verify the command and reenter or use the NETSTAT HELP command for more information on the parameters for NETSTAT commands.

T01NT003E Session *sess_num* not found

Reason: The session number given in the command was invalid.

Action: Verify the session number and reenter.

T01NT004E Port number *port_num* not found

Reason: The port number given in the NETSTAT command was invalid.

Action: Verify the port number and reenter the command.

T01NT005W No matching selection criteria for session cancellation

Reason: The parameters specified on the command line did not match an existing session.

T01NT006E MEDIA name *med_name* not found

Reason: The media name specified in the NETSTAT command was invalid.

Action: Verify the correct media name and reenter the command.

T01NT007I proto session *sess_num* cancelled. *local_ip:local_port <--> rem_ip:rem_port*

Reason: This is a confirmation of a valid cancel. proto gives the protocol in use. sess_num gives the session number that was cancelled. The local and remote IP addresses and ports are also given.

Action: None. This is an Informational message.

T01NT008E Attempt to get *latch* latch failed with RC=*ret_code*

Reason: This is a general latch failure. *latch* can be SAVXLTCH, SAVXTLAT, SAVXNETL, SAVXMSTR. The *ret_code* is the error returned from the latch code.

Action: Contact Customer Support if the condition persists.

T01NT009I T01SNETS IGPOL failure. RC=*ret_code*

Reason: This is an internal failure.

Action: Contact Customer Support if the condition persists.

T01NT012E Request failed; incomplete results

Reason: The NETSTAT command failed to complete. This can occur if the display is very long, if the environment is changing while the command is processing, or a storage outage is detected. In the event of a storage outage, it will be preceded by T01TN018.

Action: Repeat the command. Contact Customer Support if the condition persists.

T01NT013E Service request failed; code *code*

Reason: Internal error.

Action: Contact Customer Support if the condition persists.

T01NT014E Service not responding

Reason: Internal error.

Action: Contact Customer Support if the condition persists.

T01NT015E PCORE storage request failed

Reason: Internal error.

Action: Contact Customer Support if the condition persists.

T01NT016E The configuration is inadequate to service this request

Reason: The NETSTAT command was issued too early and the product task group is in the process of coming up.

Action: Wait until the product is fully operational, then reissue the command.

T01NT017E ALESERV failure - RC = *ret_code*

Reason: Internal error.

Action: Contact Customer Support if the condition persists.

T01NT018E STORAGE obtain failure - RC = *ret_code*

Reason: Internal error. The RC may be 4, 8, 12(X'C') or 16(X'10')

Return codes 4-12 result directly from the storage obtain request. Code 16 identifies an attempt to obtain storage outside the limits established by the MAXSTGPCT configuration parameter, and may at times be followed by message T01NT012.

Action: Contact Customer Support if the condition persists.

T01NT019E POST failure - RC = *ret_code*

Reason: Internal error.

Action: Contact Customer Support if the condition persists.

T01NT020I Job *job_name* processing:*request*

Reason: This message gives the *job_name* that is processing the *request* specified on the command line.

Action: None. This is an Informational message.

T01NT021I *help parameter description*

Reason: This message is displayed in response to a NETSTAT HELP command. The text for this message displays a description of the parameter and is followed by one or more messages describing the syntax. The text for this message varies for each parameter displayed.

T01NT022I *help parameter syntax*

Reason: This message displays in response to a NETSTAT HELP command. It is preceded by a message that gives a description of a NETSTAT parameter. This message gives the syntax for each parameter and may be one or more lines. The text for this message varies for each parameter displayed.

Action: None. This is an Informational message.

T01NT023I *ipaddr <--> ipname*

Reason: This message is in response to the NETSTAT RESOLVE command.

ipaddr The IP address in dotted notation.

ipname The IP name character string if the resolution is successful.

Action: None. This is an Informational message.

T01NT024I *addr lni_name* local MAC address: *cont_hw_addr*

Reason: This message is a response to a NETSTAT ARP request. It identifies the local environment. *addr* is the IP address of the local host, *lni_name* gives the local LNI description and *cont_hw_addr* is the hardware address of the controller.

Action: None. This is an Informational message.

T01NT025I IP Address..... MAC Address. Flag

Reason: This message is in response to a NETSTAT ARP request. It is the header for the following message (T01NT026I).

Action: None. This is an Informational message.

T01NT026I *ip_addr mac_addr flag*

Reason: This message is in response to a NETSTAT ARP request. It lists the IP address, hardware address and flag for each entry in the ARP tables. Flags are: V (valid), D (dead), S (static), ? (if in process of being resolved).

Action: None. This is an Informational message.

T01NT027I *request_type* cannot be resolved

Reason: This message is in response to the NETSTAT RESOLVE command. *request_type* is the IP name or IP address that could not be resolved.

Action: None. This is an Informational message.

T01NT028I Protocol:*proto* Maximum listen queue: *num*

Reason: This message is a response to the NETSTAT CNFG command by protocol. It gives the network protocol and maximum value for the listen queue.

Action: None. This is an Informational message.

T01NT029I Default queue sizes : send=*send_num* receive=*rcv_num*

Reason: This message is a response to the NETSTAT CNFG command. It gives the default sizes for the send and receive queues.

Action: None. This is an Informational message.

T01NT030I Maximum queue sizes : send=*send_num* receive=*rcv_num*

Reason: This message is a response to the NETSTAT CNFG command. It gives the maximum sizes for the send and receive queues.

Action: None. This is an Informational message.

T01NT031I Default buffer sizes: send=*send_num* receive=*rcv_num*

Reason: This message is a response to the NETSTAT CNFG command. It gives the default sizes for the send and receive buffers.

Action: None. This is an Informational message.

T01NT032I Maximum buffer sizes: send=*send_num* receive=*rcv_num*

Reason: This message is a response to the NETSTAT CNFG command. It gives the maximum size for the send and receive buffers.

Action: None. This is an Informational message.

T01NT033I Maximum data sizes : send=*send_num* receive=*rcv_num*

Reason: This message is a response to the NETSTAT CNFG command. It gives the maximum data sizes for the send and receive data.

Action: None. This is an Informational message.

T01NT034I IP addr: *ip_addr name*

Reason: Displays the IP address in both dot-notation and host name. This is part of the NETSTAT CNFG LNI display.

Action: None. This is an Informational message.

T01NT035I Network: address=*addr* mask=*mask network_name*

Reason: This is part of the NETSTAT CNFG LNI display and shows the IP address, network mask, and network name.

Action: None. This is an Informational message.

T01NT036I Subnet : address=*ip_addr* mask=*subnet_mask*

Reason: This is part of the NETSTAT CNFG LIN display and shows the IP address and subnet mask.

Action: None. This is an Informational message.

T01NT037I ARP timeouts: routers=*rout_time* sec others=*other_time* sec

Reason: This is part of the NETSTAT CNFG LNI display and shows the router time for ARP timeout response. *other_time* is for non-router time for an ARP response.

Action: None. This is an Informational message.

T01NT038I Idle network: timeout=*timeout* sec queries=*num_queries*

Reason: This is part of the NETSTAT CNFG LNI display and is for idle network detection. *timeout* is the dormant network time and *num_queries* is the number of queries sent after the timeout expires.

Action: None. This is an Informational message.

T01NT039I LNI=*lni_name* MAC=*hw_addr* status=*status control_block_addr*

Reason: This is part of the NETSTAT CNFG LNI display. It gives the LNI name, the hardware address, the status of the connection and the control block address.

Action: None. This is an Informational message.

T01NT040I DEV=*dev_name* Adptr *adapt_num addr_devctl_block* status=*dev_status* mode *mode*

Reason: This is part of the NETSTAT CNFG LNI display. It gives the device name, the adapter number, address device control block, the device status and mode of operation for the hardware. The device name is different from the LNI name for the IBM 3172 which has multiple adapters. In the event of a CTC LNI, one of the following will be displayed:

Rd/wr mode -if the DEVADDR CCUU is configured as the read device.

Wr/rd mode -if the DEVADDR CCUU is configured as the write device.

Action: None. This is an Informational message.

T01NT041I In :*bytes_in* bytes *pkts_in* packets *num_discards* discards

Reason: This message is in response to the NETSTAT CNFG command. It gives the number of bytes, packets, and discards received on the connection described just above. It is preceded by messages T01NT060I, T01NT061I and T01NT062I and followed by T01NT042I. *discards* are not errors.

Action: None. This is an Informational message.

T01NT042I Out: *bytes_out* bytes *pkts_out* packets *discards* discards

Reason: This message is in response to the NETSTAT CNFG command. It gives the number of bytes, packets, and discards sent the connection described just above. It is preceded by messages T01NT041I, T01NT060I, T01NT061I and T01NT062I. *discards* are not errors.

Action: None. This is an Informational message.

T01NT043I Destination Router Flags Intfc Mask

Reason: This message is displayed in response to the NETSTAT ROUTE command when the MASK option is selected, instead of messages T01NT044I and T01NT045I. It is a header for the information that follows in message T01NT048I.

Action: None. This is an Informational message.

T01NT044I Destination Router Flags Ref Use Intfc

Reason: This message is in response to the NETSTAT ROUTE command. It is a header for the information that follows (T01NT045I).

Action: None. This is an Informational message.

T01NT045I *ip_addr ip_addr flags outgoing_cum outgoing_curr int_net_name*

Reason: This message is in response to the NETSTAT ROUTE command. It is preceded by the T01NT044I message that gives the header.

ip_addr	Destination.
flags	One of the following flags: U (up; if not shown, connection is down) G (gateway) H (host) D (dynamic entry) M (modified recently)
outgoing_cum	Cumulative count of outgoing packets.
outgoing_curr	Current count of outgoing packets.
int_net_name	Internal network name.

Action: None. This is an Informational message.

T01NT046I Session # IP address Lport:Rport Pro Appl SLU Elapsed

Reason: This message is a response to the NETSTAT USERS command. It is a header for the information that follows (T01NT047I).

Action: None. This is an Informational message.

T01NT047I *sess# ip_addr local_port:remote_port proto appl slu elapsed_time*

Reason: This message is in response to the NETSTAT USERS command. It is preceded by the header information (T01NT046I). It gives the session number, IP address, local and remote ports, and the protocol. The *appl* is the service name, *slu* is secondary lu, and *elapsed_time* gives the time that the connection has been up.

Action: None. This is an Informational message.

T01NT048I *ip_addr ip_addr flags int_net_name mask*

Reason: This message is in response to the NETSTAT ROUTE command when the MASK option is selected. It is preceded by the T01NT043I message that gives the header.

ip_addr Destination.

flags One of the following flags:

- U (up; if not shown, connection is down)
- G (gateway)
- H (host)
- D (dynamic entry)
- M (modified recently)

int_net_name Internal network name.

mask The subnet mask used by this route entry.

Action: None. This is an Informational message.

T01NT049I GateD configuration member *mem_name*

Reason: This is a response to the NETSTAT CNFG command and identifies the GateD configuration member in use.

Action: None. This is an Informational message.

T01NT050I IP: TOS=*type_of_service*. TTL=*time_to_live*. Reassembly timeout=*value* Forward/NoFWD MAXVIPA=*max_vipa*

Reason: This message is a response to the NETSTAT CNFG command. It shows configuration parameters specified on the IP statement of the TCPCFG*xx* configuration member. The parameters displayed are the type of service, time to live, reassembly timeout, forwarding indicator and the maximum number of active dynamic VIPAs.

Action: None. This is an Informational message.

T01NT052I The specified session is not a TCP session

Reason: This is in response to a CHANGE command. CHANGE is valid for TCP only.

Action: None. This is an Informational message.

T01NT053I *number* connected session(s) found

Reason: This is in response to the NETSTAT CONN COUNT command and gives the total number of connected sessions.

Action: None. This is an Informational message.

T01NT054I *protocol type ip_addr : port<-->ip_addr : port state*

Reason: This message is a response to the NETSTAT CONN command. It shows the active connection information. It gives the access method type, protocol, IP address, port number, and state of the connection. This message is followed by T01NT055I.

Possible values are shown in the following table:

protocol	UDP
	ICMP
	IP
	OSPF
	RAW
	TTCP
type	OE
	IUCV
	ATLI
	IUHP

Note: IUHP is displayed if you are using HPNS or if you are running with INTERNALVMCF (INTERNALIUCV) and you are using IUCV.

state

Null: Initial state or the connection has been closed.

Ground: Connection bound to a reserved port.

Bound: Connection bound to unreserved port.

Note: The following apply to TCP only:

Connecting: Connection request is waiting completion

Listening: Waiting for a connection request from any remote TCP and port.

Connected: Normal state for data transfer representing a state from/to that data can be transferred.

Sending: Connection is in a send only state.

Receiving: Connection is in a receive only state.

Dead: A data transfer connection is being closed.

Failed: Indicates that a socket close was tried for a connection that did not successfully connect.

Action: None. This is an Informational message.

T01NT055I Bytes: *bytes_in* in, *bytes_out* out. Ses# *sess_num* Idle *hh:min:sec* J=*job_name*

Reason: This message is a response to the NETSTAT CONN command and shows active connection information. It shows the number of bytes in and out, the session number, the time the connection was idle and the job name. It is preceded by message T01NT054I.

Action: None. This is an Informational message.

T01NT056I SEPM *ctl_blk_addr*, associated with SPCB *ctl_blk_addr*

Reason: This display is from the NETSTAT CONN command and is used for internal tracking of control blocks.

Action: None. This is an Informational message.

T01NT060I Media name *med_name* Type *type* MTU=*mut_size*

Reason: This message is a response to the NETSTAT CNFG command. It shows the media name of the configuration, the type, and the MTU for the configuration.

Action: None. This is an Informational message.

T01NT061I Network *network_name* IP address *ip_addr*

Reason: This message is a response to the NETSTAT CNFG command. It shows the network name and IP address for the configuration.

Action: None. This is an Informational message.

T01NT062I LNI *type* Device *device* Adapter *addr*

Reason: This message is a response to the NETSTAT CNFG command. It shows the LNI type, and the device and adapter information.

Action: None. This is an Informational message.

T01NT063I *

Reason: This is from the SEP parameter available on the NETSTAT CONN command and separates the output by placing a single asterisk (*) on a line to make the display more readable.

Action: None. This is an Informational message.

T01NT064I ReXMIT: MAXcount=*count* MINtime=*min_time* MAXtime = *max_time* RTO=*num*

Reason: This message is a response to the NETSTAT CNFG command. It gives the retransmit information for maximum count, minimum and maximum time for a retransmit and the RTO value set for the configuration.

Action: None. This is an Informational message.

T01NT065I DelayedACK=*num,num* FWidle =*idle* KEEPalive=*kal state*

Reason: This message is a response to the NETSTAT CNFG TCP command.

Action: None. This is an Informational message.

T01NT066I MINdev=*num* Round trip dev=*trip_num* HASH=*hash_val* Fast RX=*xmit*

Reason: This message is a response to the NETSTAT CNFG TCP command. It gives the minimum deviation, roundtrip deviation, hash values, and fast retransmission count.

Action: None. This is an Informational message.

T01NT067I SCALE = *scale* Close timeout = *close_time* Connect timeout= *conn_time*

Reason: This message is a response to the NETSTAT CNFG TCP command. It gives scales, close timeout, and connect timeout.

Action: None. This is an Informational message.

T01NT068I HH:MM:SS - *nnn* active TCP connections

Reason: This is the first of three RTM messages. It contains a time stamp and an active connection count.

Action: None. This is an Informational message.

T01NT069I LNI *aaaaaaaa*: *bbb*/*ccc* bytes/packets in, at *ddd* bytes/sec

Reason: Second RTM message.

aaaaaaaa	Hardware interface name
bbbb	Total byte count.
ccc	Total packet count.
ddd	Rate based on the time difference from the last sampling.

Action: None. This is an Informational message.

T01NT070I LNI aaaaaaaa: bbb/ccc/ bytes/packets out, at ddd bytes/sec

Reason: Third RTM message.

aaaaaaaa	Hardware interface name
bbbb	Total byte count.
ccc	Total packet count.
ddd	Rate based on the time difference from the last sampling.

Action: None. This is an Informational message.

T01NT071I Data bytes :*rel* released *retran* rexmitted'

Reason: This display is in response to the NETSTAT CONN SESS FULL command.

rel Incoming packets that can be ignored.

retran Number of retransmitted packets.

Action: None. This is an Informational message.

T01NT072I Packets :*num_in* in, *num_out* out

Reason: This display is in response to the NETSTAT CONN SESS FULL command. It gives the number of packets in and the number of packets out.

Action: None. This is an Informational message.

T01NT073I :*num_pkts* released *num_pkts* rexmitted

Reason: This message is the second part of T01NT072I and gives the number of packets released and retransmitted.

Action: None. This is an Informational message.

T01NT074I ACK packets :*num_pkts* in, *num_pkts* out, *dups* duplicates

Reason: This display is in response to the NETSTAT CONN SESS FULL command. It gives the number of ACK packets that have been received and sent and the number of duplicates received.

Action: None. This is an Informational message.

T01NT075I Errors :*chk_err* checksum errors, *format_err* format errs

Reason: This display is in response to the NETSTAT CONN SESS FULL command. It gives the number of checksum errors and format errors (non-checksum errors).

Action: None. This is an Informational message.

T01NT076I Sequence num:Initial send=*send_seq* Initial receive=*init_seq*

Reason: This display is in response to the NETSTAT CONN SESS FULL command. This display is only seen if the session requested is a TCP session. It is followed by T01NT077I - T01NT080I.

Action: None. This is an Informational message.

T01NT077I :Next send =*next_send* Next receive =*next_rcv*

Reason: This display is in response to the NETSTAT CONN SESS FULL command. This display is only seen if the session requested is a TCP session.

Action: None. This is an Informational message.

T01NT078I :Sent unacked=*sent* Sent window upd=*upd*

Reason: This display is in response to the NETSTAT CONN SESS FULL command. This display is only seen if the session requested is a TCP session.

Action: None. This is an Informational message.

T01NT079I :Highest sent=*seq_num* Rcvd window upd=*seq_num*

Reason: This display is in response to the NETSTAT CONN SESS FULL command. This display is only seen if the session requested is a TCP session.

Action: None. This is an Informational message.

T01NT080I :Being timed = *seq1* Last ACK sent = *seq2*

Reason: A NETSTAT command was used to display a TCP session.

seq1 Sequence number of the datagram being timed for retransmission.

seq2 Sequence number of the last ACK sent.

Action: None. This is an Informational message.

T01NT082I Window sizes:Send maximum=*max_send* Send current = *curr_val*

Reason: A NETSTAT command was used to display a TCP session.

max_send Maximum send window size.

curr_val Current send window size.

Action: None. This is an Informational message.

T01NT083I :Receive max = *max_send* Receive curr = *curr_val*

Reason: A NETSTAT command was used to display a TCP session.

max_send Maximum receive window for the connection.

curr_val Current receive window.

Action: None. This is an Informational message.

T01NT084I :Congestion = *win_size1* Available = *win_size2*

Reason: A NETSTAT command was used to display a TCP session.

win_size1 Congestion window size, as described by the Van Jacobson algorithm.

win_size2 Available window size.

Action: None. This is an Informational message.

T01NT085I :Threshold = *thresh* MSS = *snd_seg_size*

Reason: A NETSTAT command was used to display a TCP session.

thresh Threshold window size, as described by the Van Jacobson algorithm.

snd_seg_size Send message segment size for the session.

Action: None. This is an Informational message.

T01NT090I TCP variables:RT start = *rt_time* Scaled RT = *scaled_rt_time*

Reason: A NETSTAT command was used to display a TCP session.

rt_time Elapsed time for the round trip.

scaled_rt_time Scaled round trip time, as described by the Van Jacobson algorithm.

Action: None. This is an Informational message.

T01NT091I :TCP idle time = *idle_time* Scaled RT var = *rt_time*

Reason: A NETSTAT command was used to display a TCP session.

idle_time Amount of time the session has been idle.

rt_time Scaled round trip variance, as described by the Van Jacobson algorithm.

Action: None. This is an Informational message.

T01NT093I Timers: Retransmit = *retrans_timer* Persist = *persist_timer*

Reason: A NETSTAT command was used to display a TCP session.

retrans_timer Value for the retransmit timer.

persist_timer Value for the persist timer.

Action: None. This is an Informational message.

T01NT094I :Keepalive = *keepalive_timer* Timewait = *timewait_timer*

Reason: A NETSTAT command was used to display a TCP session.

keepalive_timer Time value for the keep alive timer.

timewait_timer Time value for the time wait timer.

Action: None. This is an Informational message.

T01NT095I :Delayed ACK = *ack_timer* Linger = *linger_timer*

Reason: A NETSTAT command was used to display a TCP session.

ack_timer Time value for the delayed ACK timer.

linger_timer Time value for the linger timer.

Action: None. This is an Informational message.

T01NT096I VIPANET: IP address=*ip_address* Subnet mask=*subnet_mask*

Reason: This message is a response to the NETSTAT CNFG command. It shows configuration parameters specified on the VIPANET statement of the TCPCFG*xx* configuration member. The parameters displayed are the IP address and subnet mask for a dynamic VIPA subnet.

Action: None. This is an informational message.

T01NT100I Session # IP address Appl PLU SLU Logmode

Reason: This message is a response to the NETSTAT TELNET command. It is a header for the information that follows (T01NT101I).

Action: None. This is an Informational message.

T01NT101I *sess# ip_addr appl plu slu log_mode*

Reason: This message is a response to the NETSTAT TELNET command. It is preceded by the header information (T01NT100). It shows the session number, IP address, application name, primary LU name, secondary LU name, and logon mode name.

Action: None. This is an Informational message.

T01NT102I State: *state* LPort: *local_port* RPort: *remote_port*

Reason: This message is a response to the NETSTAT TELNET command. It shows the session state, local port number, and remote port number.

state will be one of the following:

- Closed
- Listening
- SYN sent
- SYN received
- Established
- Close wait
- FIN wait 1
- Closing
- Last ACK
- FIN wait 2
- Time wait
- Unknown

Action: None. This is an Informational message.

T01NT103I BytesIn: *bytes_in* BytesOut: *bytes_out* Elapsed: *elapsed*

Reason: This message is a response to the NETSTAT TELNET command. It shows the number of bytes in and out and the session elapsed time.

Action: None. This is an Informational message.

T01NT104I TNProto: *protocol* TNUserID: *userid*

Reason: This message is a response to the NETSTAT TELNET command. It shows the Telnet protocol and user ID.

protocol will be one of the following:

- NVT
- TN3270
- TN3270E
- Unknown

Action: None. This is an Informational message.

T01NT105I RTM: Total:Under 1 sec = *total_count* IP:Under 1 sec = *IP_count*

Reason: This message is a response to the NETSTAT TELNET command. It shows response time measurement data identifying the number of interactions where the total response time was under one second, and the number of interactions where IP response time was under one second.

Action: None. This is an Informational message.

T01NT106I Under 2 sec = *total_count* IP:Under 2 sec = *IP_count*

Reason: This message is a response to the NETSTAT TELNET command. It shows response time measurement data identifying the number of interactions where the total response time was under two seconds, and the number of interactions where IP response time was under two seconds.

Action: None. This is an Informational message.

T01NT107I Under 5 sec = *total_count* IP:Under 5 sec = *IP_count*

Reason: This message is a response to the NETSTAT TELNET command. It shows response time measurement data identifying the number of interactions where the total response time was under five seconds, and the number of interactions where IP response time was under five seconds.

Action: None. This is an Informational message.

T01NT108I Under 10 sec= *total_count* IP:Under 10 sec= *IP_count*

Reason: This message is a response to the NETSTAT TELNET command. It shows response time measurement data identifying the number of interactions where the total response time was under 10 seconds, and the number of interactions where IP response time was under 10 seconds.

Action: None. This is an Informational message.

T01NT109I Over 10 sec= *total_count* IP:Over 10 sec= *IP_count*

Reason: This message is a response to the NETSTAT TELNET command. It shows response time measurement data identifying the number of interactions where the total response time was over 10 seconds, and the number of interactions where IP response time was over 10 seconds.

Action: None. This is an Informational message.

T01NT110I *Group Member IP Address Token*

Reason: This message is in response to the NETSTAT XCF command. It is a header for the information that follows (T01NT111I).

Action: None. This is an Informational message.

T01NT111I *group_name member_name ip_address member_token*

Reason: This message is in response to the NETSTAT XCF command. It is preceded by the header information (T01NT110I). It shows the XCF group name, XCF member name, IP address, and XCF member token.

Action: None. This is an Informational message.

T01NT112I *State: state Type: type Status: status*

Reason: This message is in response to the NETSTAT XCF command. It shows the XCF member state, member type, and member operational status.

state is one of the following:

- Not defined
- Created
- Active
- Quiesced
- Failed
- Unknown

type is a decimal number representing the member type.

status is one of the following:

- Down
- Shutting down
- Starting up
- Up
- Unknown

Action: None. This is an Informational message.

T01NT113I *System: system_name STC: stc_name*

Reason: This message is in response to the NETSTAT XCF command. It shows the system name that the member was last active on and the JOB, STC, MOUNT, or LOGON name from the primary address space when the member last joined the XCF group.

Action: None. This is an Informational message.

T01NT114I *bind_security_configuration_parameters*

Reason: This message is displayed in response to a NETSTAT CNFG BINDSEC command. The text of the message shows one or more bind security configuration parameters and will vary depending on the parameters displayed.

Action: None. This is an Informational message.

Chapter 27

T01OE*nnn* UNIX System Services Messages

This chapter describes messages written by the UNIX System Services interface, formerly OpenEdition. These messages are T01OE000 through T01OE999.

Messages in this range detail information about the network protocol layers, Transmission Control Protocol (TCP), Internet Protocol (IP), Internet Control Message Protocol (ICMP), and User Datagram Protocol (UDP). These messages help someone familiar with the TCP/IP suite of protocols to troubleshoot problems arising in this area on the local system, remote system, or network.

T01OE*nnn*

T01OE001E Call to module *mod_name*, *V=value* failed, rtrn value=*ret_code*, rsn code=*rsn_code*

Reason: A call to module *mod_name* failed with a return value of value, a return code of *ret_code*, and a reason code of *rsn_code*.

This could be a normal message if the module is BPXVOSIT and Unix System Services is not running at the time of the message.

Action: Check the logs for other related error messages. Let the other messages guide your action.

If *ret_code* is EMVSINITIAL (x'9C') and *rsn_code* is in the range of JRSAFnoUser (x'F9') - JRSAFInternal (x'FE'), the OpenEdition PFS will not be operational until the RACF configuration error is corrected and the region has been recycled.

If *ret_code* is EACCESS (x'6F'), and if the address family is AF_INET and the PFS is configured to work under the control of converged sockets (that is, it needs a master socket to operate), the OpenEdition PFS will not be operational until the RACF configuration error is corrected and the address space recycled.

T01OE003E LOAD of module *mod_name* failed, Abend code=*abend_code*, rsn code=*rsn_code*

Reason: A load request for module *mod_name* failed with a ABEND code of *abend_code* and a reason code of *rsn_code*.

Action: If OpenEdition is running at your installation and the module name is BPXVOSIT make sure its library is in the linklist.

T01OE004I Connection To OpenEdition Established -- Provider *provider*

Reason: A cross memory connection was established between the ACP task group and the OpenEdition address space to support Socket API functions. The name of the provider in the UNIX System Services address space is *provider.*

Action: None. This is an Informational message.

T01OE005I Connection To OpenEdition Shutdown -- Provider *provider*

Reason: The cross memory connection to UNIX System Services address space was shutdown normally.

Action: Determine why UNIX System Services terminated. Restart UNIX System Services to reestablish socket connectivity.

T01OE006I Connection To OpenEdition Terminated -- Provider *provider*

Reason: The cross memory connection to UNIX System Services address space was shutdown abortively.

Action: Determine why UNIX System Services terminated. Restart UNIX System Services to reestablish socket connectivity.

T01OE007E Unable to locate and call OpenEdition module: *mod_name*

Reason: An attempt to establish connection with UNIX System Services failed because the module indicated could not be located. This is an installation error.

Action: If the module name *mod_name* is BPX*mmmmm*, check installation of UNIX System Services.

T01OE008W An abend occurred processing SAPI CB *cccc* abend in module *mod_name*, Addr=*addr*.

Reason: Message generated for diagnostic purposes.

Action: Check for an SVC dump titled T01AMSCK - Abended T01AMSCK=00000000, MWA=00X000000, DB#A=000000000. Contact Customer Support.

T01OE101I TCP/IP PFS Name:*pfs_name* Initialization Started

Reason: Initialization of the UNIX System Services PFS named *pfs_name* has begun.

Action: None. This is an Informational message.

T01OE102I TCP/IP PFS Name:*pfs_name* Using *product_name* Subsystem Name:*subsys_name*

Reason: The UNIX System Services PFS named *pfs_name* is using the product address space with the subsystem identifier of *subsys_name*.

Action: None. This is an Informational message.

T01OE103I TCP/IP PFS Name:*pfs_name* Initialization Complete

Reason: Initialization of the UNIX System Services PFS named *pfs_name* successfully completed.

Action: None. This is an Informational message.

T01OE104I TCP/IP PFS Name:*pfs_name* Subtask:*tcb_address* Started

Reason: The UNIX System Services PFS named *pfs_name* attached a subtask with a Task Control Block (TCB) located at address *tcb_address* in the UNIX System Services address space.

Action: None. This is an Informational message.

T01OE105E TCP/IP PFS Name:*pfs_name config_type* Initialization Failed; PFS Terminating

Reason: Initialization of the UNIX System Services PFS named *pfs_name* failed.

Action: Check for any prior error messages issued from this PFS. Correct any incorrect configuration statements and restart UNIX System Services.

T01OE106E TCP/IP PFS Name:*pfs_name* Subtask:*tcb_addr* Terminated

Reason: The UNIX System Services PFS named *pfs_name* subtask located at address *tcb_address* in the UNIX System Services address space, terminated unexpectedly.

Action: Locate any error's prior error message that may explain why the subtask prematurely terminated. The subtask is automatically restarted.

T01OE107I TCP/IP PFS Name:*pfs_name* Terminating

Reason: The UNIX System Services PFS named *pfs_name* has started termination.

Action: None. This is an Informational message.

T01OE108E TCP/IP PFS Name:*pfs_name* Load Failed for *mod_name*, Code = *abend_code* Reason = *rsn_code*

Reason: The UNIX System Services PFS named *pfs_name* attempted to load module *mod_name* into storage but failed with an ABEND code of *abend_code* and a reason code of *rsn_code*.

Action: Determine why the LOAD failed by using code as a reference. The most probable causes are that the specified module is not in a data set in the STEPLIB concatenation, insufficient storage exists in the region or the load data set has gone into secondary extents after UNIX System Services started.

T01OE109I TCP/IP PFS Name:*pfs_name* NETWORK STMT: TYPE(*type*) DOMAINNAME(*domain_name*) DOMAINNUMBER(*domain_number*) MAXSOCKETS(*sockets*)

Reason: The UNIX System Services PFS named *pfs_name* processed the NETWORK statement with a TYPE of type, DOMAINNAME of *domain_name*, DOMAINNUMBER of *domain_number* and MAXSOCKETS of *sockets*.

Action: None. This is an Informational message.

T01OE110W TCP/IP PFS Name:*pfs_name* Sysid:*subsys_name* Transport Provider Not Operational

Reason: The UNIX System Services PFS named *pfs_name* determined the product address space with a subsystem identifier of *subsys_name* does not have an operational UNIX System Services PFS Transport Provider active.

Action: Verify that the address space is executing on the same processor with a subsystem identifier of *subsys_name.* If the product is not active, start it. If the product is active, verify that the TCP task group is active and if not, restart it.

T01OE111E TCP/IP PFS Name:*pfs_name* Sysid:*subsys_name* Domain:*domain_number* Not Supported

Reason: The UNIX System Services PFS named *pfs_name* was configured to support the Domain *domain_number.* This domain is not supported by the PFS.

Action: Correct the domain number and restart UNIX System Services. If the condition continues, contact Customer Support.

T01OE112E TCP/IP PFS Name:*pfs_name* Unable To Obtain Storage For Inodes, Value = *ret_code*

Reason: The UNIX System Services PFS named *pfs_name* was attempting to allocate the inode pool via a Storage Obtain system call. The call failed with a return code of *ret_code*.

Action: Use the return code from the Storage Obtain call to determine why the call failed. If the call failed for insufficient virtual storage in the region, either increase the region size or decrease the maximum number of sockets.

T01OE113W TCP/IP PFS Name:*pfs_name* Name/Token Pair: *token_name*/*token_value* Previously Defined

Reason: The UNIX System Services PFS named *pfs_name* attempted to create a Named Token with name *token_name*, but the call to IEANTCR returned with return code 4, indicating the Named Token already exists. *token_value* is the value of token as reported by the IEANTRT service. The specified token is deleted by calling IEANTDL and an attempt is made to recreate the Named Token by calling IEANTCR a second time.

Action: The probable cause of this message is the definition of multiple PFSs within UNIX System Services that have the same product subsystem as the TCP service provider. A single product address space can only service one PFS per MVS. Correct the PFS definitions to point to a separate product address space or remove the duplicate definitions.

T01OE114E TCP/IP PFS Name:*pfs_name function_name* Error: *error_code* for Name:*token_name*

Reason: The UNIX System Services PFS named *pfs_name* issued a call to the MVS Named Token Function *function_name* and the call returned an error code indicating a permanent error had occurred.

Action: Determine the cause of the error by locating error code *error_code* for service *function_name* in IBM manual *MVS/ESA SP V5 AUTHORIZED ASSEMBLER SERVICES REFERENCE ENF-ITT, GC28-1476-00*. Follow the recommended action for the error to correct the problem.

T01OE115I TCP/IP PFS Name:*pfs_name config_type* PARM:

Reason: The UNIX System Services PFS named *pfs_name* has started the listing of configuration parameter data. This message may be followed by message T01OE116I.

Action: None. This is an Informational message.

T01OE116I *config_type_parm*

Reason: This is a continuation message for T01OE115I and is repeated until all configuration parameter data is listed.

Action: None. This is an Informational message.

T01OE117E TCP/IP PFS Name:*pfs_name config_type* Invalid Syntax

Reason: The UNIX System Services PFS named *pfs_name* encountered invalid configuration parameter data.

Action: Correct the parameter data and restart UNIX System Services.

T01OE118E TCP/IP PFS Name:*pfs_name config_type* Keyword:*kywd* Invalid Or Missing

Reason: The UNIX System Services PFS named *pfs_name* determined that a configuration keyword was invalid or not specified.

Action: Correct the required keyword parameter and restart UNIX System Services.

T01OE119E TCP/IPPFS Name:*pfs_name config_type* Keyword:*kywd* Parm:*parm* Invalid Or Missing

Reason: The UNIX System Services PFS named *pfs_name* determined that a configuration keyword specified incorrect or omitted parameter data.

Action: Correct the keyword parameter data and restart UNIX System Services.

T01OE120E TCP/IP PFS Name:*pfs_name config_type* Keyword:*kywd* PARM:*parm* Invalid Syntax

Reason: The UNIX System Services PFS named *pfs_name* determined that a configuration keyword parameter data was specified incorrectly.

Action: Correct the keyword parameter data and restart UNIX System Services.

T01OE121E TCP/IP PFS Name:*pfs_name* Subtask *taskname* Subcommand *cmnd* ABEND:*abend_code*; STCQ=*addr* SCMD=*addr* VCMD=*addr*

Reason: The UNIX System Services PFS named *pfs_name* detected an ABEND *abend_code* having occurred in a subtask while processing the specified subcommand. The addresses of various control blocks are specified by *addr* to assist in problem determination.

Action: This condition indicates an internal error in T010PST. Contact Customer Support, unless the problem was caused by some operational type problem (for example, UNIX System Services canceled, insufficient virtual storage, and so on).

Chapter 28

T01PM*nnn* Port Mapper Messages

This section describes messages issued by the Port Mapper. These messages include T021M000 through T01PM999.

T01PM*nnn*

T01PM001I Portmap initialization successful.

Reason: The Portmapper was successfully initialized.

T01PM002E Portmap initialization failed - reason code = *rsn_code*

Reason: Initialization for the Port Mapper failed with reason code *rsn_code*

Action: Use the reason code to direct your action.

T01PM004E Portmap Unknown command in configuration member *mem_name*. Command *cmnd_name* Is Not Valid

Reason: The portmapper encountered an unknown command *cmnd_name* in configuration *mem_name*.

Action: Verify and correct the command name in the specified member.

T01PM005E Only one *cmnd_name* command allowed in configuration member *mem_name*

Reason: Only one command is allowed in the configuration member specified.

Action: Correct the entry in the configuration member and retry.

T01PM006E Required keyword *kywd* was not specified on the *stmnt* statement in configuration member *mem_name*

Reason: The required keyword was not given on the statement *stmnt* in the configuration member specified.

Action: Edit the configuration member to add the keyword.

T01PM007E Parsing of input parameters in configuration member *mem_name* failed, reason code = *rsn_code*

Reason: The input parameters in the specified configuration member caused a parsing error with reason code *rsn_code*.

Action: Use the reason code to direct your action.

T01PM008E Configuration member *mem_name* did not contain a MAP options statement

Reason: The specified configuration member did not contain a MAP options statement.

Action: Edit the configuration member to include a MAP options statement.

T01PM009E Configuration member *mem_name* does not exist. Initialization failed

Reason: Initialization failed because the specified configuration member was not found.

Action: Verify the location of the specified configuration member.

T01PM010E command *cmnd_name* failed in member *mem_name* on statement *stmnt*

Reason: The *cmnd_name* command failed in the statement specified in the configuration member.

Action: Correct the error and restart the MAP task group.

T01PM012E Return code *ret_code* received from configuration file I/O handler

Reason: The configuration file I/O handler returned *ret_code*.

T01PM013I Portmap shutdown successful

Reason: The portmapper shut down successfully.

Action: None. This is an Informational message.

T01PM14I Portmap completed execution - return code = *ret_code*

Reason: The portmapper completed its execution with *ret_code*.

Action: None. This is an Informational message.

T01PM015E Abend detected during port mapper processing; taskgroup terminating

Reason: The portmapper is terminating because an ABEND occurred.

T01PM016E Load failed for DNRSCALL; Portmap task group terminating

Reason: The portmapper is terminating because the load for DNRSCALL failed.

Chapter 29

T01S4*nnn* Spool#4 Messages

This chapter describes messages written by Spool4. These messages are T01S4000 through T01S4999.

T01S4*nnn*

T01S4004E RC=*rc*, SRC=*src*,DSN=*dsn*

Reason: Ptask SPOOL#4 may have terminated with an error. This message follows one of the messages T01S4005I through T01S4019I, T01S4021I, or T01S4022I. The message that precedes this message explains the usage of fields in this message. The fields used in this message depend on the content of the previous message.

Action: Use the information in the previous message to direct the action you take.

T01S4005I Spool#4 error: normal termination

Reason: The job is going down or a user requested that this task terminate.

T01S4006I Spool#4 error: Bad Mail configuration

Reason: Program Spool#4 cannot find the startup parameters. Program Spool#4 terminates.

Action: Check the APPCFG*xx* macro SMTP for a remote containing either a valid remote number or destination name.

T01S4007I Spool#4 error: PCore alloc failed

Reason: There is not enough region available to run this ptask

Action: At job start-up, raise the region parameter of the job. If the job has been running for a while this message is indicative of another task allocating all the storage but not releasing it. Save all output from the job. Check the JES and T01 log files for related messages. Contact Customer Support.

T01S4008I Spool#4 error: PCore free failed

Reason: The task could not release storage allocated.

Action: Check the JES and T01 log files for related messages. Contact Customer Support.

T01S4009I Spool#4 error: IEFSSREQ failed

Reason: A call to the subsystem interface failed. Program SPOOL#4 terminates. Message T01S4004E follows in which the following fields are used:

RC=*rc* IEFSSREQ register 15 return code:

- 0 Success, request went to a subsystem.
- 4 Subsystem does not support this function.
- 8 Subsystem exists but is not up.
- C Subsystem does not exist
- 10 Disastrous error.
- 14 Logical error (bad SSOB format, ...).

SRC=*src* SSOB return code:

- 0 Success.
- 4 No more data sets to select.
- 8 Job not found.
- C Invalid search arguments.
- 10 Unable to process now.
- 14 Duplicate jobnames.
- 18 Invalid jobname/jobid combination.
- 1C Invalid destination specified.

Action: Let the RC=rc and SRC=src direct your response. Save all output from the job. Check the JES and T01 log files for related messages. Contact Customer Support.

T01S4010I Spool#4 error: SSOBRETN error

Reason: A call to the subsystem interface failed. Message T01S4004E follows in which the following fields are used:

RC=*rc* IEFSSREQ register 15 return code:

- 0 Success, request went to a subsystem.
- 4 Subsystem does not support this function.
- 8 Subsystem exists but is not up.
- C Subsystem does not exist.
- 10 Disastrous error.
- 14 Logical error (bad SSOB format, ...).

SRC=*src* SSOB return code

- 0 Success.
- 4 No more data sets to select.
- 8 Job not found.
- C Invalid search arguments.
- 10 Unable to process now.
- 14 Duplicate jobnames.
- 18 Invalid jobname/jobid combination.
- 1C Invalid destination specified.

Action: Let the RC=rc and SRC=src direct your response. Save all output from the job. Check the JES and T01 log files for related messages. Contact Customer Support.

T01S4011I Spool#4 error: JES DSN allocation failed

Reason: An attempt to allocate a JES data set failed. Message T01S4004E follows in which the following fields are used:

RC=*rc* PYNDAL error code.

SRC=*src* SVC 99 error code

DSN=*dsn* Name of the JES data set.

This message can indicate using SPOOL#3 JES remote definitions with program SPOOL#4.

Action: Your product support systems programmer should check that correct JES definitions for program SPOOL#4 are used. Save all output

T01S4012I Spool#4 error: Mail DSN alloc failed

Reason: An attempt to allocate a MAIL data set failed. Message T01S4004E follows in which the following fields are used:

RC=*rc* PDYNAL error code.

SRC=*src* SVC 99 error code.

DSN=*dsn* Name of the JES data set.

Action: Save all output from the job. Check the JES and T01 log files for related messages. Contact Customer Support.

T01S4013I Spool#4 error: JES DSN open failed

Reason: An attempt to open a JES data set failed. The data set is deleted. Message T01S4004E follows in which the following fields are used:

RC=*rc* POPEN error code.

DSN=*dsn* Name of the JES data set.

Action: Save all output from the job. Check the JES and T01 log files for related messages. Contact Customer Support.

T01S4014I Spool#4 error: Mail DSN open failed

Reason: An attempt to open a MAIL data set failed. Message T01S4004E follows in which the following fields are used:

RC=*rc* POPEN error code.

DSN=*dsn* Name of the MAIL data set.

Action: Save all output from the job. Check the JES and T01 log files for related messages. Contact Customer Support.

T01S4015I Spool#4 error: JES DSN close failed

Reason: An attempt to close a JES data set failed. The data set is deleted. Message T01S4004E follows in which the following fields are used:

RC=*rc* PCLOSE error code.

DSN=*dsn* Name of the JES data set.

Action: Save all output from the job. Check the JES and T01 log files for related messages. Contact Customer Support.

T01S4016I Spool#4 error: Mail DSN close failed

Reason: An attempt to close a MAIL data set failed. Message T01S4004E follows in which the following fields are used:

RC=*rc* PCLOSE error code.

DSN=*dsn* Name of the MAIL data set.

Action: Save all output from the job. Check the JES and T01 log files for related messages. Contact Customer Support.

T01S4017I Spool#4 error: JES DSN free failed

Reason: An attempt to free a JES data set failed. Message T01S4004E follows in which the following fields are used:

RC=*rc* PDYNAL error code.

SRC=*src* SVC 99 error code.

DSN=*dsn* Name of the JES data set.

Action: Save all output from the job. Check the JES and T01 log files for related messages. Contact Customer Support.

T01S4018I Spool#4 error: Mail DSN free failed

Reason: An attempt to free a MAIL data set failed. Message T01S4004E follows, in which the following fields are used:

RC=*rc* PDYNAL error code.

SRC=*src* SVC 99 error code.

DSN=*dsn* Name of the MAIL data set.

Action: Save all output from the job. Check the JES and T01 log files for related messages. Contact Customer Support.

T01S4019I Spool#4 error: Spool#4 abended

Reason: Ptask SPOOL#4 ABENDed. Message T01S4004E follows, in which the following fields are used:

RC=*rc* ABEND code.

SRC=*src* ABEND code.

PSW=*psw* PSW at the time of the ABEND.

SPOOL#4:*address* Starting address of CSECT SPOOL#4.

Action: Save all output from the job. Check the JES and T01 log files for related messages. Contact Customer Support.

T01S4021I Spool#4 error: Bad JES DSN LRECL

Reason: A JES data set had an LRECL greater than 255. The data set is deleted. Message T01S4004E follows, in which the following fields are used:

RC=*rc* LRECL.

DSN=*dsn* Data set name that was deleted.

Action: Verify that the programs writing JES data sets to the remote or destination is not writing data sets with an LRECL greater than 255.

T01S4022I Spool#4 Error: Bad JES record length

Reason:

Program SPOOL#4 reads files off the JES spool destination defined on the REMOTE parameter in the SMTP statement in APPCFG*xx*. While attempting to read a record, it discovered that the record length returned was larger than the LRECL of the JES data set. This is a permanent error preventing the full delivery of the mail message.

SPOOL#4 issues this message and related message T01S4004E. When T01S4004E accompanies this message, the parameter descriptions are as follows:

RC	Length of the record.
SRC	JES data set LRECL.
DSN	JES data set name.

SPOOL#4 places error message T01S4022I into the truncated mail output data set in an attempt to inform the user that there is a problem with the delivery of this message.

Action:

Look for defects in the application that places data sets on the JES spool. Specifically, look for instances where it could write blocks of data in which a record may be larger than the LRECL of the data set.

Chapter

30 T01SF*nnn* Server FTP Messages

This section describes messages issued by Server FTP. These messages include T01SF000 through T01SF999.

T01SF*nnn*

T01SF001I USER=*userid time* HOST=*host_name*

Reason: This is an informative message indicating that a user signed on. This message is written to the log during FTP server control connection termination.

userid	User name.
time	Time at which the user signed on.
host_name	Host from which the user signed on.

Action: None. This is an Informational message.

T01SF002E PCORE failed for server FTP workarea

Reason: Server FTP attempted to acquire storage through the pcore facility, but the request failed.

Action: This error could be temporary, or indicative of other problems within the product. If the problem persists, check the JES log for dumps. The T01LOG and JES log of the job should be saved. Contact Customer Support.

T01SF003E Server FTP ATopen failed, code = *code*

Reason: An attempt by the FTP server to open an FTP connection failed. If the code is 8, this message may be preceded by message T01TC010I. See message T01ST011E for an Reason of the return codes.

Action: Verify that the remote host is functioning properly.

T01SF005E Host *host_number*, timeout on Telnet connection

Reason: The FTP server attempted to send an FTP reply to the host at *host_number*. There was no response and the reply timed out.

Action: Check the log for TCP/IP-layer messages that might indicate a problem with the connection.

T01SF006E Server FTP configuration definition statement missing; port=*port_num*.

Reason: A parameter list SERVICE statement defining a well-known port (*port_num*) for Server FTP was established, but no FTP configuration statement for this port was found.

Action: Verify the configuration statements within the APPCFG*xx* parameter member. Remove the SERVICE statement or define an FTP statement for the preceding port.

T01SF007I *nnn text*

Reason: These messages are written to the Accounting Log for FTP server events such as login or data transfer completion. Each message (a message may be multiple lines) is prefixed with an FTP server reply code *nnn*. See "FTP Server Messages" in *Unprefixed Messages* for the reason for the reply code and text.

Action: None. This is an Informational message.

Chapter

31 T01SM*nnn* SMTP Messages

This chapter describes messages issued by the SMTP Mail Server. These messages include T01SM001 through T01SM999.

T01SM*nnn*

T01SM001I I/O Error: *err_text*

Reason: A call to the ungetc subroutine is passed a bad or null value in the iobp (or stream). As a result, there is no place to push the character on the stream. The message is generated by SNDMSG, SSMTP, USMTP, or WHOIS. The message is written to the error log.

err_text will be one of the following:

unget: undefined unit	A null value was passed in the iobp (or stream).
can't ungetc	A bad value was passed in the iobp (or stream).

Action: None. This is an Informational message.

T01SM002I MENU ERROR (*err_text*): *string*

Reason: Module _cnvcord parses the format control string and discovers an error. *string* is the format control string menu field. The message is written to the error log.

err_text will be one of the following:

form ?	The f conversion field does not equal any legitimate value (C, S, U, D, or B).
len>4	The length field (*nn*) is too large.
no len	There is no length field.
no %	There is no % in the format control string menu field number.

T01SM003E PARM ERROR (*err_text*): *string*

Reason: A call to the module _argscan was made incorrectly. The message is written to the error log.

err_text will be one of the following:

no -	The call was made using the Multics/UNIX style (hyphen letter). Not all arguments of the format control string menu field are of the Multics/UNIX style. The bad format control field string is printed. *string* is the bad format control string menu field.
????	The call made with a bad pointer in the argv array. *string* is the string pointed to by the argv value.
length	The call was made with a bad parameter string (*string*) length, as determined by (*nn*) in the format control string field. *string* is the input string.

Action: Save all output from the job. Notify Customer Support.

T01SM006W Local Subsystem ID: *ssid* Lookup error is: *ac.ec text*, Diag=x'*code*'

Reason: A GETHOSTBYNAME call to DNR for the specified subsystem ID failed. See DNR Return Codes in *Unprefixed Messages* for more information on action, error, and diagnostic codes.

ssid	The subsystem ID.
ac	The action code.
ec	The error code.
text	The message text.
code	The diagnostic code.

Action: The error message and codes should direct your actions. One possible reason is that the local subsystem ID is not defined in the DNR alias member.

This message can be due to a temporary resource initialization coordination problem or it can be due to a bad DNR configuration. Multiple occurrences of this message may be an indicator of configuration or network problems.

The TCP base product must be able to resolve the subsystem name into a host name. You can use the TSO command DNRGET to verify whether the TCP base product can correctly map the subsystem name into a host name.

An MVS host IBMHOST.COLUMBIA.COM using the subsystem name ACSS with an IP address of 138.42.32.24 can be verified with the following TSO command:

```
DNRGET H BYN ACSS SYSID(ACSS)
```

If the response from the DNRGET command does not indicate a correct host name and IP address combination, check the installation's DNR definitions in the configuration files DNRALC*xx* and DNRHST*xx*.

These are the DNRALC*xx* file entries for the local host IBMHOST.COLUMBIA.COM:

```
ACSS            IBMHOST.COLUMBIA.COM
IBMHOST     IBMHOST.COLUMBIA.COM
```

These are the DNHRST*xx* file entries for the local host IBMHOST.COLUMBIA.COM:

```
IBMHOST.COLUMBIA.COM            138.42.32.24
```

T01SM007E Local Subsystem ID: *ssid* Lookup error return code: *rc*

Reason:

A GETHOSTBYNAME call to DNR for the specified subsystem ID failed. Module GETHBDNR could not accommodate the request and returned the specified error code.

ssid The subsystem ID

rc The error code:

- 2 Bad arguments were passed to GETHBDNR.
- 4 No core was available for the request.
- 5 A software error occurred when processing the DIRSRV call in GETHBDNR.

Action:

This error could be temporary or indicative of other problems within the product. If the problem persists, the ACSNAP log should be checked for dumps. The WTP and JES log of the job should be saved. Contact Customer Support. This message can be due to a temporary resource initialization coordination problem or it can be due to a bad DNR configuration. Multiple occurrences of this message may indicate configuration or network problems.

The TCP base product must be able to resolve the subsystem name into a host name. Use the TSO command DNRGET to verify whether the TCP base product can correctly map the subsystem name into a host name.

To verify the MVS host IBMHOST.COLUMBIA.COM, using the subsystem name ACSS with an IP address of 138.42.32.24, use this TSO command:

```
DNRGET H BYN ACSS SYSID(ACSS)
```

If the response from the DNRGET command does not indicate a correct host name and IP address combination, check the installation's DNR definitions in configuration files DNRCALC*xx* and DNRHST*xx*.

These are the DNRALC*xx* file entries for the local host IBMHOST.COLUMBIA.COM:

```
ACSS            IBMHOST.COLUMBIA.COM
IBMHOST     IBMHOST.COLUMBIA.COM
```

These are the DNRHST*xx* file entries for the local host IBMHOST.COLUMBIA.COM:

```
IBMHOST.COLUMBIA.COM            138.42.32.24
```

T01SM008I Bad local domain *dom_name*

Reason: The inbound mailer ptask SSMTP discovered that its *subsystemid* maps to a name that contains invalid syntax. Ptask SSMTP issues this message and exits with a return code of 1024.

The cause of the problem could be one of the following:

- The DNRCCC*xx* configuration files are misconfigured. There could be startup syntax errors in the DNRLOG.
- The DNRALC*xx* and DNRHST*xx* configuration files do not map the subsystem name for TCP to its fully qualified name.
- The name server processing this product's requests does not map the product name correctly.

Action: Take the following actions:

- Check your DNRCCC*xx* configuration file for errors.
- Examine the DNRLOG for errors.
- Check that the TCP subsystem id in the DNRALC*xx* file maps to the fully qualified host name in file DNRHST*xx*.
- When running DNR in global mode, verify that the *nameservers* know your TCP MVS host name.

- Issue the following command under TSO while this product is running (where LINK library routine DNRGET is available via STEPLIB or LNKLST*xx*:

  ```
  DNRGET HOST BYNAME ssss SY(ssss)
  ```

  Where *ssss* is the subsystem ID for this product.

  If the DNRGET does not return the fully qualified host name for your MVS host and its IP address, then your site's DNR configuration or its name servers configuration should be examined in detail.

T01SM009I Bad foreign domain *dom_name*

Reason: There is an invalid SSMTP FROM= parameter. The FROM= parameter is to override the domain name (*dom_name*) on the HELO command.

Action: The FROM= parameter is part of a testing facility within the product for use by Customer Support only. This message should never appear in the log. The *dom_name* is the test domain that should replace domain name on an HELO command. Contact Customer Support.

T01SM010I *lhost* SMTP Receiver starting for host *rhost*

Reason: The SSMTP receiver is starting between the local host and remote host.

Action: None. This is an Informational message.

T01SM011I Write error to foreign host: *reason*

Reason: SSMTP has trouble communicating with a foreign host. The reason parameter is the reason why it failed. The message is written to the error log. See "perror Messages" in *Unprefixed Messages and Codes* for more information.

Action: The *reason* should direct your actions.

T01SM012W SYSOUT alloc failed: *reason*

Reason: SSMTP tries to allocate a mail spool file and fails. The reason why it failed is given (*reason*). See "perror Messages" in U*nprefixed Messages and Codes* for more information.

Action: The *reason* should direct your actions.

T01SM013I Bad HELO command – no domain given

Reason: The HELO command sent to the inbound mail handler did not contain a domain. All mail HELO commands should contain a valid principal host domain name. This is a clear violation of the RFC 1123 MAIL specification. This message is a warning; mail processing for the session continues.

Action: The system administrator should examine the T01LOG to see if it can be determined which mailer is in violation of the MAIL protocol specifications. The system administrator should then report the problem to Customer Support.

T01SM014I HELO command domain truncated

Reason: The HELO command sent to the inbound mail handler contained a domain name that was greater than 64 characters. The mail handler uses only the first 64 characters of the domain name for its processing. This message is a warning; mail processing for the session continues.

Action: None. This is an Informational message.

T01SM015I VALIDATION FAILED FOR CHAR/DBCS TRANSLATION MODULE *mod*

Reason: An invalid character set translation table (*mod*) was specified or defined. This table may have been created incorrectly or was not created for the specified transopt (single or double).

Action: Verify character set matches the translation mode.

T01SM016E ACLOAD FAILED - MODULE *module*

Reason: This message appears immediately before an ABEND issued by the ACLOAD routine in ACCOM when a PLOAD macro specifies a module (*module*) not usable for subsequent processing. Unless local modifications are made to the product's distribution libraries, this message should only appear during testing, not in a product environment.

The code of the subsequent ABEND identifies the reason:

706 The module is linked as nonexecutable.

C06 The module is not found in the correct library, or is linked to be either overlay or scatter.

D06 An error occurred during the BLDL process.

Action: The product LOAD library must be in the STEPLIB DD statement of the startup JCL. If the ABEND code is a D06 and the module name is ARPAINIT, probably the LOAD library was not placed into (or was removed from) the STEPLIB DD statement in the startup JCL.

In most cases a relink with the correct attributes corrects this condition. The D06 ABEND is triggered by system action, and may require the assistance of a system programmer to resolve.

T01SM017I Can't open log/acct file

Reason: The LOG/ACCT file could not be opened.

Action: None. This is an Informational message.

T01SM018W file size received:*filesize* exceeds DIVERT limit: *divert_lim*

Reason: The received file size *filesize* was greater than the DIVERT limit *divert_lim*. The file is diverted as specified in the DIVERT parameter in APPCNFG.

T01SM019I Bad HELO command – invalid domain: *dom*

Reason: The HELO command sent to the inbound mail handler contained invalid characters in the domain name. All mail HELO commands should contain a valid principal host domain name. This message is a warning; mail processing for the session continues.

Action: The system administrator should examine the T01LOG to determine which mailer is in violation of the MAIL protocol specifications and call Customer Support.

T01SM020I CANNOT PLOAD SMTPUSR

Reason: Module SMTPUSR cannot be loaded from the STEPLIB DD libraries. This message is a warning; mail processing for the session continues.

Action: The system administrator should examine the STEPLIB DD libraries for the module. Regenerate SMTPUSR if missing.

T01SM021I RCPT TO User: *user* replaced with table entry *table*

Reason: SSMTP processed a RCPT TO mail command for inbound mail. Exit SMTPUSR replaced the specified user name with the specified table entry. Exit SMTPUSR lets a user name be mapped into a TSO user ID. This is an informational message to keep track of a RCPT TO user being changed by exit SMTPUSR.

Action: None. This is an Informational message.

T01SM022I MAIL RCVR: Discard partial message

Reason: The input mail file receives an unexpected EOF or an input error. All partially generated mail files are deleted.

Action: None. This is an Informational message.

T01SM023I ... RCVD: *time* HOST:*host m* lines *n* chars

Reason: SSMTP receives a mail message from the specified host.

time The time at which the mail message is diverted.

host The host from which the mail message is diverted.

m The length of the mail message in lines.

n The length of the mail message in characters.

Action: None. This is an Informational message.

T01SM025I MAIL COPY: to printer

Reason: SSMTP sends a mail message to the printer.

Action: None. This is an Informational message.

T01SM026I Write error on temporary file: *reason*

Reason: SSMTP has trouble writing a mail file on the JES spool. *reason* is the reason it failed. See "perror Messages" in U*nprefixed Messages and Codes* for more information.

Action: The reason should direct your actions.

T01SM027I MAIL DIVERTED: *time* HOST:*host m* lines *n* chars

Reason: SSMTP is diverting a mail message to the printer from a host. The APPCFG*xx* parameter that may cause mail to be diverted to a printer is "DIVERT=" in the SMTP statement.

time The time at which the mail message is diverted.

host The host from which the mail message is diverted.

m The length of the mail message in lines.

n The length of the mail message in characters.

Action: None. This is an Informational message.

T01SM030I Mail RCVR: session terminated prematurely

Reason: The Mail RCVR session ended before completing.

Action: None. This is an Informational message.

T01SM031I No host found in RCPT TO: *text*

Reason: The inbound mailer received a RCPT TO command in which the text (*text*) contains improper syntax. There was no @ in the text. It cannot be determined to which host the mail should be directed. The RCPT TO command is rejected.

Action: None. This is an Informational message.

T01SM032I RCPT TO command host *host* is not our host

Reason: SSMTP processed a RCPT TO mail command for inbound mail. The specified host address (*host*) on the RCPT TO mail command is different from this product's host domain. The RCPT TO command is accepted and this informational message is written into the log.

Action: The system administrator should decide whether or not another host is mistakenly sending mail to its system and react accordingly.

T01SM033I Alias host:*host* sent on RCPT TO command

Reason: SSMTP processed a RCPT TO mail command for inbound mail. The specified host (*host*) on the RCPT TO mail command is actually an alias name for the host. SSMTP accepts the RCPT TO command.

Action: The system administrator should decide whether or not the host is actually an alias as defined by the domain name resolver. If it is an alias, contact the postmaster of the sender of the mail message to change their host definition for your host. Sending of an alias in a RCPT TO mail command violates the SMTP protocol specifications.

T01SM034I Host: *host* Lookup error is:*ac.ec message*, Diag=x'*code*'

Reason: A GET HOST BY NAME call to DNR for the specified host failed. See "DNR Return Codes" in *Unprefixed Messages and Codes* for more information on action, error, and diagnostic codes.

host The host name.

ac The action code.

ec The error code.

message Message text for the action and error codes.

code The diagnostic code.

Action: The error message and codes should direct a user's action.

T01SM036I Host: *host* Lookup error return code: *rc*

Reason: A GET HOST BY NAME call to DNR for the specified host. Module GETHBDNR was unable to accommodate the request and returned the specified error code.

host The host for which the call failed.

rc The error code:

- -2 Bad arguments were passed to GETHBDNR.
- -4 No core was available for the request.
- -5 A software error occurred when processing the DIRSRV call in GETHBDNR.

Action: This error could be temporary or indicative of other problems within the product. If the problem persists, check for related SVC dumps that may have been issued. The T01LOG and JES log of the job should be saved. Contact Customer Support.

T01SM037I Host: *host* Lookup error is:*ac.ec* message, Diag=x'*code*'

Reason: A GET HOST BY NAME call to DNR for the specified host failed. See "DNR Return Codes" in *Unprefixed Messages and Codes* for more information on action, error, and diagnostic codes.

host	The host name.
ac	The action code.
ec	The error code.
message	Message text for the action and error codes.
code	The diagnostic code.

Action: The error message and codes should direct a user's action.

T01SM039I SSMTP HELO Validation failed for host: *host*

Reason: The HELO command sent to SSMTP contains a host (host) not in the host names table (ARPINAMS).

Action: Save all output from the job. Get the list of commands entered by the user that generated this error.

If the problem is an unknown or undefined host, check that the host is in the product's Host Names Table. Insert the needed host entry if necessary and reassemble the Host Names Table. Ignore this message if the user inadvertently entered a nonexistent host.

Retry the operation. If the problem persists, call Customer Support.

T01SM040I Mailer Daemon: *mail_errormsg*

Reason: USMTP cannot deliver mail because of the specified reason (*mail_errormsg*). The file is set aside but left on the system for examination or for another attempted transmission. The file name is printed in the accounting log. For more information about *mail_errormsg*, see the chapter "mail Error Messages" *Unprefixed Messages and Codes*. This message is followed by message T01SM042I.

Action: None. This is an Informational message.

T01SM041I Unknown domain literal: *domain* sent on RCPT TO command

Reason: SSMTP processed a RCPT TO mail command for inbound mail. The host address on the RCPT TO mail command was a domain literal. The specified domain literal (*domain*) is unknown to the product's domain name system. SSMTP accepts the RCPT TO command.

Action: The system administrator should decide whether or not to add a host definition for domain into the product's domain name system.

T01SM042I Mailer Daemon: File set aside: *dataset*

Reason: USMTP cannot send the specified data set (*dataset*) across the TCP connection. The set is left on the system for manual inspection.

Action: None. This is an Informational message.

T01SM043I Unknown host: *host* sent on RCPT TO command

Reason: SSMTP processed a RCPT TO mail command for inbound mail. The specified host (host) on the RCPT TO mail command is unknown to the product's domain name system. SSMTP accepts the RCPT TO command.

Action: The system administrator should decide whether or not to add a host definition for host into the product's domain name system.

T01SM044I Mailer Daemon: FILE OPEN FAILED *file reason*

Reason: USMTP fails to open a mail file. See "perror Messages" in *Unprefixed Messages and Codes* for more information.

file The mail file.

reason The reason for failing.

Action: The perror message should suggest the proper course of action.

T01SM045I Mailer Daemon: Sent OK; delete file *file*

Reason: USMTP is unsuccessful in sending the specified mail file (*file*).

T01SM046I Mailer Daemon: Error reason

Reason: USMTP is unable to send a file across the TCP connection or the mail data set is in error. The mail file is on disk for a later attempt at transmission. See "perror Messages" in *Unprefixed Messages and Codes* for more information.

Action: The reason (*reason*) directs the appropriate course of action.

T01SM047I Mailer Daemon: File reopen failed *file reason*

Reason: USMTP fails to open the specified mail file for the second time. See "perror Messages" in *Unprefixed Messages and Codes* for more detailed information.

Action: The perror message should suggest the proper course of action.

T01SM048I Mailer Daemon: REDELIVERY FAILED (*file*) -- *reason*

Reason: USMTP fails to deliver the specified mail file for the second time. The message is written to the error log. See "perror Messages" in *Unprefixed Messages and Codes* for detailed information.

Action: The reason (*reason*) directs the appropriate course of action.

T01SM049I Mailer Daemon: Send file=*dataset*

Reason: USMTP tries to use SMTP services to the specified mail data set (*dataset*).

Action: None. This is an Informational message.

T01SM050I DNR indicates that Host: *host* has no mail support

Reason: A GET HOST SERVICE BY NAME call to DNR for the specified mail host (*host*) returned. The results indicate that the host does not support an inbound TCP mail service at port 25. This host is dropped as a potential source to send this mail message.

Action: If the host does indeed support inbound mail at TCP port 25 then the server for the network must be updated to reflect this information.

T01SM051I Host: *host* failed get service by name due to: *ac.ec text*, Diag=x'*code*'

Reason: A GET HOST SERVICE BY NAME call to DNR for the specified host. An attempt is made to send the current mail message to the host anyway.

See "DNR Return Codes" in *Unprefixed Messages and Codes* for more information on action, error, and diagnostic codes.

host The host name.

ac The action code.

ec The error code.

message Message text.

code The diagnostic code.

Action: The error message and codes should direct a user's action.

T01SM052I Host: *host* get service by return code: *rc*

Reason: A GET HOST SERVICE BY NAME call to DNR for the specified host failed. Module GETHBDNR was unable to accommodate the request. An attempt is made to send the current mail message to the host anyway.

host The host for which the call failed.

rc The error code:

- -2 Bad arguments were passed to GETHBDNR
- -4 No core was available for the request
- -5 A software error occurred when processing the DIRSRV call in GETHBDNR.

Action: This error could be temporary or indicative of other problems within the product. If the problem persists, check for related SVC dumps that may have been issued. Save the T01LOG and JES log of the job. Call Customer Support.

T01SM056I MX record *pref host* dropped for route *route*

Reason: The specified host could not be used with the specified preference to deliver mail sent to the specified route. This MX record is dropped for this route. The host could be down or inaccessible, the server could indicate that the host does not support mail, and so on. This is an informational message.

pref The preference.

host The host name.

route The route.

Action: None. This is an Informational message.

T01SM057I Route: *route* failed get route by name due to: *ac.ec text*, Diag=x'*code*'

Reason: A GET ROUTE BY NAME call to DNR for the specified route failed. See "DNR Return Codes" in *Unprefixed Messages and Codes* for more information about action, error, and diagnostic codes.

An attempt is made to send the current mail message to the route anyway.

route The route name.

ac The action code.

ec The error code.

message Message text.

code The diagnostic code.

Action: The error message and codes should direct a user's action.

T01SM058I Route: *route* get route by name return code: *rc*

Reason: A GET ROUTE BY NAME call to DNR for the specified route failed. Module GETHBDNR was unable to accommodate the request. An attempt is made to send the current mail message to the host anyway.

route The host for which the call failed.

rc The error code:

- -2 Bad arguments were passed to GETHBDNR
- -4 No core was available for the request
- -5 A software error occurred when processing the DIRSRV call in GETHBDNR.

Action: This error could be temporary or indicative of other problems within the product. If the problem persists, check for related SVC dumps that may have been issued. Save the T01LOG and JES log of the job. Call Customer Support.

T01SM062I Alias host:*alias* in X-to replaced with *host_name*

Reason: USMTP decided that the host on the outbound X-to mail header is an alias. USMTP replaces the alias with the host name as defined by the installation in the RCPT TO SMTP command. The processing of the outbound mail continues.

Action: If the specified alias is an alias for the specified host, no action is required. Otherwise, the system administrator should examine the *host* and *alias* definitions to the product.

T01SM063I Mailer Daemon: file *dataset* Sent OK to host: *host/route*

Reason: The outbound mailer succeeded in sending a mail data set.

Action: None. This is an Informational message.

T01SM064I Mailer Daemon: Failed (Requeue File) *reason<--buff_msg*

Reason: USMTP tries to send a file across the TCP connection and fails. The mail file is still on disk for a later attempt at transmission. See "perror Messages" in *Unprefixed Messages and Codes* for more information.

Action: The reason directs the appropriate course of action.

T01SM065I MAILER Daemon: Failed to host: *host* (Requeue file via UMAPTEMP) *errmsg* <==*index*

Reason: Mailer daemon failed sending to host *host*. The file is requeued in accordance with the UMAPTEMP parameter of the SMTP configuration statement.

host	The HOST that failed.
errmsg	Detail of the error condition.
index	Error code. Equivalent to the error detail.

Action: None. This is an Informational message.

T01SM073E ERROR *host_name* too long: *hhhhhhhh*

Reason: The outbound mailer program USMTP made a call to DNR trying to resolve the subsystem name into a host name. The host name returned was longer than 255 characters. Program USMTP exits with the x'400' ABEND for bad parameters. *hhhhhhhh* was the host name returned.

Action: Check your DNR configuration or domain name server configuration for your installation's MVS host name and confirm that it is 255 characters or less.

T01SM074I Mailer Daemon: Starting for *host* (alias *alias*) *time*

Reason: USMTP ptask starts on the specified host with the specified alias at the specified time.

Action: None. This is an Informational message.

T01SM075I Mailer Daemon: SMTP Sender Daemon *date* Directory: *hlq*

Reason: USMTP's ptask is running.

date	The SMTP version date.
hlq	The high-level qualifier for mail data sets that USMTP attempts to mail.

Action: None. This is an Informational message.

T01SM076I Mail Host: *host* was not resolved by DNR, Lookup error is: *ac.ec text*, Diag=x'*code*'

Reason: A GET HOST BY NAME call to DNR for a host failed. The mail host name is made of the host and domain parameters on the host statement in the DNRALCx*x* member. DNR could not resolve the current host name. DNR could be misconfigured or not ready to receive the DIRSRV call from the APP task group.

See "DNR Return Codes" in *Unprefixed Messages and Codes* for more information on action, error, and diagnostic codes.

host The host name.

ac The action code.

ec The error code.

message Message text.

code The diagnostic code.

Action: The error message and codes should direct a user's action. If outbound mail is being sent, then ignore this message.

T01SM077I Mail Host: *host* was not resolved by DNR, Lookup error rc: *rc*

Reason: A GET HOST BY NAME call to DNR for the specified mail host failed. Module GETHBDNR was unable to accommodate the request and returned the specified error code.

host The host for which the call failed.

rc The error code:

- -2 Bad arguments were passed to GETHBDNR
- -4 No core was available for the request
- -5 A software error occurred when processing the DIRSRV call in GETHBDNR.

Action: This error could be temporary or indicative of other problems within the product. If the problem, persists the log should be checked for dumps. The T01LOG and JES log of the job should be saved. Call Customer Support.

T01SM078I Current USMTP cycle is: cccccccc

Reason: The current outbound mail cycle number is cccccccc. This information message only comes out when the TRACE parameter has been placed on the SMTP statement in the file APPCFG*xx*.

Action: None. This is an Informational message.

T01SM080I Cleaning database of *dataset*, data set no longer exists

Reason: An attempt is made to allocate and read the specified data set (*dataset*). The DAIR dynamic allocation messages indicate that the data set does not exist. The product mail database has this file entry removed.

Action: None. This is an Informational message.

T01SM081I Mailer Daemon: PCORE GET FAILED-Work Area

Reason: Insufficient storage is available for the USMTP ptask parser work area. The pcore macro fails. At this time, USMTP gives up trying to send this message. It tries again later when more storage is available.

Action: Save all output from the job. Check the WTO and JOB logs for related messages. Check for related SVC dumps that may have been issued. Use the SNAP command to spin off a formatted snap dump of storage for the task group. If there is a lot of the same kind of storage, a ptask may be allocating storage in a loop. Retry the operation first on the current version of this product. Retry the operation again. If it fails again, bring the product down and back up again. Raise the region size on the job. If the error recurs, call Customer Support.

T01SM082I Examining file: *file* for proper format

Reason: Outbound mail client USMTP reads file (*file*) to see if the internal format of the X-to and X-from headers are valid.

Action: None. This is an Informational message.

T01SM083I file: *ffffffff*, Next cycle=*nnnnnnnn*, Current cycle is cccccccc

Reason: Mail file *ffffffff* could not be delivered to all hosts. The current outbound mail cycle number is cccccccc. A re-attempt to send mail file *ffffffff* will be made during cycle *nnnnnnnn*. This information message only comes out when the TRACE parameter has been placed on the SMTP statement in file APPCFG*xx*.

Action: None. This is an Informational message.

T01SM084I Mail Daemon: New file *file*

Reason: USMTP successfully tries to read the specified mail file (*file*).

Action: None. This is an Informational message.

T01SM085I Mailer Daemon: PCORE GET FAILED-Work Area

Reason: Insufficient storage is available for the USMTP ptask work area. The pcore macro fails. At this time, USMTP gives up trying to send this message. It tries again later when more storage is available.

Action: Save all output from the job. Check the WTO and JOB logs for related messages and for related SVC dumps that may have been issued. Use the SNAP command to spin off a formatted snap dump of storage for the task group. If there is a lot of the same kind of storage, a ptask may be allocating storage in a loop. Retry the operation first on the current version of this product. Retry the operation again. If it fails again, bring the product down and back up again. Raise the region size on the job. If the error recurs, call Customer Support.

T01SM087I File: *dataset* has unsupported DCB *attribute=type/value*

Reason: DCB attributes of an outbound mail data set were examined, and a DCB attribute was found to be of an unsupported type or value. Client mail program USMTP ignores the file and it is not sent.

dataset	The outbound mail data set.
attribute	The DCB attribute (either LRECL, BLOCKSIZE, RECFM, or DSORG).
type/value	The unsupported type or value.

Action: Examine the process that created the specified data set. Correct the process so it creates files with supported DCB attributes.

T01SM088I Mailer Daemon: dsn *dataset* is marked in error, file ignored

Reason: The outbound mailer read the mail database and the specified data set (*dataset*) was marked non-sendable and non-returnable by a previous mail attempt to send the file.

Action: Examine the data set for bad formats. Previous mail T01LOG contains messages of the mail transfer failure. The system programmer should delete the record for this mail data set.

T01SM089I DCB for dsn=*dataset* recfm=*r* lrecl=*l* blksize=*b* dsorg=*p*

Reason: This is an informational message that lists the DCB attributes of the specified outbound mail data set.

dataset The outbound mail data set.

- r The record format.
- l The logical record length.
- b The blocksize.
- p The data set organization.

Action: None. This is an Informational message.

T01SM096I File: *dataset* set aside, may have ABENDed previously

Reason: When client mail program USMTP was reading the mail database, it found a record indicating that the specified data set (*dataset*) may have previously been involved with an ABENDing USMTP. Client mail program USMTP ignores the file; it is not sent.

Action: Examine the data set. Examine the current and previous runs of this product. Look for any related SVC dumps that may have been issued. Call Customer Support if an ABEND with USMTP can be found. Optionally, delete the mail database (not the data set) while the product is not running. Bring the product up; see if mail client program ABENDs and produces an SVC dump.

T01SM097E Unable to write the mail database: *mmmmmmmm*

Reason: The outbound mailer program USMTP failed to write to the mail database *mmmmmmmm*. All previous cycle and count information for all mail data sets in the database is reset to zero.

Action: Examine T01LOG (the ERROR log) for related failure messages. Let those messages guide your action. Most times another address space has the mail database allocated and needs to free the database.

T01SM098I Mailer Daemon: -X SYSOUT FAIL (*reason*)

Reason: USMTP tries to utilize the XJES USMTP parameter for testing. The XJES parameter allows parsing of the headers but bypasses the SMTP function and sends the message output to the printer. The SYSOUT data set failed allocation due to the reason given (*reason*). See "perror Messages" in *Unprefixed Messages and Codes* for more information.

Action: The reason determines user response.

T01SM099I Mailer Daemon: Cannot open conn (rc=*rc*) – *reason* Host=*host address*

Reason: USMTP tries to open a TCP connection to the specified host. See "perror Messages" in *Unprefixed Messages and Codes* for a detailed description.

rc	The return code.
reason	The Reason of the return code.
host	The number of the host to which the TCP connection fails.
address	The IP address.

Action: The reason directs the appropriate course of action.

T01SM103I Mailer Daemon: (user *user_name*) (to host *host_name*) *msg_text*

Reason: A user was rejected by the specified host.

Action: The message text should direct a user's response.

T01SM106I MAIL SENT: *time* Host *host* by route *route m* lines *n* chars

Reason: USMTP sends a mail message.

time	The time at which the message was sent.
host	The destination host.
route	The route by which the message was sent.
m	The length of the message in lines.
n	The length of the message in characters.

Action: None. This is an Informational message.

T01SM107I Mailer Daemon: Send failed (rc=*code*) to Host *host n* by route *route*

Reason: USMTP tries to send mail to the specified host. The open fails.

code	The return code with which the open failed.
host	The destination host.
n	The host number of the destination host.
route	The route by which the message was sent.

Action: Save all output from the job. Check the WTO and JOB logs for related messages. Check for any related SVC dumps that may have been issued. Get the list of commands entered by the user that generated this error.

Start a TCPEEP trace using the TELNET, DEBUG, and SYSOUT options. Retry the operation. If the error recurs, try to determine from the trace which host software is misbehaving. Call Customer Support.

T01SM110I ... TO: *to_uid to_host* FROM: *from_uid@from_host*

Reason: USMTP sends mail from a user ID on the local host to a user ID on a remote host.

to_uid	The user ID to which the mail was sent.
to_host	The destination host.
from_uid	The user ID on the local host from which the mail was sent.
from_host	The local host from which the mail was sent.

Action: None. This is an Informational message.

T01SM111I UNDELIVERABLE To: *to_host* From: *from_host*

Reason: The outbound mailer failed in sending mail to a specified destination host from another host. Other messages in the T01LOG describe the nature of this failure.

Action: None. This is an Informational message.

T01SM112E USMTP Mailer Daemon: File Read Error *reason*

Reason: USMTP tries to copy mail text to the network and fails; reason tells why the file cannot be read. See "perror Messages" in *Unprefixed Messages and Codes* for more information.

Action: The reason should direct your actions.

T01SM113I Mailer Daemon: SYSOUT alloc fail: *reason*

Reason: USMTP cannot deliver a message and tries to allocate another SYSOUT file for error purposes. The allocation of this new mail data set fails.

The SYSOUT data set failed allocation due to the reason given (*reason*). The mail message is redirected. See "perror Messages" in *Unprefixed Messages and Codes* for more information.

Action: The reason should direct your actions.

T01SM114I Mailer Daemon: Send failed, redeliver *reason*

Reason: USMTP cannot deliver mail because of the reason given (*reason*). The mail message is redirected.

Action: See "perror Messages" in *Unprefixed Messages and Codes* and take the action specified there

T01SM116I ... Redelivery OK to: (*user*) *tu_id m* lines *n* chars

Reason: USMTP redelivers a mail message. Mail redelivery is governed by the RETURN= parameter on the SMTP statement in APPCFG*xx*.

user	The user to which the mail is directed.
tu_id	The TSO ID of the user.
m	The length of the mail message in lines.
n	The length of the mail message in characters.

Action: None. This is an Informational message.

T01SM117I Mailer Daemon:Notify (*user*) *userid* of failure to host=*host*, reason=*reason*

Reason: Mailer Daemon failed in communication with host *host* for reason *reason*. Notify User *user/userid* of this failure.

user	The user to be notified.
userid	The userid of the user to be notified.
host	The host that failed.
reason	The reason the host failed.

Action: None. This is an Informational message

T01SM120I Mailer Daemon: Detected previous failure. Forcing: *reason*

Reason: The FLIST parameter tells the outbound mailer how many entries to reserve for a table of IP hosts that had temporary failures during a USMTP send cycle. Once an outbound mail attempt fails with a non-permanent error to a host during a send cycle, not all subsequent mail requests to the host are attempted and this message displays with the original reason for the temporary failure.

The FLIST feature prevents a bad mail host from backing up the outbound mail queue with a number of temporary errors.

Action: None. This is an Informational message.

T01SM121I NULL X-from: <> in outbound mail message

Reason: The X-from header in the mail file contained neither a user nor host address. It was of the format: X-from: <> This is an informational message.

Action: None. This is an Informational message.

T01SM124I X-FROM User: *user* replaced with table entry *table*

Reason: USMTP processed an X-from mail header for outbound mail. Exit SMTPUSR replaced the specified user name with the specified table entry. Exit SMTPUSR lets a TSO user ID be mapped into a name. This is an informational message to keep track of a X-from user being changed by exit SMTPUSR.

Action: None. This is an Informational message.

T01SM125I Alias host:*alias* in X-from replaced with *host_name*

Reason: USMTP decided that the host on the outbound X-from mail header is an alias. USMTP replaces the alias with the host name as defined by the installation in the MAIL FROM SMTP command. The processing of the outbound mail continues.

Action: None if *alias* is an alias for host *host*. If the specified alias is not supposed to be an alias for the host name, the system administrator should examine the host and alias definitions to this product.

T01SM126I SEND: Field = *field*

Reason: The USMTP trace facility has been turned on in APPCFG*xx*. The X-from and X-to headers are being parsed and printed in the event log. These messages are for debugging purposes with mail. All the data found on the X-from and X-to lines are defined as Tokens and printed in subsequent messages. The field to parse is given (*field*).

Action: None. This is an Informational message.

T01SM127I Null x-header found

Reason: The X-from or X-to header in the mail file contained neither a user nor a host address.*n* The outbound mail request is rejected. Check the log for related messages for the data set name.

Action: Make sure the mail file data set was deleted. Check for message T01SM045I, which indicates it was deleted. Try to find what software generated this mail message with a null X-header and correct it. If the destination coded on the RETURN parm of the SMTP statement is LOCAL, a copy of the message can be found in one of the SYS*xxxxx* data sets of the started task in the class specified on the RETURN parm.

Action: None. This is an Informational message.

T01SM128I Bad character c xxxxxxxx

Reason: An X-from or X-to header in the mail file contains incorrect syntax. Character c is the start of the incorrect syntax on the X-header. *xxxxxxxx* is the X-header data. The outbound mail request is rejected. Check the log for related messages for the data set name.

Action: Make sure the mail file data set was deleted. Check for message T01SM045I, which indicates it was deleted. Try to find what software generated this mail message with bad syntax and correct it. If the destination coded on the RETURN parm of the SMTP statement is LOCAL, a copy of the message can be found in one of the SYS*xxxxx* data sets of the started task in the class specified on the RETURN parm.

T01SM129I Bad Literal string in x-header: xxxxxxx

Reason: An X-from or X-to header in the mail file contains incorrect syntax. A literal string does not terminate in the X-header. The X-header terminates before the ending double quote is found. *xxxxxxx* is the X-header data. The outbound mail request is rejected. Check the log for related messages for the data set name.

Action: Make sure the mail file data set was deleted. Check for message T01SM045I, which indicates it was deleted. Try to find what software generated this mail message with bad syntax and correct it. If the destination coded on the RETURN parm of the SMTP statement is LOCAL, a copy of the message can be found in one of the SYS*xxxxx* data sets of the started task in the class specified on the RETURN parm.

T01SM130I Bad Domain Literal in x-header: xxxxxxx

Reason: An X-from or X-to header in the mail file contains incorrect syntax. A domain literal contains incorrect syntax. *xxxxxxx* is the X-header data. The outbound mail request is rejected. Check the log for related messages for the data set name.

Action: Make sure the mail file data set was deleted. Check for message T01SM045I, which indicates it was deleted. Try to find what software generated this mail message with bad syntax and correct it. If the destination coded on the RETURN parm of the SMTP statement is LOCAL, a copy of the message can be found in one of the SYS*xxxxx* data sets of the started task in the class specified on the RETURN parm.

T01SM131I X-header enclosed incorrectly: xxxxxxx

Reason: An X-from or X-to header in the mail file contains incorrect syntax. Special characters < and > are found in the X-header text. These special characters are allowed to envelop X-header message text but they must be balanced (that is, the beginning of the message text contains < while the end of the message text should be >). *xxxxxxx* is the X-header text. The outbound mail request is rejected. Check the log for related messages for the data set name.

Action: Make sure the mail file data set was deleted. Check for message T01SM045I, which indicates it was deleted. Try to find what software generated this mail message with bad syntax and correct it. If the destination coded on the RETURN parm of the SMTP statement is LOCAL, a copy of the message can be found in one of the SYS*xxxxx* data sets of the started task in the class specified on the RETURN parm.

T01SM132I No Local user found in x-header: xxxxxxx

Reason: An X-from or X-to header in the mail file contains incorrect syntax. Only hosts and routes were found in the X-header. *xxxxxxx* is the X-header text. The outbound mail request is rejected. Check the log for related messages for the data set.

Action: Make sure the mail file data set was deleted. Check for message T01SM045I, which indicates it was deleted. Try to find what software generated this mail message with bad syntax and correct it. If the destination coded on the RETURN parm of the SMTP statement is LOCAL, a copy of the message can be found in one of the SYS*xxxxx* data sets of the started task in the class specified on the RETURN parm.

T01SM133I No host found in x-header: xxxxxxx

Reason: An X-from or X-to header in the mail file contains incorrect syntax. No host or route was found in the X-header. *xxxxxxx* is the X-header text. The outbound mail request is rejected. Check the log for related messages for the data set.

Action: Make sure the mail file data set was deleted. Check for message T01SM045I, which indicates it was deleted. Try to find what software generated this mail message with bad syntax and correct it. If the destination coded on the RETURN parm of the SMTP statement is LOCAL, a copy of the message can be found in one of the SYS*xxxxx* data sets of the started task in the class specified on the RETURN parm.

T01SM145I RCPT TO contains invalid user: *user* not in table

Reason: SSMTP received an invalid user (*user*) on the RCPT TO command for inbound mail. Exit SMTPUSR rejected the user as a valid user on the system. The RCPT TO command is rejected.

Action: If the user name is valid then the user must be entered in the SMTPUSR table or the SMTPUSR exit must be changed by the system administrator.

Chapter

32 T01SN*nnn* SNMP Messages

This chapter describes messages issued by Simple Network Management Protocol (SNMP). These messages are T01SN000 through T01SN999.

SNMP messages are written to SNMLOG by the SNMP program.

In the SNMCFG*nn* file, the LOGGING parameter determines the messaging level.

If set to:	Then:
PRODUCTION	All error (E) and informational (I) messages are issued.
TRACE	All trace (T) messages are issued.
DEBUG	All debug (D) messages are issued.
DUMP	All dump (P) messages are issued and are followed by either: ■ A formatted dump of a data structure ■ A hexadecimal dump of a buffer

Note: Each level also displays all messages from the lower level. For example, if LOGGING TRACE is set, then both Production and Trace messages are issued.

T01SN*nnn*

T01SN001T Snmp select failed: ERRNO *errno*

Reason: In the main loop, the SNMP mvsselect socket call returned with a -1 condition code. The associated errno is returned in the message.

Action: The errno codes for mvsselect are defined in your *Programmer's Reference*.

T01SN002T badVersion: *version (source)*

Reason: An SNMP packet was received from *source*, but the SNMP *version* is not supported.

Action: Verify that the management software on the source host is configured correctly.

T01SN003T badCommunity: *community (source)*

Reason: An SNMP packet was received from *source*, but the *community* name is not valid.

Action: Verify that the SNMCFG*nn* file or the source host is configured correctly.

T01SN004T authorizationFailure: *(source)*

Reason: An SNMP packet was received from *source*, but the view is not allowed for the community.

Action: Verify that the SNMCFG*nn* file or the source host is configured correctly.

T01SN005T badOperation: *opcode (source)*

Reason: An SNMP packet was received from *source*, but the *opcode* is not a valid SNMP opcode.

Action: Verify that the management software on the source host is configured correctly.

T01SN006T Trap = NULL

Reason: While preparing to send an SNMP trap, a data area had to be reallocated. The trap is sent, but if this continues, there is a potential for storage shortages.

Action: Monitor the SNMLOG. If the message is issued continuously, save the SNMLOG and contact Customer Support.

T01SN007T SOCKET call failed serrno is *errno*

Reason: While initializing SNMP an attempt to open the UDP socket failed.

Action: The errno codes for sockets are defined in your *Programmer's Reference*.

T01SN008T BIND call failed serrno is *errno*

Reason: While initializing SNMP an attempt to bind to the UDP socket failed.

Action: The errno codes for bind are defined in your *Programmer's Reference*.

T01SN009T Receive from failed

Reason: An attempt to read data from the SNMP UDP socket failed.

Action: Verify that the interface you want to use is operational. If so, save the Joblog and contact Customer Support.

T01SN010T SNS#SNM.make_dtrap_vbl: *value* in DPI Trap

Reason: An SNMP DPI trap packet was received from the sub-agent, but the *value* in the trap is not supported.

Action: Save the SNMLOG and contact Customer Support.

T01SN011T Object *oid* not DPI

Reason: Although the Object MIB being queried exists, it is not a DPI registered object.

Action: Verify that the management software on the source host is configured correctly.

T01SN012T OID *oid* not found

Reason: An SNMP DPI request was sent and the sub-agent sent a response. However, the Object ID (*oid*) in the response did not match an existing SNMP object.

Action: Verify that the sub-agent software is configured correctly.

T01SN013T Main loop: Session *session* timed out waiting for OID: *oid*

Reason: An SNMP DPI request for the specified *oid* was sent to a sub-agent, and the reply was not sent within the time-out period. The session is terminated. If LOGGING DUMP is specified in the SNMCFG*nn* file, a hex dump of the Wait Object data area and current time will follow.

Action: Verify the status of the sub-agent application.

T01SN014T SNPS#IO: io = NOTOK

Reason: A call to perform I/O returned a -1 condition code.

Action: Verify that the interface you want to use is operational. If so, save the Joblog and contact Customer Support.

T01SN015T read_dpi_socket: Socket closed rc=*rc*

Reason: An attempt to read the SNMP DPI port failed with return code *rc*.

Action: Verify the status of the sub-agent application.

T01SN016T read_dpi_socket: Socket closed rc<0, Errno = *errno*

Reason: An attempt to read the SNMP DPI port failed with an error condition code. The errno is displayed.

Action: Verify the status of the sub-agent application. The errno codes for read are defined in your *Programmer's Reference*.

T01SN017T read_dpi_socket: Socket closed rc=*rc*

Reason: An attempt to write to the SNMP DPI port failed with return code *rc*.

Action: Verify the status of the sub-agent application. The error codes for write are defined in your *Programmer's Reference*.

T01SN018T read_dpi_socket: Socket closed DPI RC= *rc* for MIB=*oid*

Reason: A non-zero return code was received from the sub-agent in a DPI Response packet for the specified *oid*.

Action: Check application or network manager software for errors.

Note: This can occur if the application registers MIBs before the support for them exists.

T01SN020D Error reading primitive, *type* bytes: *number*

Reason: While trying to parse a MIB element, an error was detected. The data type and size are displayed.

Action: Verify the status of the management software on the source host.

T01SN021D Error reading initial octet: *rc*

Reason: While reading from the SNMP port, an error occurred. The error return code is displayed.

Action: The errno codes for recvfrom are defined in your *Programmer's Reference*.

T01SN023D Error reading initial length octet: *rc*

Reason: While reading from the SNMP port, an error occurred. The error return code is displayed.

Action: The errno codes for recvfrom are defined in the your *Programmer's Reference*.

T01SN024D Error building indefinite constructor, *rc*

Reason: While reading a constructed type from the SNMP port, an error occurred. The error return code is displayed.

Action: The errno codes for recvfrom are defined in the your *Programmer's Reference*.

T01SN025D Error building cons., stream at *byteno*, wanted *length*: *rc*

Reason: While reading a constructed type from the SNMP port, an error occurred. The error return code is displayed.

Action: The errno codes for recvfrom are defined in the your *Programmer's Reference*.

T01SN028D SNPE#FR: WARNING: duplicately free'd pe 0x*address*

Reason: The memory address of a Presentation Element that is being freed a second time is displayed.

Action: Processing continues. If errors persist, save the SNMLOG and contact Customer Support.

T01SN029D SNMIBC: variable *object_name* has no value

Reason: While building a MIB header, the value of an object was not found.

Action: Processing continues. If errors persist, save the SNMLOG and contact Customer Support.

T01SN030D SNMIBC: unknown object *object_name*

Reason: While initializing the values of an object type data area, the object type for *object_name* was not found in the MIB database.

Action: Processing continues. If errors persist, save the SNMLOG and contact Customer Support.

T01SN031D SNMIBC: no getfnx for object *object_name*

Reason: While initializing the values of an object type data area, the pointer to the Get function for the object type was NULL.

Action: Processing continues. If errors persist, save the SNMLOG and contact Customer Support.

T01SN032D SNMIBC: non-generic getfnx for object *object_name*

Reason: While initializing the values of an object type data area, the pointer to the Get function for the object type points to a unique function.

Action: Use this debugging statement to aid troubleshooting SNMP problems.

T01SN033D SNMIBC: no syntax defined for object *object_name*

Reason: While initializing the values of an object type data area, the pointer to the syntax function for parsing and creating data type for the object type was NULL.

Action: Processing continues. If errors persist, save the SNMLOG and contact Customer Support.

T01SN034D DPI socket call failed serrno is *errno*

Reason: While initializing SNMP DPI an attempt to open the TCP socket failed.

Action: The errno codes for socket are defined in the your *Programmer's Reference*.

T01SN035D DPI BIND call failed: serrno is *errno*

Reason: While initializing SNMP DPI an attempt to bind to the TCP socket failed.

Action: The errno codes for bind are defined in the your *Programmer's Reference*.

T01SN037D DPI Buffer storage unavailable, retry in 1 sec

Reason: While reading from the DPI socket, there was not enough storage available to allocate a buffer for the data. After a pause, the allocation is retried.

Action: Monitor the SNMLOG. If the message is issued continuously, analyze the product's logs for the cause of the shortage of storage.

T01SN038D SNS#SNM.make_dtrap_vbl: Unable to get OID

Reason: An SNMP DPI trap was received from the sub-agent, but the Object ID cannot be converted into an SNMP Object ID.

Action: Save the SNMLOG and contact Customer Support.

T01SN040D SNS#SNM.make_dtrap_vbl: Storage unavail for *data_area*

Reason: An SNMP DPI trap was received from the sub-agent, but there is not enough storage to allocate the data area needed to create the SNMP trap to send to the network manager software.

Action: Check the product Joblog for a reason why storage was not available.

T01SN041D can't find entry 0x*hex_value*

Reason: While dumping the Presentation Element type, the PE type cannot be determined.

Action: Save the SNMLOG and contact Customer Support.

T01SN042D SNS#SNM.ps_alloc: out of memory

Reason: There was not enough storage available to allocate a buffer for the Presentation Stream.

Action: Monitor the SNMLOG. If the message is issued continuously, analyze the product logs for the cause of the shortage of storage.

T01SN043D SNS#SNM.dg_setup: *error*

Reason: The call to setup an SNMP Trap Presentation Stream failed with return code *error*.

Action: Processing continues. If errors persist, save the Joblog and contact Customer Support.

T01SN044D *view status oid* on view statement

Reason: While creating a view, it was found to already exist.

Action: Processing continues. Verify that the SNMCFG*nn* file is configured correctly.

T01SN045D SNVALUE: no syntax defined for object *obj_name*

Reason: There was no pointer to the syntax function to build a MIB element for an SNMP PDU.

Action: An SNMP error is returned to the management host. Save the SNMLOG and contact Customer Support.

T01SN046D SNVALUE: no value defined for object *obj_name*

Reason: There was no value for building a MIB element for an SNMP PDU.

Action: An SNMP error is returned to the management host. Save the SNMLOG and contact Customer Support.

T01SN047D SNVALUE: encoding error for variable *oid*

Reason: The function encoding the MIB element for an SNMP PDU returned a -1 condition code.

Action: An SNMP error is returned to the management host. Save the SNMLOG and contact Customer Support.

T01SN048D SNVALUE: no syntax defined for object *obj_name*

Reason: There was no pointer to the syntax function to build a MIB element for an SNMP Set PDU.

Action: An SNMP error is returned to the management host. Save the SNMLOG and contact Customer Support.

T01SN049D SNVALUE: no syntax defined for object *obj_name*

Reason: There was no pointer to the syntax function to build a MIB element for an SNMP PDU.

Action: An SNMP error is returned to the management host. Save the SNMLOG and contact Customer Support.

T01SN050D SNVALUE: encoding error for variable *oid*

Reason: The function encoding the MIB element for an SNMP PDU returned a -1 condition code.

Action: An SNMP error is returned to the management host. Save the SNMLOG and contact Customer Support.

T01SN051D SNVALUE: no syntax defined for object *obj_name*

Reason: There was no pointer to the syntax function to build a MIB element for an SNMP PDU.

Action: An SNMP error is returned to the management host. Save the SNMLOG and contact Customer Support.

T01SN052D SNVALUE: out of memory

Reason: There was not enough storage available to allocate a buffer for the MIB element.

Action: Monitor the SNMLOG. If the message is issued continuously, analyze the product logs for the cause of the shortage of storage.

T01SN053D SNVALUE: encoding error for variable *oid*

Reason: The function encoding the MIB element for an SNMP PDU returned a -1 condition code.

Action: An SNMP error is returned to the management host. Save the SNMLOG and contact Customer Support.

T01SN054D SNVALUE: no syntax defined for object *obj_name*

Reason: There was no pointer to the syntax function to build a MIB element for an SNMP PDU.

Action: An SNMP error is returned to the management host. Save the SNMLOG and contact Customer Support.

T01SN055D SNVALUE: encoding error for variable *oid*

Reason: The function encoding the MIB element for an SNMP PDU returned a -1 condition code.

Action: An SNMP error is returned to the management host. Save the SNMLOG and contact Customer Support.

T01SN056D SNVALUE: no syntax defined for object *obj_name*

Reason: There was no pointer to the syntax function to build a MIB element for an SNMP PDU.

Action: An SNMP error is returned to the management host. Save the SNMLOG and contact Customer Support.

T01SN057D SNVALUE: encoding error for variable *oid*

Reason: The function encoding the MIB element for an SNMP PDU returned a -1 condition code.

Action: An SNMP error is returned to the management host. Save the SNMLOG and contact Customer Support.

T01SN058D SNSYSTE.init_system: *object*: no syntax

Reason: No syntax to parse or build the data type for the MIB object was found.

Action: Processing continues. If errors persist, save the SNMLOG and contact Customer Support.

T01SN059D SNSYSTE.init_system: *object*: unknown object

Reason: The textual name of the object (sysDescr) could not be converted to an Object ID.

Action: Processing continues. If errors persist, save the SNMLOG and contact Customer Support.

T01SN060P dereg_dpi: No Object Type for DPI object

Reason: An SNMP DPI object extension was found in the DPI session chain, but the pointer to the object type data area is NULL.

Action: Use this buffer or data area dump to aid troubleshooting SNMP problems.

T01SN101T start_trap: trap message initialzed at location *location*

Reason: SNMP traps are initialized by creating a trap message template. This message indicates that the template was created and gives the memory location for reference.

Action: Use this trace statement to aid troubleshooting SNMP problems.

T01SN102T Trap sent: *generic specific*

Reason: An SNMP trap was sent including the generic and specific trap values.

Action: Use this trace statement to aid troubleshooting SNMP problems.

T01SN103T Kernel alert: COLD START

Reason: The TCP/IP kernel sent an alert that requires a trap to be sent to all designated network manager hosts.

Action: Use this trace statement to aid troubleshooting SNMP problems.

T01SN104T Kernel alert: LINK UP

Reason: The TCP/IP kernel was sent an alert that requires a trap to be sent to all designated network manager hosts.

Action: Use this trace statement to aid troubleshooting SNMP problems.

T01SN105T Kernel alert: LINK DOWN

Reason: The TCP/IP kernel was sent an alert that requires a trap to be sent to all designated network manager hosts.

Action: Use this trace statement to aid troubleshooting SNMP problems.

T01SN120T Operation: Get_Request

Reason: An SNMP Get request packet was received.

Action: Use this trace statement to aid troubleshooting SNMP problems.

T01SN121T Operation: Get_Next_Request

Reason: An SNMP Get Next Request packet was received.

Action: Use this trace statement to aid troubleshooting SNMP problems.

T01SN122T Operation: Get_Response

Reason: An SNMP Get Response packet was received.

Action: Use this trace statement to aid troubleshooting SNMP problems.

T01SN123T Operation: Set_Request

Reason: An SNMP Set Request packet was received.

Action: Use this trace statement to aid troubleshooting SNMP problems.

T01SN124T Operation: Trap

Reason: An SNMP trap packet was received.

Action: Use this trace statement to aid troubleshooting SNMP problems.

T01SN125T read_dpi_socket: DPI socket closed

Reason: When reading the SNMP DPI port, an EOF condition code was returned. Since the port is read after an mvsselect indicates that there is data waiting for it, this is unexpected and the socket is closed.

Action: Verify the status of the sub-agent application. Use this trace statement to aid troubleshooting SNMP problems.

T01SN130T SNENC.en_type: OBJID= *oid*

Reason: An SNMP PDU is being created. The Object ID is displayed.

Action: Use this trace statement to aid troubleshooting SNMP problems.

T01SN131T SNDECC.pr_type: OBJID= *oid*

Reason: An SNMP PDU is being parsed. The Object ID is displayed.

Action: Use this trace statement to aid troubleshooting SNMP problems.

T01SN140T DPI Listen socket is *socket*

Reason: During initialization, the DPI listen socket was opened successfully.

Action: Use this trace statement to aid troubleshooting SNMP problems.

T01SN141T DPI Listen port is *port*

Reason: During initialization, the DPI listen port was opened successfully.

Action: Use this trace statement to aid troubleshooting SNMP problems.

T01SN142T Initializing DPI Object

Reason: An SNMP DPI register request was received and the DPI MIB is added to the MIB database.

Action: Use this trace statement to aid troubleshooting SNMP problems.

T01SN201T SNMP Datagram receved from *source*

Reason: An SNMP packet was received from the indicated host.

Action: Use this trace statement to aid troubleshooting SNMP problems.

T01SN202T SNMP Datagram sent to *source*

Reason: An SNMP packet was sent to the indicated host.

Action: Use this trace statement to aid troubleshooting SNMP problems.

T01SN203T Trap Sent to *source*

Reason: An SNMP trap packet was sent to the indicated host.

Action: Use this trace statement to aid troubleshooting SNMP problems.

T01SN204T Select RC = *rc*

Reason: The return code from an mvsselect to check the SNMP socket is given.

Action: Use this trace statement to aid troubleshooting SNMP problems.

T01SN211T Receive from byte total = *bytes*

Reason: This follows message T01SN201T indicating the size of the packet received.

Action: Use this trace statement to aid troubleshooting SNMP problems.

T01SN212T SNMP Sendto byte total = *bytes*

Reason: This follows message T01SN202T indicating the size of the packet sent.

Action: Use this trace statement to aid troubleshooting SNMP problems.

T01SN213T Trap Sendto byte total = *bytes*

Reason: This follows message T01SN203T indicating the size of the packet sent.

Action: Use this trace statement to aid troubleshooting SNMP problems.

T01SN214T SNPS#IO: io = OK

Reason: A call to perform I/O returned condition code zero.

Action: Use this trace statement to aid troubleshooting SNMP problems.

T01SN301I SNMP Agent Logging Started

Reason: The SNMP agent task has started and is logging.

Action: None. This is an Informational message.

T01SN302I Logging Ended

Reason: The SNMP agent task stopped and is no longer logging.

Action: None. This is an Informational message.

T01SN303I SNDPI: Called with Port = *port*

Reason: The SNMP DPI MIB object type is to be initialized. The value of the object is port.

Action: None. This is an Informational message.

T01SN401E FAILED to start_udp_server. SNM retry in 30 seconds.

Reason: During initialization, the attempt to start the SNMP server listening on the UDP port failed, but will be attempted again.

Action: Monitor the SNMLOG. If the message is issued continuously, verify that the API and DNR task groups are configured properly. Also, verify that the interface you want to use is operational.

T01SN402E FAILED to start trap server. SNM retry in 30 seconds.

Reason: During initialization, the attempt to bind the SNMP trap server to the UDP port failed, but will be attempted again.

Action: Monitor the SNMLOG. If the message is issued continuously, verify that the API and DNR task groups are configured properly. Also, verify that the interface you want to use is operational.

T01SN403E FAILED to start_dpi_server. SNM retry in 30 seconds.

Reason: During initialization, the attempt to bind the SNMP trap server to the UDP port failed, but will be attempted again.

Action: Monitor the SNMLOG. If the message is issued continuously, verify that the API and DNR task groups are configured properly. Also, verify that the interface you want to use is operational.

T01SN404E Received BAD trap type: *trap*

Reason: An internal SNMP error occurred between the SNMP agent and the TCP/IP stack.

Action: Processing continues. If errors persists, save the SNMLOG and contact Customer Support.

T01SN405E Unable to allocate memory *n*

Reason: An SNM task group internal error.

Action: Processing continues. If errors persist, save the SNMLOG and contact Customer Support.

T01SN406E Unable to send/receive PDUs *n source*

Reason: The SNMP agent is unable to receive a PDU over the UDP port from the specified host.

Action: Verify that the interface you want to use is operational. If errors persist, save the SNMLOG and contact Customer Support.

T01SN407E Invalid timeout value, default taken

Reason: The timeout value coded in the SNMCFG*xx* DPI statement was invalid.

Action: Correct the value in SNMCFG*nn*.

T01SN408E Unable to decode_SNMP_Message: *reason source*

Reason: A format error occurred attempting to read the PDU from the specified host.

Action: Processing continues. If errors persist, save the SNMLOG and contact Customer Support.

T01SN409E Unable to decode_SNMP_DPI_Message: *reason source*

Reason: A format error occurred attempting to read the PDU from the specified host.

Action: Processing continues. If errors persist, save the SNMLOG and contact Customer Support.

T01SN411E Unable to encode_SNMP_Message: *reason source*

Reason: A format error occurred attempting to create the PDU to be sent to the specified host.

Action: Processing continues. If errors persist, save the SNMLOG and contact Customer Support.

T01SN412E Could not send PDU *reason source*

Reason: The SNMP agent is unable to establish a UDP connection with the specified host.

Action: Processing continues. If errors persist, save the SNMLOG and contact Customer Support.

T01SN417E unable to initialize trap structure: no memory

Reason: An internal SNM task group error attempting to send a trap.

Action: Processing continues, however no traps are sent. If errors persist, save the SNMLOG and contact Customer Support.

T01SN418E Can't resolve hostname of self, no traps: herrno errno

Reason: The DNR call *gethostname()* failed for the MVS *localhost* IP address while initializing the SNM task group. The herrno value errno is described in *Unprefixed Messages and Codes.* Related DNR error messages can be found in the DNRLOG DD.

Processing continues, but no traps are sent.

Action: Verify that the product can resolve its own host name. Use DNRGET to verify that DNR is configured correctly. If necessary, stop and restart the DNR and SNM task groups.

T01SN419E *myhost*: can't resolve hostname of self

Reason: The DNR call *gethostbyname()* failed for the MVS *localhost* IP address while initializing the SNM task group. Related DNR error messages can be found in the DNRLOG DD.

Processing continues, but no traps are sent.

Action: Verify that the product can resolve its own host name. Use DNRGET to verify that DNR is configured correctly. If necessary, stop and restart the DNR and SNM task groups.

T01SN420E Unable to send trap: no such object: sysObjectID

Reason: An SNM task group internal error processing the System group Object ID, which is the enterprise MIB Object ID for the product.

Action: Processing continues. Save the SNMLOG and contact Customer Support.

T01SN421E unable to send trap: sysObjectID has no value

Reason: An SNM task group internal error processing the System group Object ID that is the enterprise MIB Object ID for the product.

Action: Processing continues. Save the SNMLOG and contact Customer Support.

T01SN422E Unable encode_SNMP_Message: *reason*

Reason: A format error occurred attempting to create the PDU to be sent to the specified host.

Action: Processing continues. If errors persist, save the SNMLOG and contact Customer Support.

T01SN425E Trap failed: *reason*

Reason: The SNMP agent was unable to send a trap. The *reason* is a text reason of the error.

Action: Processing continues. If errors persist, save the SNMLOG and contact Customer Support.

T01SN426E Unable to open config file

Reason: The SNMCFG*nn* member specified on the START SNM command was not found or is in error.

Action: Processing terminates. Verify that the SNMCFG*nn* member is in the SYSPARM data set concatenation and is not in error.

T01SN427E Invalid configuration statement: *statement*

Reason: A configuration statement is invalid or contains an invalid keyword. The statement in error is displayed.

Action: Correct the configuration statement and restart the SNM task group.

T01SN428E SNS#SNM: unexpected operation from: source

Reason: A trap was received from the specified host. Since SNMP is an agent, it should only send traps.

Action: Verify that the source host SNMP software is configured correctly.

T01SN430E PE not found for DPI response

Reason: An SNMP request was received for a DPI MIB. The request was forwarded to the sub-agent and a response was received. However, the SNMP Presentation Element you want to use to create the SNMP response was not found associated with the Object ID in the DPI response.

Action: Verify that the DPI sub-agent software is configured correctly. If errors persist, save the SNMLOG and contact Customer Support.

T01SN431E PS not found for DPI response

Reason: An SNMP request was received for a DPI MIB. The request was forwarded to the sub-agent and a response was received. However, the SNMP Presentation Stream you want to use to create the SNMP response was not found associated with the Object ID in the DPI response.

Action: Verify that the DPI sub-agent software is configured correctly. If errors persist, save the SNMLOG and contact Customer Support.

T01SN440E DPI LISTEN call failed: serrno is *errno*

Reason: During initialization, an attempt to listen on the DPI port failed with condition code *errno*.

Action: The errno codes for listen are defined in the your *Programmer's Reference*.

T01SN441E DPI write error *errno*, for socket *socket*

Reason: An attempt to send a DPI packet to the sub-agent in session on the specified socket failed.

Action: The errno codes for write are defined in the your *Programmer's Reference*.

T01SN442E DPI Get Port not accepted, rc=*rc*

Reason: An attempt to accept a session request on the DPI listen port failed with the specified return code.

Action: The return codes for accept are defined in the your *Programmer's Reference*.

T01SN443E Unable to allocate session memory

Reason: While attempting to accept a session request on the DPI listen, the attempt to allocate the data_area for the session failed.

Action: Analyze the logs for the cause of the shortage of storage. If errors persist, save the Joblog and contact Customer Support.

T01SN446E DPI Register Object Syntax failed: RC= *rc*

Reason: When a DPI Register request is received, a DPI Get request is sent to the sub-agent to determine the data type of the MIB. The attempt to write to the DPI port failed.

Action: The errno codes for write are defined in the your *Programmer's Reference*.

T01SN501E Too many views: view *view* ignored

Reason: The VIEW statement for *view* exceeds the maximum number of views allowed.

Action: The maximum number of views is 31. Reduce the number of views defined in SNMCFG*xx* and restart the SNM task group.

T01SN502E Unknown OID *oid* as a view

Reason: The Object ID defining the view is NULL.

Action: Verify that the SNMCFG*nn* file is defined correctly.

T01SN503E SNVIEW#.f_view: duplicate view

Reason: The VIEW *view* was already specified on another VIEW statement.

Action: Remove the duplicate view statement and restart the SNM task group.

T01SN504E Unknown OID *oid* as a view

Reason: The Object ID defining the view was not found in the MIB database.

Action: Verify that the SNMCFG*nn* file is defined correctly.

T01SN506E Invalid address *ip_address* for community *community*

Reason: The specified *community* contains an invalid IP address.

Action: Verify that the SNMCFG*nn* file is defined correctly.

T01SN507E Bad access mode *mode* for community *community*

Reason: An invalid access mode was specified for a community.

Action: Verify that the SNMCFG*nn* file is defined correctly.

T01SN508E No such view as *view* for community *community*

Reason: A view was placed on a COMMUNITY statement that did not have a corresponding VIEW statement.

Action: Verify that the SNMCFG*nn* file is defined correctly.

T01SN509E SNDPI: *oid*: unknown object

Reason: The DPI MIB object is unknown.

Action: Processing continues. If errors persist, save the SNMLOG and contact Customer Support.

T01SN602D SNENC.en_type: *type*

Reason: An SNMP PDU is being created. The data type is displayed as text.

Action: Use this debugging statement to aid troubleshooting SNMP problems.

T01SN603D SNENC: offset=*offset* value=*value*

Reason: An SNMP PDU is being created. The SNMP type is represented by the offset. The actual value of the MIB element is also displayed.

Action: Use this debugging statement to aid troubleshooting SNMP problems.

T01SN604D SNENC.en_seq type *type*

Reason: An SNMP PDU sequence is being created. The type is displayed as a number.

Action: Use this debugging statement to aid troubleshooting SNMP problems.

T01SN605D SNENC.en_set type *type*

Reason: An SNMP PDU set is being created. The type is displayed as a number.

Action: Use this debugging statement to aid troubleshooting SNMP problems.

T01SN606D SNENC.en_seqof type *type*

Reason: An SNMP PDU sequence-of is being created. The *type* is displayed as a number.

Action: Use this debugging statement to aid troubleshooting SNMP problems.

T01SN607D SNENC.en_setof type *type*

Reason: An SNMP PDU set-of is being created. The *type* is displayed as a number.

Action: Use this debugging statement to aid troubleshooting SNMP problems.

T01SN608D SNENC.en_choice: cnt *count*

Reason: An SNMP PDU choice is being created. The *count* is displayed as a number.

Action: Use this debugging statement to aid troubleshooting SNMP problems.

T01SN610D SNOBJEC.add_objects: OID=*oid*

Reason: The SNMP object is being added to the MIB database.

Action: Use this debugging statement to aid troubleshooting SNMP problems.

T01SN611D ASNOBJEC: got name *oid*, but not object

Reason: The Object ID was found, but the object type data area was not found.

Action: Use this debugging statement to aid troubleshooting SNMP problems.

T01SN612D SNOBJEC.add_objects: Overwriting DPI object

Reason: An SNMP DPI register packet was received for an object that was already in the MIB database. RFC 1229 specifies that the new object takes precedence over the existing object.

Action: Use this debugging statement to aid troubleshooting SNMP problems.

T01SN613D SNOBJEC.add_objects: Overwriting *product_name* object not allowed

Reason: By default, the product's MIBs cannot be overwritten. The default can be changed in the SNMCFG*nn* file.

Action: Use this debugging statement to aid troubleshooting SNMP problems.

T01SN615D SNS#SNM.make_dtrap_vbl: Value Set for *data_type*

Reason: An SNMP DPI trap packet is being created and the data type to be sent is shown.

Action: Use this debugging statement to aid troubleshooting SNMP problems.

T01SN652D SNDECC.pr_type: *type*

Reason: An SNMP PDU is being parsed. The *type* is displayed in text.

Action: Use this debugging statement to aid troubleshooting SNMP problems.

T01SN653D SNDECC.pr_etype: integer *value*

Reason: An SNMP PDU is being parsed. The integer type *value* is displayed.

Action: Use this debugging statement to aid troubleshooting SNMP problems.

T01SN654D SNDECC.pr_seq: type *type*

Reason: An SNMP PDU sequence is being parsed. The *type* is displayed as a number.

Action: Use this debugging statement to aid troubleshooting SNMP problems.

T01SN655D SNDECC.pr_set type *type*

Reason: An SNMP PDU set is being parsed. The *type* is displayed as a number.

Action: Use this debugging statement to aid troubleshooting SNMP problems.

T01SN656D SNDECC.pr_seqof type *type*

Reason: An SNMP PDU sequence-of is being parsed. The *type* is displayed as a number.

Action: Use this debugging statement to aid troubleshooting SNMP problems.

T01SN657D SNDECC.pr_setof type *type*

Reason: An SNMP PDU set-of is being parsed. The *type* is displayed as a number.

Action: Use this debugging statement to aid troubleshooting SNMP problems.

T01SN658D fix_mem:inconsistency

Reason: Memory was allocated for an optional object that is not present. This is not allowed, or else the object is considered to be present and attempts to process it are made. However, the Presentation Element type passed to the fix_mem function is incorrect.

Action: Use this debugging statement to aid troubleshooting SNMP problems.

T01SN659D setdval: type not implemented

Reason: An SNMP PDU is being parsed. The invalid type is displayed as a number.

Action: Use this debugging statement to aid troubleshooting SNMP problems.

T01SN690D Dequeing DPI Object from Get_Next chain:

Reason: An SNMP DPI object is to be removed from the MIB database. If LOGGING DUMP is specified in the SNMCFG*nn* file, then a formatted dump of the object data area follows.

Action: Use this debugging statement to aid troubleshooting SNMP problems.

T01SN691D Dequeing outstanding DPI Request from Wait chain:

Reason: An SNMP DPI request that has not received a response is to be removed from the wait chain. If LOGGING DUMP is specified in the SNMCFG*nn* file, a hex dump of the Wait Object data area will follow.

Action: Use this debugging statement to aid troubleshooting SNMP problems.

T01SN693D Send SNMP Reply for Object *oid*

Reason: An SNMP DPI object is to be removed from the MIB database but there was an SNMP request for the object's value. A reply is sent with an SNMP error message.

Action: Use this debugging statement to aid troubleshooting SNMP problems.

T01SN701D SNINTER.get_*structure_type* entry

Reason: The routine to build the MIB elements for the Interfaces group was called.

Action: Use this debugging statement to aid troubleshooting SNMP problems.

T01SN702D SNIPC.get_*structure_type* entry

Reason: The routine to build the MIB elements for the IP group was called.

Action: Use this debugging statement to aid troubleshooting SNMP problems.

T01SN703D SNICMPC.get_*structure_type* entry

Reason: The routine to build the MIB elements for the ICMP group was called.

Action: Use this debugging statement to aid troubleshooting SNMP problems.

T01SN704D SNTCPC.get_*structure_type* entry

Reason: The routine to build the MIB elements for the TCP group was called.

Action: Use this debugging statement to aid troubleshooting SNMP problems.

T01SN705D get_udp_*structure_type* entry

Reason: The routine to build the MIB elements for the UDP group was called.

Action: Use this debugging statement to aid troubleshooting SNMP problems.

T01SN711D SNINTER.get_*structure_type* exit

Reason: The routine to build the MIB elements for the Interfaces group completed.

Action: Use this debugging statement to aid troubleshooting SNMP problems.

T01SN712D SNIPC.get_*structure_type* exit

Reason: The routine to build the MIB elements for the IP group completed.

Action: Use this debugging statement to aid troubleshooting SNMP problems.

T01SN713D SNICMPC.get_*structure_type* exit

Reason: The routine to build the MIB elements for the ICMP group completed.

Action: Use this debugging statement to aid troubleshooting SNMP problems.

T01SN714D SNTCPC.get_*structure_type* exit

Reason: The routine to build the MIB elements for the TCP group completed.

Action: Use this debugging statement to aid troubleshooting SNMP problems.

T01SN715D get_udp_*structure_type* exit

Reason: The routine to build the MIB elements for the UDP group completed.

Action: Use this debugging statement to aid troubleshooting SNMP problems.

T01SN720D Main loop: *ecb_name* posted *ecb_value*
UDP port has data: socket
DPI session port socket has data

Reason: In the main loop of the SNMP program, the mvsselect was returned. This message indicates the reason for the return. Either an ECB was posted, or a socket has data.

Action: Use this debugging statement to aid troubleshooting SNMP problems.

T01SN721D dpiacce: o_generic RC=*rc*
dpiacce: s_generic RC=*rc*

Reason: An SNMP DPI port accepted data and is parsing it. The return code from o_generic or s_generic indicates whether the parsing was successful.

Action: Use this debugging statement to aid troubleshooting SNMP problems.

T01SN722D doit_udp: RC=*rc*

Reason: An SNMP port accepted data and parsed it. The return code indicates whether the parsing was successful. The function doit_udp calls the function process.

Action: Use this debugging statement to aid troubleshooting SNMP problems.

T01SN723D process: RC=*rc*

Reason: An SNMP port accepted data and is in the process of parsing it. The return code indicates whether the parsing was successful. The function process calls the function do_operation.

Action: Use this debugging statement to aid troubleshooting SNMP problems.

T01SN724D do_operation: RC=*rc*

Reason: An SNMP port accepted data and is in the process of parsing it. The return code indicates whether the parsing was successful. The function do_operation calls the function do_pass.

Action: Use this debugging statement to aid troubleshooting SNMP problems.

T01SN725D do_pass: RC=*rc*

Reason: An SNMP port accepted data and is in the process of parsing it. The return code indicates whether the parsing was successful. The function do_pass calls a function pointed to by the object type data area.

Action: Use this debugging statement to aid troubleshooting SNMP problems.

T01SN801D SNPE2PS.pe2ps_aux: RC= *rc*

Reason: The result of an attempt to build an SNMP PDU is displayed.

Action: Use this debugging statement to aid troubleshooting SNMP problems.

T01SN820D SNSYNTA.add_integer: Adding syntax for integers

Reason: The function indicated creates pointer structures to the routines to parse and create integer data types.

Action: Use this debugging statement to aid troubleshooting SNMP problems.

T01SN821D SNSYNTA.add_string: Adding syntax for strings

Reason: The function indicated creates pointer structures to the routines to parse and create string data types.

Action: Use this debugging statement to aid troubleshooting SNMP problems.

T01SN822D SNSYNTA.add_object: Adding syntax for objects

Reason: The function indicated creates pointer structures to the routines to parse and create object ID data types.

Action: Use this debugging statement to aid troubleshooting SNMP problems.

T01SN823D SNSYNTA.add_null: Adding syntax for nulls

Reason: The function indicated creates pointer structures to the routines to parse and create NULL data types.

Action: Use this debugging statement to aid troubleshooting SNMP problems.

T01SN824D SNSYNTA.add_ipaddr: Adding syntax for ip addrs

Reason: The function indicated creates pointer structures to the routines to parse and create integer data types.

Action: Use this debugging statement to aid troubleshooting SNMP problems.

T01SN825D SNSYNTA.add_netaddr: Adding syntax for net addrs

Reason: The function indicated creates pointer structures to the routines to parse and create hardware address data types.

Action: Use this debugging statement to aid troubleshooting SNMP problems.

T01SN826D SNSYNTA.add_counter: Adding syntax for counters

Reason: The function indicated creates pointer structures to the routines to parse and create counter data types.

Action: Use this debugging statement to aid troubleshooting SNMP problems.

T01SN827D SNSYNTA.add_gauge: Adding syntax for gauges

Reason: The function indicated creates pointer structures to the routines to parse and create gauge data types.

Action: Use this debugging statement to aid troubleshooting SNMP problems.

T01SN828D SNSYNTA.add_timeticks: Adding syntax for time

Reason: The function indicated creates pointer structures to the routines to parse and create timetick data types.

Action: Use this debugging statement to aid troubleshooting SNMP problems.

T01SN829D SNSYNTA.add_syntax: Adding syntax structure

Reason: This is a common function for creating pointer structures to the routines to parse and create data types.

Action: Use this debugging statement to aid troubleshooting SNMP problems.

T01SN840D Calling init_dpi with *port*

Reason: During initialization, the init_dpi routine is called to initialize the DPI MIB object and anchor the DPI session chain.

Action: Use this debugging statement to aid troubleshooting SNMP problems.

T01SN841D Accept completed cc=cc

Reason: An attempt to accept a session request on the DPI listen port succeeded and the socket number you should use for the session is *cc*.

Action: Use this debugging statement to aid troubleshooting SNMP problems.

T01SN842D read_dpi_socket: Read Length: *length*

Reason: An SNMP DPI packet has been received. The first two bytes are the length of the packet. This indicates how many of the first two bytes were read.

Action: Use this debugging statement to aid troubleshooting SNMP problems.

T01SN843D read_dpi_socket: Read Data Length: *length*

Reason: An SNMP DPI packet was received. This indicates how many of the data bytes were read.

Action: Use this debugging statement to aid troubleshooting SNMP problems.

T01SN845D Read_dpi_socket: RC=*rc*

Reason: A call to read_dpi_socket returned the specified return code representing the total number of bytes read.

Action: Use this debugging statement to aid troubleshooting SNMP problems.

T01SN846D Write_dpi_socket completed cc=cc

Reason: A call to write_dpi_socket returned the specified return code representing the total number of bytes written.

Action: Use this debugging statement to aid troubleshooting SNMP problems.

T01SN850D dpia: o_generic rc=*rc*
dpia: s_generic rc=*rc*

Reason: An SNMP DPI response has been received and parsed. The return code from the parsing attempt is displayed.

Action: Use this debugging statement to aid troubleshooting SNMP problems.

T01SN851D doit_udp: No Session waiting for response

Reason: When it is determined the an inbound SNMP packet is a DPI registered MIB, then the Presentation Element and Presentation Stream need to be saved to send the SNMP response. This message indicates that the PDU parsing did not set the session flag to indicate that an SNMP DPI request was sent.

Action: Use this debugging statement to aid troubleshooting SNMP problems.

T01SN854D dpiacce: SNMP Request received

Reason: An SNMP request for a DPI MIB was received.

Action: Use this debugging statement to aid troubleshooting SNMP problems.

T01SN855D dpiaacce: DPI Response received

Reason: An SNMP DPI response was received.

Action: Use this debugging statement to aid troubleshooting SNMP problems.

T01SN856D dpiaacce: Deregistering object

Reason: An SNMP DPI response was not received within the time-out period. The DPI session is closed and an SNMP error code is sent to the requesting SNMP manager host.

Action: Use this debugging statement to aid troubleshooting SNMP problems.

T01SN901P Object type:

Reason: A formatted dump of the object type data area follows.

Action: Use this buffer or data area dump to aid troubleshooting SNMP problems.

T01SN902P NULL tpe

Reason: If an error occurs processing a Presentation Element type, the function *pepsylose* returns a text reason of the error. This message is issued if the PE type is NULL.

Action: Use this buffer or data area dump to aid troubleshooting SNMP problems.

T01SN903P NULL modtyp

Reason: If an error occurs processing a Presentation Element type, the function *pepsylose* returns a text reason of the error. This message is issued if the module type passed to pepsylose is NULL.

Action: Use this buffer or data area dump to aid troubleshooting SNMP problems.

T01SN904P SNS#SNM.make_dtrap_vbl: OID *oid* has value of:

Reason: An SNMP DPI trap is being created. The Object ID sent by the sub-agent follows.

Action: Use this buffer or data area dump to aid troubleshooting SNMP problems.

T01SN905P SNPE#AL.pe_alloc: class=*class* form=*form* id=*id*

Reason: When allocating a Presentation Element, the encoding type bit fields, class, form, and id, are displayed.

Action: Use this debugging statement to aid troubleshooting SNMP problems.

T01SN906P SNMP: *operation* type *value* + *value*

Reason: While processing a Presentation Element type, the type *value* can be a combination of two values. This displays the actual makeup of the type.

Action: Use this buffer or data area dump to aid troubleshooting SNMP problems.

T01SN907P SNENC.en_obj: Encoding data:

Reason: An SNMP PDU is being created. A formatted dump of the Presentation Element type data area follows.

Action: Use this buffer or data area dump to aid troubleshooting SNMP problems.

T01SN908P SNDECC.pr_obj: Decoding data:

Reason: An SNMP PDU is being parsed. A formatted dump of the Presentation Element type data area follows.

Action: Use this buffer or data area dump to aid troubleshooting SNMP problems.

T01SN909P SNPS2PE.ps_readid: class=*class* form=*form* id=*id*

Reason: While parsing an SNMP PDU, the encoding type bit fields, class, form, and id, are displayed.

Action: Use this debugging statement to aid troubleshooting SNMP problems.

T01SN940P Received following SNMP-DPI packet:

Reason: An SNMP DPI packet was received. A hexadecimal dump of the packet follows.

Action: Use this buffer or data area dump to aid troubleshooting SNMP problems.

T01SN941P Parsed a SNMP_DPI_*type* packet:

Reason: An SNMP DPI packet was received and parsed. A hexadecimal dump of the data area representing the header follows.

Action: Use this buffer or data area dump to aid troubleshooting SNMP problems.

T01SN942P Dump the *type* packet:

Reason: An SNMP DPI packet was received and parsed. A hexadecimal dump of the data area representing the data follows.

Action: Use this buffer or data area dump to aid troubleshooting SNMP problems.

T01SN943P Sent the following SNMP-DPI packet:

Reason: An SNMP DPI packet was sent to a sub-agent. A hexadecimal dump of the packet follows.

Action: Use this buffer or data area dump to aid troubleshooting SNMP problems.

T01SN944P read_dpi_socket: Session structure:

Reason: The return from mvsselect indicated that there is data waiting for an SNMP DPI session. The call to read_dpi_socket passes the session data area. A hexadecimal dump of that data area follows.

Action: Use this buffer or data area dump to aid troubleshooting SNMP problems.

T01SN945P dpiacce: Object Type extension

Reason: An SNMP request was accepted and determined to be for a DPI MIB. The Object Type data area for that MIB element was found. A hexadecimal dump of the DPI extension for the Object Type follows.

Action: Use this buffer or data area dump to aid troubleshooting SNMP problems.

T01SN947P add_obj: enq OT: *obj, prevobj, nextobj*

Reason: An Object Type data area is being added to the MIB database. The memory address of the object type being added and the memory addresses of the previous and next object types in the Get Next chain are displayed.

Action: Use this buffer or data area dump to aid troubleshooting SNMP problems.

T01SN950P Wait Request Queued (*type*):

Reason: An SNMP DPI request was sent to the application, and the Wait Object was queued to the end of the wait chain. If LOGGING DUMP is specified in the SNMCFG*nn* file, then a hex dump of the Wait Object data area follows.

Action: Use this buffer or data area dump to aid troubleshooting SNMP problems.

T01SN951P Saving DPI PE/PS

Reason: An SNMP request was received for a DPI MIB. While the application is being queried, the Presentation Element and the Presentation Stream are saved until the SNMP response can be sent. If LOGGING DUMP is specified in the SNMCFG*nn* file, a formatted dump of the Object data area and a hex dump of the Presentation Element and the Presentation Stream follow.

Action: Use this buffer or data area dump to aid troubleshooting SNMP problems.

T01SN952P Wait Request Dequeued:

Reason: An SNMP DPI request was received from the application, and the Wait Object was dequeued from the wait chain. If LOGGING DUMP is specified in the SNMCFG*nn* file, a hex dump of the Wait Object and the Object extension data areas follow.

Action: Use this buffer or data area dump to aid troubleshooting SNMP problems.

Chapter 33

T01SO*nnn* Socket API Layer Messages

This chapter describes messages issued by the socket API layer. These messages are T01SO000 through T01SO999.

Messages in this range detail information about the network protocol layers, Transmission Control Protocol (TCP), Internet Protocol (IP), Internet Control Message Protocol (ICMP), and User Datagram Protocol (UDP). These messages help someone familiar with the TCP/IP suite of protocols to troubleshoot problems arising in this area on the local system, remote system, or network.

T01SO*nnn*

Message Format

These messages have the following general format:

code type: *text* (*protocol*) HOST=*host_addr* PORTS=*lport*:*rport* GATEWAY=*gw_addr host_name*

Variable	Description
code	Code number, in the format ACC30*x*I, where *x* can be an integer zero through three.
type	General description of the message. This provides the networking software administrator with an idea about the severity and location of the problem. Any of the following substrings can appear in the type field: MISC NOTE—Generally indicates some minor event or error that occurred with one of the protocols. This type of message has no severity. LCL EVENT—Indicates a protocol event that occurred on the local machine. The severity of this message is very low and normally requires no action.

Variable	Description
type (continued)	RMT ERROR—Indicates that the remote host or gateway did not respond properly at the specified protocol layer. Examples of this type of error include the remote system/gateway sending a TCP packet with a bad checksum or responding incorrectly at the current TCP state LCL ERROR—Indicates a problem with the network protocols on the local machine. Problems with the implementation of TCP/IP may show themselves here or the error may be caused by an application layer module.
text	Brief description of the event or error. Each message that uses a different text field that is listed in the following sections with a description of each.
protocol	String of three characters indicating the protocol that displayed the message. Possible values are: TCP Transmission Control Protocol ICM Internet Control Message protocol ECO ICMP Echo (ping) UDP User Datagram Protocol IP IP Protocol GWY Gateway ptask and IP protocol This field is optional.
host_addr	Internet address in dot notation of the host to which the local system is communicating. This field is optional.
lport	Protocol's pseudo-port on the local machine. This parameter is optional.
rport	Protocol's pseudo-port on the remote machine. This parameter is optional.
gw_addr	Internet address in dot notation of the first host gateway currently in use when communicating with the preceding host address. This parameter is optional.
host_name	Remote host textual name.

TCP State Field

Most of the TCP messages have a four-character subfield in the text field. This subfield (state) contains one of the following values:

State	Description
NULL	The local TCP is initializing. Once completed, the NULL state is exited to the listen state.
LSTN	(Listen). This state is entered after TCP receives a passive open request from a User Level Protocol (ULP), indicating TCP should wait for a connection request from a remote TCP.
SYNS	(SYN sent). The local TCP sent an open request (SYN) and is waiting for a matching open request connection from the remote TCP.
SYNR	(SYN received). The local TCP received and sent matching open requests from and to the remote TCP. It is now awaiting a confirming connection request acknowledgment (ACK) from the remote TCP.
ESTB	(Connection established). The local and remote TCPs completed the open sequence and a path is now available for the two hosts to communicate with each other.
CLSW	(Close wait). The local TCP received a connection close request (FIN) from the remote TCP and is waiting for the local ULP to issue a close request.
FINW	(FIN wait). The local TCP is waiting for either a close connection request (FIN) or an acknowledgment for a close connection request from the remote TCP.
CLSG	(Closing). In this state the local TCP is waiting for the acknowledgment of a close connection request from the remote TCP.
LACK	(A wait ACK on local FIN state). This state represents having sent both the close connection request and close connection acknowledgment and now waiting for the last ACK from the remote TCP.
TIMW	(Time wait). Waiting for enough time to pass to ensure that the remote TCP has received the ACK of its close request.
RATW	(Remote abort). When attempting to open a connection to a remote TCP, a TCP reset is returned indicating that the remote TCP refuses the connection.

T01SO001I Sockets API initialization Successful

Reason: Initialization of the Socket API environment completed successfully.

Action: None. This is an Informational message.

T01SO002E Configuration member IJTCFGxx did not contain a POOLDEF options statement for 'NAME(*name*)'

Reason: A required POOLDEF configuration statement was not coded in the configuration member IJTCFG*xx*.

Action: Add a POOLDEF statement for pool "NAME(*name*)"

T01SO003E Sockets API initialization failed - reason code = *code*

Reason: Initialization of the Socket API failed.

Values for *code* are:

00	Success.
04	Error allocating SAVT.
08	Error allocating SAVX.
0C	Error allocating ASVB.
10	Error freeing SAVX after error allocating ASVB.
14	Error allocating SPCB.
18	Error freeing ASVB after error allocating SPCB.
96	Error parsing the TCPCFG*xx* member. This message is accompanied by an error message or messages indicating the nature of the configuration error. Refer to those messages for appropriate corrective action..
204	IUVT allocation failed.
208	SSCVT allocation failed.
212	VMCF allocation failed.
216	IUVX allocation failed.
220	ASVB allocation failed.
224	IASB pool header load failed.
228	EORA pool header load failed.
232	PTPA pool header load failed.

236 IUMS pool header load failed.

240 Subsystem name already active.

304 Error creating data space.

308 Error establishing addressability to data space.

312 Error allocating data space latch.

316 Error loading data space pool header.

320 Error allocating data space storage.

Action: If insufficient virtual storage exists in the region to allocate the necessary control blocks, stop the product and restart it with a larger REGION size.

T01SO006E Socket end point create failed, Reason code = *value1*, Reason code = *value2*

Reason: An attempt to create a new socket following an inbound connection request failed. The reason code specified by *value1* represents the return code from T01XCREA. The reason code specified by *value2* is the diagnostic code returned by T01XCREA.

Action: Normally a call to T01XCREA should not fail. Contact Customer Support.

T01SO007E Invalid SAW queued to *q_name*, F = *func_code*, SEPM addr = *sepm_addr*, SAW addr = *saw_addr*

Reason: An attempt to process an asynchronous socket API unit of work failed because the Socket API Work element (SAW) at address *saw_addr* was invalid. The SAW contained either an invalid function code *func_code* for the queue *q_name* it was placed upon or the Socket End Point Main (SEPM) control block address in the SAW did not match the SEPM on which the SAW was placed.

A queue name of SEPMSAWQ represents the SEPM work queue for a connection; a queue name of SPCBWORK is the provider work queue.

Action: Dump the address space via the operator console and contact Customer Support.

T01SO009E Corrupted SAPI control block *cntl_blck* found in module *mod_name*, at address *cntl_addr*

Reason: The SAPI control block *cntl_blck* was found to be corrupted or invalid in module *mod_name*. Address *cntl_addr* is where the control should begin.

An attempt is made to take an SVCDUMP of the base product address space.

Action: Contact Customer Support.

T01SO010I TCP/IP stopping, phase = *phase*, active API address = 0

Reason: The stack is in the process of orderly shutdown, and is in phase *phase*.

phase The phase can be Drain, Stop, or Term (Terminate)

T01SO011I TCP/IP stopping, phase = Term , waiting for *cont_thread* control thread to end

Reason: The stack is in the termination phase of orderly shutdown and is waiting for the *cont_thread* control thread to end.

cont_thread The control thread can be OE, IUCV, TLI, or HPNS.

Action: None. This is an Informational message.

T01SO012R Enter 'Y' to continue termination without waiting, 'N' to wait for API users to close

Reason: The stack is in the process of orderly shutdown, but will let you terminate without waiting if you want.

Action: Enter Y to terminate the product without allowing orderly shutdown or enter N to wait for API users to close and orderly shutdown to proceed.

Chapter

34 T01ST*nnn* Server Telnet Messages

This chapter describes messages issued by Server Telnet. These messages include T01ST000 through T01ST999.

T01ST*nnn*

T01ST001E Termprof session parameters invalid | not found

The TERMPROF logmode entry (LOGMODENAME) specified for this application either:

- Contains invalid, conflicting, or incompatible session parameters
- The entry does not exist in the ACF/VTAM *xxxx*

The VTAMLIB data set name is the name of the logmode entry in error. Telnet session setup fails.

Action:

Do one of the following:

- Verify the session parameters to ensure they are syntactically correct and contain valid values
- Verify session parameters to ensure they comply with SNA protocols for LU2 sessions
- Verify that the logmode entry exists in the ACF/VTAM configuration data set (*xxxx*.VTAMLIB)

Note: Several IBM manuals contain the formats and rules for coding SNA session parameters, such as the IBM document *SNA: Reference Summary* and *SNA - Sessions Between Logical Units* include this information.

T01ST002E Unable to load *mod_name* for SIM3278/TCPIP

Reason: The SIMPCS interface module could not be loaded. The session proceeds as if SIMPCS does not exist.

Action: Check the log for the name of the module to be loaded and verify that the name was specified correctly in the SIMPCS parameter. Make sure the SIMPCS load module resides in either the JOBLIB/STEPLIB data set or in the system linklist concatenation.

T01ST003E Sim3278/TCPIP interface NOT ready

Reason: The SIMPCS interface loaded but would not honor the startup request. The session is terminated.

Action: Check the log for SIMPCS-issued messages that may explain the cause of the failure.

T01ST004I Sim3278/TCPIP interface ready for requests.

Reason: The SIMPCS interface is operational and ready to accept logon requests. The session proceeds normally.

Action: None. This is an Informational message.

T01ST005E Error in Sim3278/TCPIP *request* call.

Reason: A SIMPCS interface request failed. The session is terminated.

Action: Check the log for SIMPCS-issued messages that may explain the cause of the failure.

T01ST006I User *user_name* requested Sim3278/TCPIP logoff.

Reason: The Telnet client chose not to proceed with the SIMPCS session; in other words, the user entered L (logoff) to the SIMPCS request for the user's terminal type. The session proceeds as if SIMPCS does not exist, but in line mode only.

T01ST007E Error in Atput after Sim Terminit Rc=8.

Reason: A network write request (ATPUTM), issued after recovery from a previous TERMINIT call, failed. The session is terminated.

Action: Check the product log for messages that may explain the cause of the failure.

T01ST008E Error in Atget after Sim Terminit Rc=12.

Reason: A network solicit request (ATGET), issued after recovery from a previous TERMINIT call, failed. The session is terminated.

Action: Check the product log for messages that may explain the cause of the failure.

T01ST009E Error in Atput after Sim Terminit BUFL>0.

Reason: A network write request (ATPUT), issued after recovery from a previous TERMINIT call, failed. The session is terminated.

Action: Check the product log for messages that may explain the cause of the failure.

T01ST010I User *lu_name*/*IPaddr* signed on to Sim3278/TCPIP.

Reason: A client signed on to the Sim3278/TCPIP facility. The local ACF/VTAM network name assigned by this product on behalf of the client is represented by *lu_name*; the client's IP address is represented by *IPaddr*. The session setup proceeds normally.

T01ST011E *mod_name* Atopen failed, code = *value*

Reason: A Telnet open request failed. The module that made the call is represented by *mod_name*; the completion code value indicates the cause of the failure and can be one of the following:

X'04' Insufficient storage to allocate a TCCB; insufficient core space for the proper operation of the product.

Increase the region size for the job.

X'08' Release/disconnect from remote; indicates a network problem.

A TCPEEP of the connection might help determine the cause.

X'C' Bad parameter to ATOPN (no EPCB or APCB); indicates an internal error.

Check for any SVC dumps that may have been issued and contact Customer Support.

X'10' API error; indicates an internal error.

Check the log file for additional information and contact Customer Support.

X'14' Timeout opening the connection; normally indicates a problem communicating with the remote host.

A TCPEEP of the connection might help determine the cause.

X'18' API or transport provider termination; indicates API or transport provider termination.

X'1C' Bad parameter to ATOPN; indicates an internal error.

Check the log file for additional information and contact Customer Support.

T01ST013W ACB open failed for VLT *code1*, ACB error Rc=*code2* VLT RC=*code3* R15=*code4* R0=*code5*

Reason: The attempt to open a server Telnet session with a VTAM application failed to complete successfully. Register 15 and 0 values are options fields.

These are the possible error codes:

code1 VLT applid.

code2 ACB error code.

code3 VLT error code (in hexadecimal):

- 00 Operation completed successfully.
- 01 Normal with chained buffer.
- 02 Buffer purged by another entry.
- 03 I/O error in PUT operation.
- 04 I/O error in GET operation.
- 05 There are no applids left or passed applid is bad.
- 06 Error in REQSESS request.
- 07 Bad BIND image.
- 08 Error in SIGNAL.
- 09 Error in SETLOGON.
- 0A Error in OPENSEC.
- 0B Error in RESET.
- 0C Cannot OPEN ACB: Unknown reason.
- 0D Cannot OPEN ACB: Network seems misdefined.
- 0E Cannot OPEN ACB: Storage shortage.
- 0F Cannot OPEN ACB: VTAM shutting down.
- 10 Error in TERMSES.
- 11 VQE queueing error.
- 12 Contention resolution failed for GET.
- 14 OPEN timeout error (two minutes limit)
- 17 Data was received for an LU0 device that exceeds the buffer. LU0 does not support RU chaining. Increase the BUFFERSIZE on the TERMPROF statement for the TERMTYPE in question.

code4 Register 15 after OPEN request.

code5 Register 0 after OPEN request.

Action: If the ACB error code is nonzero, see the IBM manual *VTAM Reference Summary*.

T01ST014I RPL error: Req=*code1* R15=*code2* R0=*code3* Rtncd=*code4* Sense=*code5* Dest=*code6* VLT Rc=*code7*

Reason: A VLT attempted an RPL-based VTAM request but the request failed. Destination applid (*code6*), register 15 (*code2*), and 0 (*code3*) values are option fields.

code1 RPL request type.

code2 Register 15 after RPL request.

code3 Register 0 after RPL request.

code4 RPL return code.

code5 RPL sense.

code6 Destination applid.

code7 VLT return code (refer to VLT codes defined in message T01ST013W).

Action: Verify that VTAM is functioning properly. Use the *VTAM Reference Summary* manual for a reason for the RPL codes.

T01ST015I VLT BIND error: *bind* VLT Rc=*vlt*

Reason: A VTAM BIND was received but failed VLT's bind validity checking. The user session is terminated.

Action: Verify that VTAM is functioning properly. Use the IBM manual, *SNA Reference Summary*, for a description of a bind format. Refer to message T01ST013W for a reason for the VLT return codes.

T01ST016W VLT error: VLT Rc=*value* Lu=*lu_name* Ipaddr=*IP_addr*

Reason: A VLT error was detected.

Action: Verify that VTAM is functioning properly. Refer to message T01ST013W for a reason for the VLT return codes.

T01ST017I Connection refused: server Telnet port *port_num* not configured

Reason: This message indicates a configuration error. A SERVICE statement was defined for port *port_num* as AUTOlogon. However, no APPL statement was defined for the AUTOlogon application.

Action: Contact the system administrator. An APPL statement must be defined to match the SERVICE statement. The APPL NAME must match the SERVICE AUTO parameter and the APPL PORT parameter must match the SERVICE PORT.

T01ST022I Server Telnet User *user_id* Login Failed

Reason: An attempt to log user *user_id* in to the TCP/IP Telnet Server failed.

Action: None. This is an Informational message.

T01ST023W Call to module *mod_name* failed Rc=*hex_code*, Lu=*lu_name*, flags=*value*, ACB=*acb_code*, VLT Rc=*hex_vlt_code*

Reason: When the module *mod_name* is VTAMTERM, the rest of the message describes information returned from VTAMTERM to STELTSO during a VTAM application session startup. This message, along with related message T01ST024I, only appears when VLT ON was set under ACTEST. These messages help debug VTAM session startup problems occurring between STELTSO and VTAMTERM.

The *hex_vlt_code* variable is the hexadecimal VLT return code in message T01ST013W.

T01ST024I User *user_id* Lu=*lu_name*, local=*localIP:lport* remote=*remoteIP:rport*, VLT Rc=*code*

Reason: User *user_id* (*user_id* may be unknown) attempted to start a VTAM application session value1 with LU *lu_name*. The local host IP number and port follows the local field; the remote host IP number and port follows the remote field. This message only appears when VLT ON was set under ACTEST. These messages help debug VTAM session startup problems occurring between STELTSO and VTAMTERM.

T01ST025I VTelnet close: VTAM CLSDST timeout, code=*code*, Lu=*lu_name*

Reason: A problem was encountered with the Telnet session during termination. The session is in graceful termination and is attempting to complete pending VTAM Queued Elements (VQE). A VQE did not complete the operation within a specific time interval. The session then terminated without completing the VQE.

Action: Review previous error messages for additional information. This is not a serious problem and generally no action is required.

T01ST027I VTelnet close: net transient retcode, Lu=*lu_name*

Reason: A problem was encountered with the Telnet session and cannot be resolved. The Telnet session with the user is terminated.

Action: Verify that the remote host is available and retry Telnet function.

T01ST028I VTelnet close: 3278 bad negotiation, Lu=*lu_name*

Reason: Telnet attempts to negotiate a terminal as a 3278 and fails. The Telnet session with the user is terminated.

Action: Verify that VTAM is working properly and that the terminal is capable of 3270 support.

T01ST029I VTelnet close: idle timeout, Lu=*lu_name*

Reason: The Telnet session exceeds the time allowed for remaining idle. The Telnet session with the user is terminated.

Action: The terminal was allowed to remain idle for too long a period of time. The user must begin a new Telnet session.

T01ST030I VTelnet close: Operator stop, Lu=*lu_name*

Reason: The VTAM operator has canceled the user's Telnet session. The Telnet session with the user is terminated.

Action: The VTAM operator probably is performing maintenance. Wait a few minutes and attempt another Telnet session.

T01ST031I VTelnet close: VLT bad error, code=*code*, Lu=*lu_name*

Reason: The VTAM operator has canceled the user's Telnet session. The Telnet session with the user is terminated.

Action: The VTAM operator probably is performing maintenance. Wait a few minutes and attempt another Telnet session.

T01ST032I VTelnet close: 3278 user ended session, Lu=*lu_name*

Reason: The user explicitly requests termination of the Telnet session. Typically, sessions are terminated with PF-keys or with the END command. The Telnet session with the user is terminated.

Action: None. This is an Informational message.

T01ST033I VTelnet close: Atget error, code=*code*, Lu=*lu_name*

Reason: An attempt to read from Telnet is unsuccessful. The return code from ATGET displays (*code*). (The definitions for the hexadecimal VLT error codes are included in the description of T01ST013W.) The user Telnet session terminates.

Action: Verify that the remote host is functioning properly and retry Telnet session.

T01ST034I VTelnet close: Atput error, code=*code*, Lu=*lu_name*

Reason: An attempt to write to Telnet is unsuccessful. The return code from ATPUT is displayed (code).

Action: Verify that the remote host is functioning properly and retry the Telnet session.

T01ST035I VTelnet close: VTF display timeout, Lu=*lu_name*

Reason: VTAM message and error displays cannot be completed in a reasonable amount of time, causing the Telnet session to be found in error. The Telnet session with the user is terminated.

Action: Verify that the remote host is functioning properly and retry Telnet session.

T01ST036I VTelnet close: VFT OUT-Q timeout, Lu=*lu_name*

Reason: VTAM output queue cannot be processed within a reasonable amount of time, causing the Telnet session to be in error. The Telnet session with the user is terminated.

Action: Verify VTAM is functioning properly.

T01ST037I VTelnet close: 3278 EOB in binary, Lu=*lu_name*

Reason: While attempting to negotiate a 3278 terminal, a binary EOB control is encountered. The Telnet session with the user is terminated.

Action: Verify that the terminal is capable of supporting the desired Telnet mode of operation. Verify that the remote host is capable of supporting this mode of operation.

T01ST038I VTelnet close: 3278 bad command code, code=*code*, LU=*lu_name*

Reason: While attempting to negotiate a 3278, an unidentified control code was encountered. The Telnet session with the user is terminated.

Action: Verify that the terminal is capable of supporting 3270 protocols. Verify that the remote host is capable of supporting 3270 protocols.

T01ST039E VTelnet close: Atopen failed, code=*code*, Lu=*lu_name*

Reason: An error was encountered while attempting to establish a Telnet connection. The return code from the ATOPEN is displayed (*code*). See message T01ST011E for a detailed reason for the return (reason) code.

Action: Verify that the remote host is functioning properly.

T01ST040I VTelnet close: Losterm exit driven, Lu=*lu_name*

Reason: The losterm exit was driven for the terminal. The Telnet session and user terminate.

Action: None. This is an Informational message.

T01ST041I VTelnet close: VQE close timeout, Lu=*lu_name*

Reason: A problem was encountered with the Telnet session during termination. The session is in graceful termination and is attempting to complete pending VTAM Queued Elements (VQE). A VQE did not complete the operation within a specific time interval. The session then terminated without completing the VQE.

Action: Review previous error messages for additional information. This is not a serious problem and generally no action is required.

T01ST042I VTelnet close: VTAMappl post error, code=*post_code*, Lu=*lu_name*

Reason: The VTAMAPPL program in session with LU *lu_name* posted an error condition. The error code is given in *post_code*. The session is terminated.

Action: Recurrence of this message may indicate an internal error. Save all output and contact Customer Support.

T01ST043I Server Telnet User *user_id* Logged In

Reason: User *user_id* successfully logged into the TCP/IP Telnet Server.

Action: None. This is an Informational message.

T01ST044I Server Telnet User *user_id* Logged In

Reason: User *user_id* successfully logged in to the TCP/IP Telnet Server.

Action: None. This is an Informational message.

T01ST045I Failed, IMP down

Reason: A user attempted to access the system through VTAM. The user access attempt failed.

Action: The hardware interface to the product may be experiencing problems. Save all output from the job. Check the JES and T01LOG log files for related messages; check for any SVC dumps that may have been issued. Contact Customer Support.

T01ST046I Failed, Host *hostname* is down

Reason: A user attempted to access the specified host (*hostname*) through the system via VTAM. The user access attempt failed.

Action: Have your local network administrator verify status of the host.

T01ST047I APP logger is disabled

Reason: A user attempted to access the system through VTAM. The user access attempt failed. The APP logger function has been disabled. This message is indicative of other problems inside the APP.

Action: Save all output from the production job. Check the JES and T01LOG log files for related messages. Check for any SVC dumps that may have been issued. Contact Customer Support.

T01ST048I User protocol closed - *protocol*

Reason: A user attempted to access the system through VTAM. The user requested the use of a protocol that was not available at the time. The user access attempt failed.

Action: Have your local network administrator verify status of the specified protocol on the host. Save all output from the production job. Check the JES and T01LOG log files for related messages. Check for any SVC dumps that may have been issued. Contact Customer Support.

T01ST049I Unknown host number -> *address*

Reason: The user requested access to IP address *address* that was not known to the system. The user access attempt failed.

Action: If this was a typing error, retry the correct IP address. If the IP address is valid, have your local network administrator verify the specified address with DNR. Issue the following TSO command and examine its output:

```
DNRGET HOST BYVALUE address
```

Save all output from the production job. Check the JES, SYSPRINT, and T01LOG log files for related messages. Check for any SVC dumps that may have been issued.

T01ST050I Unknown internet host -> *hostname*

Reason: The user requested access to host (*hostname*) that was not known to the system. The user access attempt failed.

Action: If this was a typing error, retry the correct host name. If the host name is valid, have your local network administrator verify the host name with DNR. Issue the following TSO command and examine its output:

```
DNRGET HOST BYNAME hostname
```

Save all output from the production job. Check the JES, SYSPRINT, and T01LOG log files for related messages. Check for any SVC dumps that may have been issued.

T01ST051I Error unknown name -> *hostname*

Reason: The user requested access to host (*hostname*) that was not known to the system. The user access attempt failed.

Action: If this was a typing error, retry the correct host name. If the host name is valid, have your local network administrator verify the host name with DNR. Issue this TSO command and examine its output:

```
DNRGET HOST BYNAME hostname
```

Save all output from the production job. Check the JES, SYSPRINT, and T01LOG log files for related messages. Check for any SVC dumps that may have been issued.

T01ST052I Number out of range-> *number*

Reason: The user requested access to the system but attempted access with a bad value (*number*). The user access attempt failed.

Action: If this was a typing error, retry the correct host name. If the number is valid, contact your local network administrator. Save all output from the production job. Check the JES and T01LOG log files for related messages. Check for any SVC dumps that may have been issued.

T01ST053I outlog error -> *message*

Reason: The user requested access to the system. The message request was in an incorrect format. The specified message (*message*) defines the error type. The user access attempt failed.

Action: Retry the correct full syntax.

T01ST054I Syntax error -> *parameter*

Reason: The user requested access to the system. The message request was in an incorrect format. The specified parameter (*parameter*) was found in error. The user access attempt failed.

Action: Retry the correct full syntax.

T01ST055I Too many gateways

Reason: The user requested access to a system via a gateway list. There were too many gateways in the list. The user access attempt failed.

Action: Retry using a shorter list.

T01ST056I Refused by host -> *hostname*

Reason: The user requested access to the specified host (*hostname*), the host refused access. The user access attempt failed.

Action: Have your local network administrator verify that the specified host accepts logons from your node.

T01ST057I Outlog timeout

Reason: The user requested access to a system but a time out occurred before a connection could be made. The user access attempt failed.

Action: Repeat the attempted access.

T01ST058I *message*

Reason: The user requested access to a system but failed. The specified message (*message*) may be indicative of the failure reason.

Action: Have your local network administrator gather information on the problem.

T01ST059I Ambiguous name -> *hostname*

Reason: The user requested access to the specified host (*hostname*) that was ambiguous. The user access attempt failed.

Action: If this was a typing error, retry the correct host name. Have your local network administrator verify the host with DNR. Issue the following TSO command and examine its output:

```
DNRGET HOST BYNAME hostname
```

Modify host definitions as necessary.

T01ST060I AHHP illegal

Reason: The user access attempt failed.

Action: Have your local network administrator save all output from the production job. Check the JES and T01LOG log files for related messages. Check for any SVC dumps that may have been issued. Contact Customer Support.

T01ST061I Invalid internal module name parm -> *parameter*

Reason: The user attempted access to a module (*parameter*) but this module is not defined in the product as accessible. The user access attempt failed.

Action: If this was a typing error, retry using the correct syntax.

T01ST062I Syntax error, missing delimiter -> *parameter*

Reason: The user requested access to host via a gateway list with a missing delimiter. The user access attempt failed.

Action: If this was a typing error, retry using the correct gateway list format.

T01ST063I Syntax error, bad port number syntax -> *parameter*

Reason: The user requested access to a host on port *parameter*. The parameter was not a decimal number. The user access attempt failed.

Action: If this was a typing error, retry using the correct port number.

T01ST064I Syntax error, unknown keyword parameter -> *parameter*

Reason: The user requested access using keywords. The specified keyword (*parameter*) is not supported. The user access attempt failed.

Action: If this was a typing error, retry using the correct syntax.

T01ST065I Syntax error, PID parm is not decimal -> *parameter*

Reason: The user requested access using the PID keyword. The PID keyword (*parameter*) must be decimal. The user access attempt failed.

Action: If this was a typing error, retry using the correct syntax.

T01ST066I Syntax error, unknown HLP parameter -> *parameter*

Reason: The user requested access using the HLP keyword but used an unknown HLP (*parameter*). The user access attempt failed.

Action: If this was a typing error, retry using the correct HLP entry.

T01ST067I Syntax error, unsupported HLP protocol -> *parameter*

Reason: The user requested access using the HLP keyword but attempted an unsupported protocol. The user used an unknown HLP (*parameter*). The user access attempt failed.

Action: If this was a typing error,retry using the correct HLP protocol.

T01ST068I Syntax error, IP parm is not decimal -> *parameter*

Reason: The user requested access using the IP keyword. The IP keyword (*parameter*) must be decimal. The user access attempt failed.

Action: If this was a typing error, retry using the correct syntax.

T01ST069I Syntax error, MYNET parm is not decimal -> *parameter*

Reason: The user requested access using the MYNET keyword. The MYNET keyword (*parameter*) must be decimal. The user access attempt failed.

Action: If this was a typing error, retry using the correct syntax.

T01ST070I Syntax error, Bad trailing characters -> *parameter*

Reason: The user attempted access to a host but the specified characters (*parameter*) are in error. The user access attempt failed.

Action: If this was a typing error, retry using the correct syntax.

T01ST071I Syntax error, unknown host specified -> *hostname*

Reason: The user requested access to the specified host (*hostname*) that was not resolved. The user access attempt failed.

Action: If this was a typing error, retry using the correct host name. Have your local network administrator verify the host with DNR. Issue the following TSO command and examine its output:

```
DNRGET HOST BYNAME hostname
```

Modify host definitions as necessary.

T01ST072I Total gateways exceeded maximum

Reason: Access was requested via a gateway list. The product only supports access via a gateway list up to five gateways only. The user access attempt failed.

Action: Re-attempt access with fewer gateways.

T01ST073I Total source routes exceeded

Reason: The user requested access via source routes. The product only supports access via one source route. The user access attempt failed.

Action: Re-attempt access with one source route.

T01ST074I Syntax error, bad host specified -> *hostname*

Reason: The user requested access to the specified host (*hostname*) that was not resolved. The user access attempt failed.

Action: If this was a typing error, the user should re-attempt using the correct host name. Have your local network administrator verify the host with DNR. Issue this TSO command and examine its output:

```
DNRGET HOST BYNAME hostname
```

Modify host definitions as necessary.

T01ST075I APP option specified, but no application name entered

Reason: A user invoked one of the client programs (TCPEEP, FTP, FTP2, or Telnet) using the APP invocation option but no VTAM application name was entered. Client programs TCPEEP, FTP, or FTP2 then terminate. Client program Telnet uses the default VTAM application name.

Action: Refer to the *User Guide* for the syntax of the invocation option on the FTP and FTP2 commands; see the *Administrator* or *System Management Guide* for information about the ACPEEP command. A user can issue the APPLID command under Client Telnet to change the VTAM application name.

T01ST076I APP name is longer than 8 characters *vtam_app_name*

Reason: A user invoked one of the client programs (TCPEEP, FTP, FTP2, or Telnet) using the APP invocation option, but the VTAM application name (*vtam_app_name*) was longer than eight characters. Client programs TCPEEP, FTP, or FTP2 then terminate. Client program Telnet uses the default VTAM application name.

Action: Refer to the *User Guide* for the syntax of the invocation option on the FTP and FTP2 commands; see the *Administrator* or *System Management Guide* for information about the TCPEEP command. A user can use the APPLID command under Client Telnet to change the VTAM application name.

T01ST077I APP and SYS options specified, SYS option ignored

Reason: A user invoked a client program (TCPEEP, FTP, FTP2, or Telnet) using both the APP and SYS invocation options. The client program ignored the SYS option and used the APP option.

Action: Refer to the *User Guide* for the syntax of the invocation option on the FTP and FTP2 commands; see the *Administrator* or *System Management Guide* for information about the TCPEEP command.

T01ST078I Attempting logon to the application

Reason: The service name entered at the Server Telnet prompt is not defined in the APPCFG*xx* member. If a TELNET DEFAULT statement was coded, the name entered is used as a VTAM APPLLID. A T01ST014 message follows if the request fails.

Action: If this is not a VTAM APPLID, check the spelling of the TELNET command. Also, verify that the service name is defined in your APPCFG*xx* member.

T01ST079I Connection Refused: Server Telnet port *ppp* not configured

Reason: This message indicates a configuration error. A SERVICE statement was defined for port *ppp* as AUTOlogon. However, no APPL statement was defined for the AUTOlogon application.

Action: Contact the system administrator. An APPL statement must be defined that matches the SERVICE statement: the APPL NAME must match the SERVICE AUTO parameter and the APPL PORT parameter must match the SERVICE PORT.

T01ST081E NO LU AVAILABLE IN LUPOOL:*lupool_name* FOR USER:*userid* IP=*ip_addr*

Reason: The LU pool facility was attempting to acquire an LU name for a user from a remote host. All LU names identified in the pool are already allocated to other users. The request to access the VTAM application is denied and the user is returned to the Telnet prompt.

lupool_name The pool from which the LU pool facility was attempting to acquire an LU name.

userid The local logon ID used by the remote user to identify himself or herself to the local security system.

ip_addr The IP address of the remote host.

Action: You can retry the TELNET command to acquire any deallocated LU name. The system administrator can add additional LU names to the LU pool identified in the message.

T01ST082E NO APPLICABLE LURULE FOR LU ASSIGNMENT FOR USER:*userid* IP=*ip_addr*

Reason: The LU pool facility was attempting to acquire an LU name for a user from a remote host. The remote user could not match the criteria for any LURULE rule. The request to access the VTAM application is denied and the user is returned to the Telnet prompt.

Syntax Description

userid The local logon ID used by the remote user to identify himself or herself to the local security system.

ip_addr The IP address of the remote host.

Action: Contact the system administrator and request a rule to access host applications.

T01ST083E LUPOOL FACILITY NOT ACTIVE DUE TO INITIALIZATION ERROR

Reason: During initialization of the APP task group, an error was encountered during LU pool processing of the LU pool configuration file APPLUP*xx*. No users are allowed access to host applications until the problem with the LU pool configuration is resolved. Other messages are displayed prior to this message describing the errors encountered while processing the configuration file.

Action: Contact the system administrator to request correction of the problem. The system administrator must review the operator console log for messages describing the reason for initialization failure. After the problem is resolved, the APP REFRESH LUPARM(*member_name*) must be issued from the operator console.

T01ST084E Mail message truncated due to bad record length found in JES dataset: *jes_dataset_name*

Reason: Program SPOOL#4 reads files off the JES spool destination defined on the REMOTE parameter in the SMTP statement in APPCFG*xx*. While attempting to read a record it discovered that the record length returned was larger than the LRECL of the JES data set. This is a permanent error preventing the full delivery of the mail message.

SPOOL#4 places error message T01ST084E into the truncated mail output data set in an attempt to inform the user that there was a problem in the delivery of this message.

Messages T01S4004 and T01S4005 are issued in T01LOG when this error occurs. Message T01S4005 has the same JES data set name as in this message. Check for any related SVC dumps that may have been issued.

Action: Look for defects in the application that places data sets on the JES spool. Specifically, look for instances where it could write blocks of data in which a record may be larger than the LRECL of the data set.

T01ST085I ENTER PASSWORD

Reason: A TSO command is prompting for a password.

Action: None. This is an Informational message.

T01ST087I Telnet Connection Lu=*luname* Local=*local_addr* Remote=*remote_addr*

Reason: This message is written to the T01LOG when a Server Telnet connection is established. The LU (*luname*), local host IP address (*local_addr*) and the remote host IP address (*remote_addr*) are given.

Action: None. This is an Informational message.

T01ST088I ACB CLOSE FAILED - LU=*lu_name*, ACBFLAGS=*flags*; Will retry.

Reason: An attempt was made to close the session owned by *lu_name*, but VTAM could not complete the operation. The operation is attempted every 30 seconds until the close is successful.

flags gives the two bytes of flags from the ACB: ACBOFLGS & ACBERFLG. If the flags have the value '1214' for instance, this indicates that VTAM has temporarily run out of virtual memory for control blocks.

Action: The session should recover and close successfully. If this does not happen, keep the T01LOG files and the Console log and contact Customer Support.

Chapter 35

T01TC*nnn* Transport Layer Messages

This chapter describes messages issued by the Transport Layer. These messages include T01TC000 through T01TC099.

T01TC*nnn*

T01TC000W TCP packet too short.

Reason: A TCP packet arrived that was shorter than the minimum TCP header. A dump of the packet follows.

Action: Examine the remote host for this session.

T01TC001W TCP invalid offset *offset,* header length *header* . session *localIP:lport remoteIP:rport*

Reason: A TCP datagram arrived and the offset field was invalid.

offset	Offset from the packet.
header	Header length.
localIP	Local IP address.
lport	Local port number.
remoteIP	Remote IP address.
rport	Remote port number.

Action: Examine the remote host for this session.

T01TC002W TCP invalid checksum *checksum*. session *localIP:lport remoteIP:rport*

Reason: A TCP datagram arrived and the checksum field was invalid.

checksum	Checksum computer.
localIP	Local IP address.
lport	Local port number.
remoteIP	Remote IP address.
rport	Remote port number.

Action: Examine the remote host for this session.

T01TC004W TCP connection queue overflow - session *localIP:lport remoteIP:rport*

Reason: A TCP SYN packet arrived and too many connections were already pending for this port.

localIP	Local IP address.
lport	Local port number.
remoteIP	Remote IP address.
rport	Remote port number.

Action: Increase the pending connection count for the application supporting this port.

T01TC006I TCP connection opened - session *localIP:lport remoteIP:rport*

Reason: A TCP SYN packet arrived and the connection was opened.

localIP	Local IP address.
lport	Local port number.
remoteIP	Remote IP address.
rport	Remote port number.

Action: None. This is an Informational message.

T01TC007I TCP connection rejected - session *localIP:lport remoteIP:rport*

Reason: A TCP SYN packet arrived and the connection was rejected.

localIP	Local IP address.
lport	Local port number.
remoteIP	Remote IP address.
rport	Remote port number.

Action: Examine the application supporting this port.

T01TC008I TCP connection reused - session *localIP:lport remoteIP:rport*

Reason: A TCP SYN packet arrived and the connection was a duplicate of one in Time_Wait state.

localIP	Local IP address.
lport	Local port number.
remoteIP	Remote IP address.
rport	Remote port number.

T01TC009I TCP connection refused - session *localIP:lport remoteIP:rport*

Reason: A TCP RST packet arrived while trying to establish a connection, indicating the connection was refused by the remote server.

localIP	Local IP address.
lport	Local port number.
remoteIP	Remote IP address.
rport	Remote port number.

Action: Examine the remote host supporting this port.

T01TC010I TCP connection reset by remote - session *localIP:lport remoteIP:rport*

Reason: A TCP packet arrived that specified an ACK number greater than the send sequence number, indicating data was ACKed by the remote host before the data was sent.

localIP	Local IP address.
lport	Local port number.
remoteIP	Remote IP address.
rport	Remote port number.

Action: Examine the remote host supporting this connection

T01TC011I TCP connection ACKed too much - session *localIP:lport remoteIP:rport*

Reason: A TCP packet arrived which specified an ACK number greater than the connection's send sequence number, indicating data was ACKed by the remote host before the data was sent.

localIP	Local IP address.
lport	Local port number.
remoteIP	Remote IP address.
rport	Remote port number.

Action: Examine the remote host supporting this connection

T01TC012W TCP received invalid options - session *localIP:lport remoteIP:rport*, options = *options*

Reason: A TCP packet arrived containing invalid options. The options are dumped.

localIP	Local IP address.
lport	Local port number.
remoteIP	Remote IP address.
rport	Remote port number.
options	Options dumped.

Action: Examine the remote host supporting this connection.

T01TC013D TCP datagram follows:

Reason: An error was encountered in an inbound TCP datagram. The datagram is dumped following the message.

Action: This diagnostic message is preceded by an error or warning message. Follow the action for that message.

T01TC014W TCP received invalid MSS - session <*localIP*>:<*lport*> <*remoteIP*>:<*rport*>, size = <*size*> set to 536

Reason: A TCP packet arrived that contained a MSS of less than 536, which is the minimum size that a TCP/IP stack must support. The maximum segment size used for that session is 536.

localIP	Local IP address
lport	Local port number
remoteIP	Remote IP address
rport	Remote port number
size	MSS value in SYN packet options

Action: Examine the remote host supporting this connection.

T01TC050W *proto* datagram *localIP:lport remoteIP:rport* unable to route

Reason: A RAW datagram was sent for which a route could not be obtained.

proto	Protocol.
localIP	Local IP address.
lport	Local port number.
remoteIP	Remote IP address.
rport	Remote port number.

Action: Examine the network for routing problems

T01TC060W UDP invalid checksum *checksum*. session *localIP:lport remoteIP:rport*

Reason: A UDP datagram arrived and the checksum field was invalid.

checksum	Checksum computed.
localIP	Local IP address.
lport	Local port number.
remoteIP	Remote IP address.
rport	Remote port number.

.

Action: Examine the remote host for this session

T01TC061W UDP invalid length *length*. session *localIP:lport remoteIP:rport*

Reason: A UDP datagram arrived and the length field was invalid.

length	Length field received.
localIP	Local IP address.
lport	Local port number.
remoteIP	Remote IP address.
rport	Remote port number.

Action: Examine the remote host for this session.

T01TC062D UDP datagram follows:

Reason: An error was encountered in an inbound UDP datagram. The datagram is dumped following the message.

Action: This diagnostic message is preceded by an error or warning message. Follow the action for that message.

remoteIP	Remote IP address.
rport	Remote port number.

Action: Examine the network for routing problems.

T01TC081I TCP connection closed. session *localIP:lport remoteIP:rport*

Reason: The TCP connection was closed. This message may be followed by messages T01TC082S and T01TC083S, depending on your message logging configuration.

localIP	Local IP address.
lport	Local port number.
remoteIP	Remote IP address.
rport	Remote port number.

Action: None. This is an Informational message.

T01TC082S In *num_rcvd* pkts *num_bytes* bytes *num_acks* acks *num_err_pkts* errors *dup_pkts* discards *dup_bytes* bytes

Reason: A TCP connection was closed.

num_rcvd	Number of packets received.
num_bytes	Number of data bytes received.
num_acks	Number of ACK packets received.
num_err_pkts	Number of packets discarded for errors.
dup_pkts	Number of duplicate packets.
dup_bytes	Number of duplicate bytes.

T01TC083S Out *pkts_sent* pkts *bytes_sent* bytes *acks_sent* acks *pkts_disc* retransmits *pkts_retran* discards *bytes_retran* bytes

Reason: A TCP connection was closed.

pkts_sent	Number of packets sent.
bytes_sent	Number of data bytes sent.
acks_disc	Number of Ack packets sent.
pkts_disc	Number of packets discarded for errors.
pkts_retran	Number of retransmitted packets.
bytes_retran	Number of retransmitted bytes.

T01TC084I UDP connection closed. session *localIP:lport remoteIP:rport*

Reason: A UDP connection was closed.

localIP	Local IP address.
lport	Local port number.
remoteIP	Remote IP address.
rport	Remote port number.

Action: None. This is an Informational message.

T01TC085I RAW connection closed for protocol *proto*

Reason: A RAW connection has been closed. *proto* is the protocol name.

Action: None. This is an Informational message.

T01TC090W TCP datagram *localIP:lport remoteIP:rport* unable to route

Reason: A TCP datagram was sent for which a route could not be obtained.

localIP	Local IP address.
lport	Local port number.
remoteIP	Remote IP address.
rport	Remote port number.

Action: Examine the network for routing problems.

T01TC091T TCP connection reset locally - *localIP:lport remoteIP:rport*

Reason: A TCP connection was reset.

localIP	Local IP address.
lport	Local port number.
remoteIP	Remote IP address.
rport	Remote port number.

Action: Examine the application supporting this port on the remote host. If T01IP043I accompanies this message, verify that your software is configured for the correct port. If it's correct, then check for a listening application.

T01TC100W TCP keepalive processing, MBUF shortage

Reason: A TCP connection was to be probed with keepalive packets but an MBUF could not be allocated for a packet

Action: Examine the application supporting this port on the remote host.

T01TC101I TCP connection keepalives exceeded *localIP:lport remoteIP:rport*

Reason: A TCP connection was probed with keepalive packets, and did not respond.

localIP	Local IP address.
lport	Local port number.
remoteIP	Remote IP address.
rport	Remote port number.

Action: Examine the application supporting this port on the remote host.

T01TC102I TCP connection timed out *localIP:lport remoteIP:rport*

Reason: TCP was unable to establish a connection with the remote host in the specified time.

localIP	Local IP address.
lport	Local port number.
remoteIP	Remote IP address.
rport	Remote port number.

Action: Examine the remote host to be sure the application is ready to accept connections.

T01TC110W TCP logic error *localIP:lport remoteIP:rport*

Reason: A TCP connection was to be probed with persist packets, and the retransmit timer was still set.

localIP	Local IP address.
lport	Local port number.
remoteIP	Remote IP address.
rport	Remote port number.

Action: Contact Customer Support.

T01TC120I TCP retransmits exceeded, *localIP:lport remoteIP:rport*

Reason: TCP retransmitted data to a remote host, but the data was not acknowledged. The specified maximum number of retransmits has occurred. The connection is reset.

localIP	Local IP address.
lport	Local port number.
remoteIP	Remote IP address.
rport	Remote port number.

Action: Examine the remote host to see if it is operational and that the application is still active.

T01TC900D TCP *time event*: *seq1*:*seq2*(*len*) *ack* (win=*win*)

Reason: A debugging record has been written for a TCP connection.

time	Timestamp in units of 0.004096 seconds.
event	TCP event causing this record.
seq1	TCP sequence number of the beginning of the packet.
seq2	TCP sequence number of the end of the packet.
len	Length of the packet data.
ack	Acknowledgement number of the packet.
win	Window size of the packet.

T01TC901D TCP <*old*> -> *new*

Reason: A debugging record was written for a state change.

old	Old TCP state.
new	New TCP state.

T01TC902D TCP rcv_nxt *seq*, rcv_wnd *wind* lport *lip*:*lport* rport *rip*:*rport*

Reason: A TCP degugging record was written for receive information.

seq	Timestamp in units of 1004096 seconds.
wind	TCP event causing this record.
lip	TCP sequence number of the beginning of the packet.
lport	TCP sequence number of the end of the packet.
rip	Length of the packet data.
rport	Acknowledgement number of the packet.

T01TC903D TCP snd_una *suna*, snd_nxt *snxt*, snd_max *smax*, snd_urg *urg*

Reason: A TCP debugging record was written for Send information.

suna	Unacknowledged send sequence number.
snxt	Next sequence number to be sent.
smax	Maximum sequence number ever sent.
urg	Current Send Urgent sequence number.

T01TC904D TCP snd_wl1 *windu*, snd_wl2 *winda*, snd_wnd *wind*, rcv_urg *urg*

Reason: A TCP debugging record was written for Send Window information

windu	Window update sequence number.
winda	Window update acknowledgement number.
wind	Send window size.
urg	Current Receive Urgent sequence number.

T01TC905D TCP Rexmit=*rtime*, (t_rtxshift=*count*), Keep=*keeptime* Persist=*ptime*, TimeWait=*tw*

Reason: A debugging record has been written for timer information.

rtime	Retransmit timer value.
count	Current retransmit counter.
keeptime	Keepalive timer value.
ptime	Persist timer value.
tw	Time Wait timer value.

T01TC906D TCP DelAck=*dtime*, Linger=*ltime* Xfree=*free*

Reason: A debugging record was written for timer information.

dtime	Delay ack timer value.
ltime	Linger timer value.
free	Free timer value.

T01TC907D TCP t_rtt=*rrt*, t_srtt=*srtt*, t_rtvar=*var*, t_rtxcur=*cur*, t_rtseq=*seq*

Reason: A debugging record was written for round trip information.

rrt	Round trip time value.
srtt	Scaled round trip time.
var	Round trip timer variation.
cur	Current retransmit timer value.
seq	Round trip sequence number.

T01TC908D TCP *flags*

Reason: A debugging record was written for flag information. *flags* indicates the flagbits that were in the TCP header.

Chapter

36 T01TE*nnn* Telnet Server Messages

This section describes messages issued by the TN3270E Telnet server. These messages include T01TE000 through T01TE099.

T01TE*nnn*

T01TE001E Session *sess_num* Plu *pluname* Slu *sluname function* error *error_info* in *module*

Reason: The TN3270E server session identified by *sess_num* encountered an error for VTAM function *function*. *error_info* is the sense information.

Action: Contact Customer Support

T01TE002D Session *sess_num* Plu *pluname* Slu *sluname* FMH-Data received, not negotiated

Reason: The TN3270E server session identified by *sess_num* received FMH-Data from the client but the client did not negotiate FMH support.

Action: This may indicate a problem with the TN3270(e) client. If the problem persists, contact Customer Support.

T01TE003D Session *sess_num* Plu *pluname* Slu *sluname* RESPONSE-FLAG set, not negotiated

Reason: The TN3270E server session identified by *sess_num* received RESPONSE-FLAG from the client, but the client did not negotiate RESPONSE support.

Action: This can indicate a problem with the TN3270(e) client. If the problem persists, contact Customer Support.

T01TE004I Session *sess_num* Plu *pluname* Slu *sluname* Intervention Required

Reason: The TN3270E server session identified by *sess_num* received "Intervention Required" from the identified printer.

Action: Determine the action needed at the printer.

T01TE005D Session *sess_num* Plu *pluname* Slu *sluname* Negative Response, Sense *sense*

Reason: The TN3270E server session identified by *sess_num* received a negative response from the client.

Action: If this is causing an error (it is not an error by itself), it may indicate a problem with the TN3270(e) client. If the problem persists, contact Customer Support.

T01TE006I Session *sess_num* Plu *pluname* Slu *sluname* Intervention Cleared

Reason: The TN3270E server session identified by *sess_num* received "Intervention Cleared" from the identified printer.

Action: None. This is an Informational message.

T01TE007I Session *sess_num* - close timed out

Reason: The TN3270E server session identified by *sess_num* did not close the connection in a timely manner.

Action: Contact Customer Support.

T01TE008W Session *sess_num* - IDLE timeout

Reason: The TN3270E server session identified by *sess_num* is closing the connection due to IDLE timeout.

Action: The IDLE timeout can be turned off or set to a specific duration, using the IDLE parameter of the SERVICE statement in the APPCFG*xx* configuration member.

T01TE100E Session *sess_num* Plu *pluname* Slu *sluname function* in *module*

Reason: The TN3270E server session identified by *sess_num* encountered a VTAM error for *function.* This message should be accompanied by message T01TE101E.

Action: Contact Customer Support.

T01TE101E R15=*nnnnnnnn* R00=*nnnnnnnn rtncd_fdbk2 sense* REQ *req*

Reason: The TN3270E server session identified by *sess_num* received an error for the VTAM function specified in message T01TE100E.

Action: Contact Customer Support.

T01TE102E BIND error *sense info* Image: *bind_image*

Reason: The TN3270E server session identified by *sess_num* encountered a BIND error.

Action: Check the LOGMODE names for the device type.

T01TE110E Session *sess_num* Slu *sluname function* error: R15=*nnnnnnnn* ACBERFLG=*xx*

Reason: The TN3270E server session identified by *sess_num* received an error during an OPEN or CLOSE of the ACB.

Action: Display the VTAM status of the LU. Verify the VTAM configuration for the applications. Verify the LU pool configuration in the stack.

T01TE111E GENCB error for EXLST - error *error_info*

Reason: The TN3270E Server encountered an error preparing the VTAM environment.

Action: Contact Customer Support.

T01TE120E Session *sess_num* Plu *pluname* Slu *sluname function* error *error_info* in *module*

Reason: The TN3270E server session identified by *sess_num* encountered a VTAM error for *function*

Action: Contact Customer Support.

T01TE121I Session *sess_num* Slu *sluname* Local *ip_address:port* Remote *ip_address:port*

Reason: The TN3270E server session identified by *sess_num* has acquired an SLU name from the LUPOOL and has opened the ACB.

Action: None. This message provides a SLU-name-to-IP-address map.

T01TE122I Session *sess_num* Appl *applid* Plu *pluname* Slu *sluname* Logmode *logmode*

Reason: The TN3270E server session identified by *sess_num* initiated a session with the requested application.

Action: None. This is an Informational message.

T01TE216D Session *sess_num* Data received during negotiation:

Reason: The TN3270E server session identified by *sess_num* received non-negotiation data during Telnet negotiation. The data is dumped to the message log. Processing continues.

Action: This may indicate a problem with the TN3270(e) client. If the problem persists, contact Customer Support.

T01TE217E Session *sess_num* Invalid Tn3270e negotiation:

Reason: The TN3270E server session identified by *sess_num* received an invalid TN3270E negotiation command. The input is dumped following the message. The session is terminated.

Action: This may indicate a problem with the TN3270(e) client. If the problem persists, contact Customer Support.

T01TE218E Session *sess_num* Invalid Tn3270 datastream:

Reason: The TN3270E server session identified by *sess_num* received an invalid TN3270 datastream in fullscreen mode. The input is dumped following the message. The session is terminated.

Action: This may indicate a problem with the TN3270(e) client. If the problem persists, contact Customer Support.

T01TE219E Session *sess_num* Invalid Tn3270e header:

Reason: The TN3270E server session identified by *sess_num* received an invalid TN3270E header in a TN3270E message. The input is dumped following the message. The session is terminated.

Action: This may indicate a problem with the TN3270(e) client. If the problem persists, contact Customer Support.

T01TE220E Session *sess_num* Error in translate table *trans_table*

Reason: The TN3270E server session identified by *sess_num* recognized errors in the translate table identified by *trans_table*.

Action: If the translate table identified by *trans_table* is provided by the installation, examine the translate table to ensure that it is valid. If it is one of the default tables provided by the product, or if the problem persists, contact Customer Support.

T01TE300I *task_id* socket (*sd*) listening on *localip:port*

task_id—Task identifier.

sd—Socket descriptor.

localip:port—Local IP address and port.

Reason: The task is listening on specified IP address and port.

Action: None. This is an Informational message.

T01TE301I *task_id* Task started – *cb_addr*

task_id—Task identifier.

cb_addr—Address of the control block for the task.

Reason: A task to support the TN3270E/SSL Server has been started.

Action: None. This is an Informational message.

T01TE302I *task_id* returning

id—Unique identifier associate with the client session.

Reason: The task is returning.

Action: None. This is an Informational message.

T01TE303I *task_id* accept(listen_sd) client(client_sd) remoteip:port on port localport

task_id – task identifier.

listen_sd—Socket descriptor of the listener.

client_sd—Socket descriptor of the client connection.

remoteip:port—IP address and port of remote client.

localport—Local port the client has connected to.

Reason: A remote client has connected to the listener.

Action: None. This is an Informational message.

T01TE304I (Text varies, see below)

task_id `listen socket(`*sd*`) closed,error` *errno errno2*

task_id—Task identifier.

sd—Socket descriptor.

errno errno2—Error information from the socket function.

Reason: The listen socket has been closed. If an error occurred closing the socket, error information is additionally displayed.

Action: None. This is an Informational message.

T01TE305E (Text varies, see below)

task_id Port(*port*) socket() error *errno errno2*

task_id Port(*port*) ioctl(*sd*) SIOCSETRTTD(*provider*) error *errno errno2*

task_id Port(*port*) setibmsockopt(*sd*) EioIfNewTP error *errno errno2*

task_id Port(*port*) setsockopt(*sd*) REUSEADDR error *errno errno2*

task_id Port(*port*) ioctl(*sd*) FIONBIO error *errno errno2*

task_id Port(*port*) bind(*sd*) error *errno errno2*

task_id Port(*port*) listen(*sd*) error *errno errno2*

task_id—Task identifier.

port—Listening port.

sd—Socket descriptor.

errno errno2—Error information from the socket function.

Reason: An error occurred initializing a listening port Once a minute there will be an attempt to initialize the port.

If the T04STSSL server is running in the same address space as the Unicenter TCPaccess stack and that stack is the only stack the server is listening on and it is being shutdown with either **P** or **P CLEAR,** then ignore this message (it is a timing related issue). If the server is running in a different address space than the stack and that stack is the only stack the server is listening on and it is being shutdown with either **P** or **P CLEAR,** then first terminate the server to avoid receiving the messages. Otherwise the message indicates an error occurred and the server is no longer listening on the port (it will periodically attempt to re-initialize the port); the stack(s) that are servicing that port may have shutdown or failed (the appropriate recovery action should be taken on each such stack).

Action: Activate the necessary resources. (Stack, IP address, and so on.)

T01TE306E *task_id* Pass-socket(*sd*) error *rc*

task_id—Task identifier.

sd—Socket descriptor of the socket being passed.

rc—Return code.

Reason: An error occurred passing a socket to a sub-task. The connection is closed.

Action: Record return codes less than 100.

Return code 100 is set when there is not enough virtual storage.

Return code 101 is set when all available sub-tasks are at the maximum session limit. Configure the server with an additional worker TCB.

Return code 102 is set when there is an error allocating an ITIME block.

Return code 103 is set when there is an error processing the translate table.

T01TE307E (Text varies, see below)

task_id Listen Info Array malloc error

task_id master_read malloc error

task_id master_excpt malloc error

task_id read_fd malloc error

task_id excpt_fd malloc error

task_id error_fd malloc error

task_id sdArray malloc error

task_id malloc error obtaining gsk_init

task_id—Task identifier

Reason: An error occurred obtaining storage for a structure or array. The specified task terminates.

Action: Determine storage requirements.

T01TE308E *task_id* No listen blocks

task_id—Task identifier.

Reason: The listen module determined there were no PORTs to listen on. The listener will terminate.

Action: Contact support.

T01TE309E *task_id* Unable to locate ListenBlock for client(*sd*)

task_id—Task identifier.

sd—Socket descriptor of client.

Reason: The listen module could not locate the associated PORT block. The client connection is closed.

Action: Contact support.

T01TE310E *task_id* Task attach error *rc*

task_id—Task identifier.

rc—Return code.

Reason: An error occurred attaching a sub-task.

Action: Record *rc*.

T01TE311E *task_id* Tsess(*id*) takesocket(*sd*) error *errno errno2*, *remoteip*:*port*

task_id—Task identifier.

id—Unique identifier associate with the client session.

sd—Socket descriptor of the passed socket.

errno errno2—Error information from the socket function.

remoteip:*port*—IP address and port of remote client.

Reason: An error occurred taking a passed socket. The client connection is terminated.

Action: Record *errno* and *errno2*.

T01TE312I *task_id* Tsess(*id*) started(*sd*) *remoteip*:*port*

task_id—Task identifier.

id—Unique identifier associate with the client session.

sd—Socket descriptor of the socket now used to communicate with the client.

remoteip:*port*—IP address and port of remote client.

Reason: The client has established a connection with an assigned sub-task.

Action: None. This is an Informational message.

T01TE313E (Text varies, see below)

task_id Tsess(*id*) FIONBIO error *errno errno2*

task_id Session(*id*) OOBINLINE error *errno errno2*

task_id Session(*id*) SO_SNDBUF error *errno errno2*

task_id Session(*id*) SO_RCVBUF error *errno errno2*

task_id—Task identifier.

id—Unique identifier associate with the client session.

errno errno2—Error information from the socket function.

Reason: An error occurred while setting socket options. The client connection is terminated.

Action: Record *errno* and *errno2*.

T01TE314E *task_id* Session(*id*) gsk_secure_soc_init error *rc*

task_id—Task identifier.

id—Unique identifier associate with the client session.

rc—Return code.

Reason: An error occurred initializing the socket for SSL. The client connection is terminated.

Action: Record *rc*. Ensure the client software enabled SSL for the connection.

Refer to the appendix "error *rc* Mapping" for information on how to map the *string* and the SSL function from this message to determine the error as described in the IBM manual *System Secure Socket Layer Programming Guide and Reference*.

T01TE315I *task_id* Tsess(*id*) close

task_id—Task identifier.

id—Unique identifier associate with the client session.

Reason: The client connection has been closed.

Action: None. This is an Informational message.

T01TE316E *task_id* Session(*id*) gsk_initialize error *rc* for *keyring*

task_id — Task identifier.

id — Unique identifier associate with the client session.

rc — Return code.

keyring – SAF keyring or HFS file.

Reason: An error occurred initializing SSL for the sub-task. The specified task terminates.

Action: Record *rc*.

T01TE317E (Text varies, see below)

task_id Session(*id*) selectex() error *errno errno2*

task_id — SelRcvry() error *errno errno2*

task_id — Task identifier.

id — Unique identifier associate with the client session.

errno errno2 — error information from the socket function.

Reason: An error occurred during a selectex() socket function. The specified task terminates.

Action: Record *errno* and *errno2*.

T01TE318E *task_id* accept(*sd*) error *errno errno2* task_id accept(sd) NewTP errno errno2 task_id givesocket(sd) error errno errno2

task_id — Task identifier.

sd — Socket descriptor.

Reason: The listener received an error attempting to accept the connection or pass the connection to a sub-task. The client connection is terminated. For "NewTP" the CINET environment has changed, the listening socket will be closed and re-opened.

In the case that message T01TE305E follows this message, see the description for T01TE305E .

Action: If a problem exists, record *errno* and *errno2*.

T01TE319E *task_id* getclientid() error *errno errno2*

task_id—Task identifier.

errno errno2—error information from the socket function.

Reason: The task received an error from getlientid(). The specified task terminates.

Action: Record *errno* and *errno2*.

T01TE320I *task_id* Domain(*domain*) Provider(*provider*)

task_id—Task identifier.

domain—Protocol family (default 2 for AF_INET).

provider—If specified, it is the value from the TNGLOBAL statement.

Reason:

Action: None. This is an Informational message.

T01TE321I task_id cipher_version cipher_specs

task_id—Task identifier.

cipher_version—Version 2 or version 3 cipher spec.

cipher_specs—cipher specs.

Reason: These are the respective cipher type that will be used if not configured in APPCFG.

For V2:

Value	APPCFG Specification	SSL Cipher Specification
1	RC4	SSL_RC4_128_WITH_MD5
2	RC4_EX	SSL_RC4_128_EXPORT40_WITH_MD5
3	RC2	SSL_RC2_CBC_128_CBC_WITH_MD5
4	RC2_EX	SSL_RC2_CBC_128_CBC_EXPORT40_WITH_MD5
6	DES	SSL_DES_64_CBC_WITH_MD5
7	3DES	SSL_DES_192_EDE3_CBC_WITH_MD5

For V3:

Value	APPCFG Specification	SSL Cipher Specification
01	NULL_MD5	SSL_RSA_WITH_NULL_MD5
02	NULL_SHA	SSL_RSA_WITH_NULL_SHA
03	RC4_MD5_EX	SSL_RSA_EXPORT_WITH_RC4_40_MD5
04	RC4_MD5	SSL_RSA_WITH_RC4_128_MD5
05	RC4_SHA	SSL_RSA_WITH_RC4_128_SHA
06	RC2_MD5_EX	SSL_RSA_EXPORT_WITH_RC2_CBC_40_MD5
09	DES_SHA	SSL_RSA_WITH_DES_CBC_SHA
0A	3DES_SHA	SSL_RSA_WITH_3DES_EDE_CBC_SHA

Note: DES_SHA is 56 bits, 3DES_SHA is 168 bits.

Action: None. This is an Informational message.

T01TE322E *task_id* gsk_get_cipher_info Rc=*retcode*

task_id—Task identifier.

retcode—Return code.

Reason: An error occurred issuing gsk_get_cipher_info(). The return code is *retcode*.

Action: Record the return code.

T01TE323E *task_id* accept(*listen_sd*) getsockname(*client_sd*) error *errno errno2*

task_id—Task identifier.

listen_sd—Socket descriptor of the listener.

client_sd—Socket descriptor of the client connection.

errno errno2—Error information from the socket function.

Reason: An error occurred during the getsockname() function. The client connection has been closed.

Action: Record *errno* and *errno2*.

T01TE324E *task_id* setrlimit error *errno*

task_id—Task identifier.

Reason: An error occurred setting the resource limit for the number of sockets.

Action: For error 139, make sure the user ID associated with the address space has superuser privilege. Record *errno*.

T01TE330E Too many PORTs defined.

Reason: There are too many ports defined. Over 2,000.

Action: Reduce the number of ports.

T01TE331E Unable to Set Dub Defaults in *module_name*

module_name—Module name attempting to issue the Set Dub Defaults.

Reason: Attempt to issue OpenEdition Set Dub Defaults failed. Either the kernel is not ready or message T01OE001 will accompany this message.

Action: Examine T01OE001 message or the OpenEdition address space.

T01TE332I Task *task_id* has terminated

task_id—Task identifier.

Reason: A task terminated.

Action: Obtain messages from T01LOG.

T01TE401E Tsess(*id*) Plu *pluname* Slu *sluname* *function* error *error_info* in *module*

Reason: The server session *id* encountered and error for VTAM function *function*. *error_info* is the sense information.

Action: Contact Customer Support.

T01TE402D Tsess(*id*) Plu *pluname* Slu *sluname* FMH-Data received, not negotiated

Reason: The server session *id* received FMH-Data from the client but the client had not negotiated FMH support.

Action: The may be problem with the client. If the problem persist, contact support.

T01TE403D Tsess(*id*) Plu *pluname* Slu *sluname* RESPONSE-FLAG set, not negotiated

Reason: The server session *id* received RESPONSE-FLAG from the client but the client had not negotiated RESPONSE support.

Action: The may be problem with the client. If the problem persist, contact support.

T01TE404I Tsess(*id*) Plu *pluname* Slu *sluname* Intervention Required

Reason: The server session *id* received "Intervention Required" from the identified printer.

Action: Determine the action needed at the printer.

T01TE405D Tsess(*id*) Plu *pluname* Slu *sluname* Negative Response, Sense *sense*

Reason: The server session *id* received a negative response from the client.

Action: If the is causing an error (it is not an error by itself), it may indicate a problem with the client. If the problem persists, contact support.

T01TE406I Tsess(*id*) Plu *pluname* Slu *sluname* Intervention Cleared

Reason: The server session *id* received "Intervention Cleared" from the identified printer.

Action: None. This is an Informational message.

T01TE407I Tsess(*id*) - close timed out

id—Unique identifier associate with the client session.

Reason: A timer was set at the beginning of close processing. The timer expired before close completed. Most likely a VTAM TERMSESS has not completed.

Action: If the problem persists, contact support.

T01TE408W Tsess(*id*) – IDLE timeout

id—Unique identifier associate with the client session.

Reason: The IDLE timer has expired without any activity on the connection.

Action: The client connection is closed.

T01TE500E Tsess(*id*) Plu *pluname* Slu *sluname function* in *module*

id—Unique identifier associate with the client session.

pluname—Name of the requested PLU.

sluname—Name of the ACB associated with the client.

Reason: The server encountered a VTAM error for *function*. This should be accompanied by message T01TE501E.

Action: Use message T01TE501 to determine error.

T01TE501E R15=error_code R00=reason_code rtncd_fdbk2 sense REQ req

Reason: VTAM information to accompany message T01TE500E.

T01TE502E BIND error *sense info* Image: *bind_image*

Reason: VTAM information to accompany message T01TE500E.

T01TE510E Tsess(*id*) Slu *sluname function* error: R15=*error* ACBERFLG=*xx*

id—Unique identifier associated with the client session.

sluname—Name of the ACB associated with the client.

function—Either OPEN or CLOSE.

Reason: The server encountered an error during OPEN or CLOSE of the ACB.

Action: Display the status of the LU. Verify the VTAM configuration. Verify the LU POOL configuration.

T01TE511E GENCB error for EXLST – error *error_info*

Reason: The server encountered an error preparing the VTAM environment.

Action: Contact Customer Support.

T01TE520E Tsess(id) Plu pluname Slu sluname function error error_number in module

id — Unique identifier associate with the client session.

Reason: The Server session identified by *id* encountered a VTAM error for *function*

Action: Contact customer support.

T01TE521I Tsess(*id*) Slu *sluname* Local *ip_address:port* Remote *ip_address:port*

id — Unique identifier associate with the client session.

Reason: This message is written to provide an SLU-name-to-IP-address map.

Action: None. This is an Informational message.

T01TE522I Tsess(*id*) Appl *applid* Plu *pluname* Slu *sluname* Logmode *logmode*

id — Unique identifier associate with the client session.

Reason: The Server session identified by *id* has initiated a session with the requested application.

Action: None. This is an Informational message.

T01TE616D Tsess(*id*) Data received during negotiation: *data*

id — Unique identifier associate with the client session.

data — The data.

Reason: The Server session identified by *id* received non-negotiation data during telnet negotiation. The data is dumped to the message log. Processing continues.

Action: This may indicate a problem with the client. If the problem persists, contact Customer Support.

T01TE617E Tsess(*id*) Invalid Tn3270E negotiation: *input*

id—Unique identifier associate with the client session.

input—Input received from the client.

Reason: The Server session identified by *id* received an invalid TN3270E negotiation command. The input is dumped to the message log. The session is terminated.

Action: This may indicate a problem with the client. If the problem persists, contact Customer Support.

T01TE618E Tsess(*id*) Invalid Tn3270E datastream: *input*

id—Unique identifier associate with the client session.

input—Input received from the client.

Reason: The Server session identified by *id* received an invalid TN3270 data stream in fullscreen mode. The input is dumped to the message log. The session is terminated.

Action: This may indicate a problem with the client. If the problem persists, contact Customer Support

T01TE619E Tsess(*id*) Invalid Tn3270E header: *input*

id—Unique identifier associate with the client session.

input—Input received from the client.

Reason: The Server session identified by *id* received an invalid TN3270E header in a TN3270E message. The input is dumped to the message log. The session is terminated.

Action: This may indicate a problem with the client. If the problem persists, contact Customer Support.

T01TE620E Tsess(*id*) Error in translate table *table*

id—Unique identifier associate with the client session.

table—Name of the translate table.

Reason: An error occurred locating or loading the translate table.

Action: Verify configuration.

Chapter

37 T01UD*nnn* Server UDP Messages

This chapter describes messages issued by the Server UDP mini services. These messages include T01UD000 through T01UD999.

T01UD*nnn*

T01UD001I UDPSERV starting

Reason: UDP mini-service started. (echo, discard, date/time, or char-gen).

T01UD002I UDPSERV closing -- SEND: *n segs n bytes*. RECVD: *n segs n bytes*

Reason: This message provides statistics about the UDP protocol.

n	Number of segments or bytes transmitted or received.
segs	UDP segments transmitted or received.
bytes	Octets transmitted or received by UDP.

Chapter

38

T01US*nnn* User SMTP Messages

This chapter describes messages issued by the User SMTP. These messages are T01US000 through T01US999.

T01US*nnn*

T01US001I LPR (*rem_host*) access from remote network not allowed

Reason: A print request was received from a host that is not on the local network. The print request from the remote host (*rem_host*) was ignored and the connection was terminated.

Action: Verify the APPCFG*xx* PARM member for LPR restricted access and modify if appropriate.

T01US002I LPR (*rem_host*) unsupported LINE PRINTER protocol command

Reason: The Line Printer Server (LPS) received an unknown command from a remote host (*rem_host*). Valid commands include lpc, lpr, lpq, and lprm. The command is discarded and the connection is released.

Action: Notify the remote client that this command is unknown or not supported.

T01US003I LPR (*rem_host*) no line feed <LF> found in command line

Reason: The LPS received a command or subcommand that was not terminated with a line feed from remote host *rem_host*.

Action: Notify the remote client that this command format is incorrect.

T01US004I LPR (*rem_host*) bad LPR subcommand received

Reason: The LPS received an lpr subcommand that was either unknown or unsupported from remote host *rem_host*. Valid lpr subcommands include abort, control, and data. The subcommand is discarded, the open file is closed and the connection is released.

Action: Notify remote client that this lpr subcommand is bad.

T01US005I (*rem_host*) invalid printer queue name

Reason: An alternate printer was requested from the lpr's print job command, but either no name was found or the name was greater than eight characters.

Action: If the SELECT option was specified on the APPCFG*xx* LPR statement, the default DESTID is tried. If the FLUSH option was specified, then the file is purged.

T01US006I (*rem_host*) Sysout dynamic allocation failed

Reason: On receipt of an lpr command from another host, an attempt is made to allocate resources to perform the print request. One such resource is the SYSOUT data set. Verify options selected in APPCFG*xx*'s LPR statement. If the installation specified NOSELECT, then the default DESTED failed. If SELECT or FLUSH was specified, then the remote "print queue name" was invalid.

Action: If the SELECT option was specified on APPCFG*xx*'s LPR statement, then the default DESTID is tried. If the FLUSH option was specified, then the file is purged. If the default DESTID is invalid verify name is defined to JES.

T01US007I (*rem_host*) Sysout DCB open failed

Reason: After the SYSOUT data set is allocated, USPOOL attempts to open it for use. Any failure with opening the SYSOUT data set causes the print request to be ignored and the TCP connection to be closed.

Action: See the description of message T01US006I for additional information.

T01US008I (*rem_host*) Sysout dynamic allocation FREE failed

Reason: When the remote host (*rem_host*) closes the TCP connection after sending the contents of the file it wanted printed, the SYSOUT data set is released. No matter how the free request completes, the TCP connection is closed.

Action: See the description of message T01US006I for additional information.

T01US009I (*rem_host*) Sysout OUTLIM exceeded

Reason: The number of lines printed by USPOOL for this LPR request exceeded the site maximum as specified on the OUTLIM parameter of the APPCFG*xx*'s LPR statement. The rest of the output is flushed.

Action: If the site-specified limit for OUTLIM is too small, increase the OUTLIM value of the LPR statement in APPCFG*xx* to a higher value. JES2 users can specify OUTLIM=0 on the LPR statement to disable SYSOUT outline checking.

T01US010I LPR (*rem_host*) network record length exceeds Sysout LRECL

Reason: An lpr data file was received that contained a record longer than the LRECL of the SYSOUT data set (254 characters). The SYSOUT file is closed and the connection released. The remaining output is flushed.

Action: Examine the file being printed to ensure that it contains eor (end-of-record) markers. Data files must contain eor markers (LF, FF, CR) to indicate end of record. lpr records in excess of 254 characters are not supported.

T01US011I LPR (*rem_host*) LPR ABORT job subcommand received

Reason: The LPS received an lpr abort subcommand from a remote host (*rem_host*). The SYSOUT file is closed and the connection released.

Action: None. This is an Informational message.

T01US012I LPR (*rem_host*) unsupported PRINT JOBS command received

Reason: The LPS received a "print any waiting jobs" (lpc) command from a remote host (*rem_host*). This command is not supported by the LPS. The request is discarded and the connection is released.

Action: Notify the remote client that this command is not supported.

T01US013I LPR (*rem_host*) unsupported QUEUE STATE command received

Reason: The LPS received a "queue state" (lpq) command from a remote host (*rem_host*). This command is not supported by the LPS. The request is discarded and the connection is released.

Action: Notify the remote client that this command is not supported.

T01US014I LPR (*rem_host*) unsupported REMOVE JOBS command received

Reason: The LPS received a "remove from queue" (lprm) command from a remote host (*rem_host*). This command is not supported by the LPS. The request is discarded and the connection is released.

Action: Notify the remote client that this command is not supported.

T01US015I LPR (*rem_host*) idle timeout expired; connection abort

Reason: The remote host did not respond within the allotted time.

Action: Retry later.

T01US016E LPR configuration definition statement missing; port = *port_num*

Reason: There is no port defined under the LPR statement in member APPCFG*xx*.

Action: Define a port in member APPCFG*xx*.

T01US017E LPR(*rem_addr*) bad data in bytes RECVD COUNT or > than 100Meg.

Reason: The remote client transmitted a data file header containing an invalid "data bytes to be received count". The field either has bad characters that are not numbers or the number is greater than 99,999,999. *rem_addr* is the Internet address in dot notation of the remote host.

Action: A TCPEEP trace verifies the data being received. The client should be corrected so that it contains the correct value in the "data bytes to be received count".

T01US018W Local Subsystem ID: *ssid* Lookup error is: *ac.ec text*, Diag=x'*code*'

Reason: A GET HOST BY NAME call to DNR for the specified subsystem ID failed. See "DNR Return Codes" in *Unprefixed Messages and Codes* for more information on action, error, and diagnostic codes.

ssid	The subsystem ID.
ac	The action code.
ec	The error code.
text	The message text.
code	The diagnostic code.

Action: The error message and codes should direct a user's action. One possible reason is that the local subsystem ID is not defined in the DNR alias member.

This message can be due to a temporary resource initialization coordination problem or it can be due to a bad DNR configuration. Multiple occurrences of this message may be an indicator of configuration or network problems.

The TCP base product must be able to resolve the subsystem name into a host name. One can use the TSO command DNRGET to verify whether the TCP base product can correctly map the subsystem name into a host name.

An MVS host IBMHOST.COLUMBIA.COM using the subsystem name ACSS with an IP address of 138.42.32.24 can be verified with the following TSO command:

```
DNRGET H BYN ACSS SYSID(ACSS)
```

If the response from the DNRGET command does not indicate a correct host name and IP address combination then one should check the installation's DNR definitions in configuration files DNRALC*xx* and DNRHST*xx*.

These are the DNRALC*xx* file entries for the local host BMHOST.COLUMBIA.COM:

```
ACSS           IBMHOST.COLUMBIA.COM.
IBMHOST  IBMHOST.COLUMBIA.COM.
```

These are the DNRHST*xx* file entries for the local host IBMHOST.COLUMBIA.COM:

```
IBMHOST.COLUMBIA.COM.        138.42.32.24
```

T01US019W Local Subsystem ID: *ssid* Lookup error return code: *rc*

Reason: A GET HOST BY NAME call to DNR for the specified subsystem ID failed. Module GETHBDNR could not accommodate the request and returned the specified error code.

ssid The subsystem ID.

rc The error code:

- -2 Bad arguments were passed to GETHBDNR.
- -4 No core was available for the request.
- -5 A software error occurred when processing the DIRSRV call in GETHBDNR.

Action: This error could be temporary or indicative of other problems within the product. If the problem persists, check for related SVC dumps that may have been issued. The T01LOG and JES log of the job should be saved. Contact Customer Support. This message can be due to a temporary resource initialization coordination problem or it can be due to a bad DNR configuration. Multiple occurrences of this message may be an indicator of configuration or network problems.

The TCP base product must be able to resolve the subsystem name into a host name. One can use the TSO command DNRGET to verify whether the TCP base product can correctly map the subsystem name into a host name.

An MVS host IBMHOST.COLUMBIA.COM using the subsystem name ACSS with an IP address of 138.42.32.24 can be verified with this TSO command:

```
DNRGET H BYN ACSS SUBSYS(ACSS)
```

If the response from the DNRGET command does not indicate a correct host name and IP address combination then one should check the installation's DNR definitions in configuration files DNRALC*xx* and DNRHST*xx*.

These are the DNRALC*xx* file entries for the local host IBMHOST.COLUMBIA.COM:

```
ACSS              IBMHOST.COLUMBIA.COM.
IBMHOST       IBMHOST.COLUMBIA.COM.
```

These are the DNRHST*xx* file entries for the local host IBMHOST.COLUMBIA.COM:

```
IBMHOST.COLUMBIA.COM.                138.42.32.24
```

T01US020I Null x-header found

Reason: The X-from or X-to header in the mail file contained neither a user nor host address. The outbound mail request is rejected. Check the log for related messages for the data set name.

Action: Delete the mail file data set. Try to find which software generated this mail message with a null X-header and correct it.

T01US021I Bad character *c* in x-*header*

Reason: An X-from or X-to header in the mail file contains incorrect syntax. Character *c* is the start of the incorrect syntax on the X-header. *header* is the X-header data. The outbound mail request is rejected. Check the log for related messages for the data set name.

Action: Delete the mail file data set. Try to find which software generated this mail message with bad syntax and correct it.

T01US022I Bad Literal string in x-header: xxxxxxx

Reason: An X-from or X-to header in the mail file contains incorrect syntax. A literal string does not terminate in the X-header. The X-header terminates before the ending double quote is found. *xxxxxxx* is the X-header data. The outbound mail request is rejected. Check the log for related messages for the data set name.

Action: Delete the mail file data set. Try to find what software generated this mail message with bad syntax and correct it.

T01US023I Bad Domain Literal in x-header: xxxxxxx

Reason: An X-from or X-to header in the mail file contains incorrect syntax. A domain literal contains incorrect syntax. *xxxxxxx* is the X-header data.

The outbound mail request is rejected. Check the log for related messages for the data set name.

Action: Delete the mail file data set. Try to find which software generated this mail message with bad syntax and correct it.

T01US024I X-header enclosed incorrectly: xxxxxxx

Reason: An X-from or X-to header in the mail file contains incorrect syntax. Special characters < and > are found in the X-header text. These special characters are allowed to envelope X-header message text but they must be balanced (that is, the beginning of the message text contains < while the end of the message text should be >.) *xxxxxxx* is the X-header text.

The outbound mail request is rejected. Check the log for related messages for the data set name.

Action: Delete the mail file data set. Try to find which software generated this mail message with bad syntax and correct it.

T01US025I No local user found in x-header: xxxxxxx

Reason: An X-from or X-to header in the mail file contains incorrect syntax. Only hosts and routes were found in the X-header. *xxxxxxx* is the X-header text.

The outbound mail request is rejected. Check the log for related messages for the data set.

Action: Delete the ail file data set. Try to find which software generated this mail message with bad syntax and correct it.

T01US026I No host found in x-header: xxxxxxx

Reason: An X-from or X-to header in the mail file contains incorrect syntax. No host or route was found in the X-header. *xxxxxxx* is the X-header text. The outbound mail request is rejected. Check the log for related messages for the data set.

Action: Delete the mail file data set. Try to find which software generated this mail message with bad syntax and correct it.

Chapter

39 T01VT*nnn* VTAMAPPL Messages

This chapter describes messages issued by VTAMAPPL. The messages are T01VT000 through T01VT999.

T01VT*nnn*

T01VT001E ACB open error: R15=*code1* ACBERFLG=*code2* ACBname=*name*

Reason: VTAMAPPL attempts to open a VTAM ACB but the open fails.

code1 VTAM R15 OPEN return code.

code2 VTAM OPEN error flag setting.

name ACB name used by the application.

Action: Codes 1 and 2 are described in the IBM manual *VTAM Programming*. Take corrective action accordingly. In many cases the VTAM APPLID definition is the cause of this error.

T01VT002E RPL ERROR: R15= *code1* R0= *code2* REQ= *code3* RTNCD= *code4* SSEI= *code5* SSEO= *code6* with offset= *code7*

Reason: VTAMAPPL attempted an RPL-based VTAM request but the request failed.

These are the error codes:

code1	VTAM macro return code.
code2	VTAM request return code.
code3	VTAM request type.
code4	VTAM RPL return code.
code5	VTAM RPL input sense data.
code6	VTAM RPL output sense data.
code7	Offset in the module that issued the message.

Action: Verify that VTAM is functioning properly. The RPL codes are described in the IBM manual *VTAM Programming*.

T01VT003E INSUFFICIENT MEMORY

Reason: VTAMAPPL makes a request for pcore storage but insufficient storage is available.

Action: Save all output from the job. Check the WTO and JOB logs for related messages. Check the ACSNAP DD for any related dump. Use ACTEST to issue the psnap command to print all of the product's storage into the ACSNAP DD. If there is a proliferation of the same kind of storage, a ptask may be allocating storage in a loop.

Retry the operation first on the current version of the product. If it fails again, bring the product down. Raise the region size on the job. Start up the product. Retry the operation. If the error recurs, contact Customer Support.

T01VT004E VTAMappl - initialized VTAM Applid (Plu) = *applid*

Reason: The specified VTAM APPLID was initialized, which is gotten from the APPCGF*xx* member on the HOST parameter for ACBNAME(*applid*). This message does not appear if the ACB OPEN fails.

T01VT005D *text*

Reason: Module VTAMECHO was invoked to echo data from a Telnet session. *text* is the data that was echoed. The requested data is logged.

Chapter

40 T02IU*nnn* IUCV Messages

This section describes messages issued by the IUCV routine. These messages include T02IU000 through T02IU099.

T02IU*nnn*

T02IU001I IUCV initialization successfully completed, subsystem is *subsys_name*

Reason: The TCP/IP IUCV address space successfully completed all initialization steps and is ready to process IUCV requests. The subsystem interface named *subsys_name* has been activated.

Note: Any pre-existing VMCF environment has been superseded.

Action: None. This is an Informational message.

T02IU002I IUCV termination successfully completed

Reason: The TCP/IP IUCV address space successfully completed the termination process and can no longer process IUCV requests.

Note: Any pre-existing VMCF environment has been restored.

Action: None. This is an Informational message.

T02IU003E IUCV initialization failed, reason code = *rsn_code*

Reason: The TCP/IP IUCV address space failed to initialize. The reason code identifies the reason for the failure.

4	IUVT allocation failed.
8	SSCVT allocation failed.
12	VMCF allocation failed.
16	IUVX allocation failed.
20	ASVB allocation failed.
24	IASB pool header load failed.
28	EORA pool header load failed.
32	PTPA pool header load failed.
36	IUMS pool header load failed.
40	Subsystem name already active.

Action: Allocation failures are generally due to insufficient storage in either the private area of the address space, or in CSA. Make sure sufficient storage exists.

Pool header load failures are generally due to either insufficient storage in the private area of the address space or the module could not be located. Pool header module names have the format T02L*xxx*P, where *xxx* is the first three characters of the pool name (for example, pool header for IASB is T02LIASP).

T02IU004W IUCV storage release of VMCF_CVT failed, reason code = *rsn_code*

Reason: During termination processing, an attempt was made to release storage being used for the VMCF_CVT. The VMCF_CVT is allocated in common storage. *rsn_code* is the return code from the Storage Release service.

Action: Check the return code from the Storage Release service and take the appropriate action.

Chapter 41

T03PE*nnn* TCPEEP Messages

This chapter describes messages issued by the TCPEEP trace facility. These messages include T03PE001 through T03PE999.

T03PE*nnn*

T03PE000I TCPEEP Processing Completed

Reason: TCPEEP completed processing.

Action: None. This is an Informational message.

T03PE001E The Trace A.S. (Subsystem Id *ssid*) is not active

Reason: The Trace Address Space for subsystem *ssid* is not active. If TRCSSID(*ssid*) was not specified then the default subsystem ACTR is not active. If TRCSSID(*ssid*) was specified then that Trace address space is not active.

Action: Either start the Trace address space or use TRCSSID(*ssid*) to use a Trace address space other than the default.

T03PE002E HALT REQUEST REQUIRES INSTANCE

Reason: The HALT keyword was specified with the INSTANCE keyword.

Action: Reissue the request specifying the INSTANCE keyword.

T03PE003E TCPEEP requires a Trace Address Space separate from TCP/IP The Trace A.S. (Subsystem Id *ssid*) is not active. TCPEEP cannot initialize without the Trace Address Space

Reason: The trace address space for subsystem *ssid* is not active. This usually means the Trace Address Space was never started (since the last IPL). This address space is separate from the stack running TCP/IP. If TRCSSID(*ssid*) was not specified, then the default subsystem ACTR is not active. If TRCSSID(*ssid*) was specified, then that Trace address space is not active.

Action: Either start the Trace address space or use TRCSSID(*ssid*) to use a trace address space other than the default.

T03PE004E MAXIMUM TRACE INSTANCES ALREADY ACTIVE

Reason: A request to start a new trace instance was made but the maximum number has already been reached.

Action: Issue TRACE CT,ON,COMP=*ssid* and reply with R *nn*,OPTIONS=(STATUS),END to display all the active traces. You must stop one of the active traces before initiating a new trace.

T03PE005E ERROR PARSING GROUPS

Reason: Either the *group_id* is too long, or the *filter* data is too long, or extraneous data after the filter specification was entered.

Action: Verify the (*group_id,'filter'*) specification is correct.

T03PE006E Invalid GROUP name *group_id*

Reason: A *group_id* was entered that is invalid.

Action: See the administrator or system management guide for valid names.

T03PE007E Invalid combination of parameters

Reason: An invalid combination of parameters were entered, for example, HALT and SYSOUT.

Action: See the syntax in the *Administrator* or *System Management Guide* .

T03PE008I *nnnn* RECORDS, ENTER 'H' TO HALT TCPEEP TRACE

Reason: This is issued in response to an attention interrupt (ATTN key for LU2 or PA1 for LU0).

Action: To continue the trace press the ENTER key with no data. To stop the trace, enter **H**.

T03PE009I SYSOUT allocated to *ddname*

Reason: A dynamically allocated SYSOUT data set has been allocated.

Action: None. This is an Informational message.

T03PE010I TRACE INITIATED - INSTANCE: *nn*

Reason: A trace has been started with instance number *nn*.

Action: None. This is an Informational message.

T03PE011S Abend SDWANAME SDWAMODN SDWACMPC SDWAGR12 SDWANXT1

Reason: An abend occurred. Various SDWA fields are displayed.

Action: Contact Customer Support.

T03PE012E TRACE UP FAILED: RC=*rc* RSN=*reason*

Reason: An error occurred initiating a trace.

Action: Contact Customer Support.

T03PE013E TRACE DOWN FAILED: RC=*rc* RSN=*reason*

Reason: An error occurred stopping a trace.

Action: Contact Customer Support.

T03PE014E TRACE READ FAILED: RC=*rc* RSN=*reason*

Reason: An error occurred retrieving trace data.

Action: Contact Customer Support.

T03PE020E SYSOUT ALLOCATION ERROR RC=*rc* ERR=*error* INFO=*info*

Reason: An error occurred during dynamic allocation.

Action: Contact Customer Support.

T03PE021E Error OPENing SYSOUT RC=*rc*

Reason: An error occurred during OPEN for the dynamically allocated SYSOUT data set.

Action: Contact Customer Support.

T03PE022E Error Generating FILTER RC=*rc* CC=*cc func*

Reason: An error occurred generating the filter. *func* is either STOR or PARS for STORAGE or IKJPARS respectively.

Action: If *func* is PARS make sure the filter parameters are valid.

T03PE023E Error LOADing *module*

Reason: An error occurred trying to load *module*.

Action: Make sure the LINK library is in either the STEPLIB or the linklist.

T03PE024E No STORAGE for PROGRAM Table

Reason: An error occurred trying to obtain storage.

Action: Inspect the REGION size for TSO.

T03PE025W Trace data lost

Reason: Trace data was lost.

Action: To prevent lost trace data, use one of the following methods:

- Change the BUFFERS parameter
- Use an in-memory wrap trace
- Use the external writer

Chapter 42

T03TR*nnn* Trace Messages

This section describes messages issued by the trace routine. These messages include T03TR000 through T03TR999.

T03TR*nnn*

Messages T03TR000 through T03TR899 are issued in the trace address space. T03TR9*nn* messages are issued by the T03 Trace Facility in processing the OPTIONS= parameters of the TRACE CT,ON,COMP**=** command and the TRACE CT,OFF,COMP= command.

No T03TR*nnn* Message Displayed

If no T03TR*nnnn* message is displayed for your TRACE CT command, look for:

ITT004I START/STOP FAILED FOR TRACE COMP=ssid RETURN=00000008, REASON= *nnnnnnnn*.

Reason: An error occurred and the return code and reason code were passed back to the TRACE CT command.

Action: Contact Customer Support.

T03TR000I TRACE *ssid* initialization started

Reason: The trace address space using subsystem id *ssid* has started initialization.

Action: None. This is an Informational message.

T03TR001I TRACE *ssid* initialization successfully completed

Reason: The trace address space using subsystem ID *ssid* has completed initialization.

Action: None. This is an Informational message.

T03TR002I TRACE *ssid* termination successfully complete

Reason: The trace address space using subsystem ID *ssid* has terminated.

Action: None. This is an Informational message.

T03TR003E TRACE *ssid* initialization failed; reason code = *rsn_code*

Reason: The trace address space using subsystem ID *ssid* failed to initialize. The reason code is *rsn_code*.

Action: Contact Customer Support.

T03TR004E STORAGE OBTAIN error for TAB (and TGV)- R15=*ret_code* R00=*rsn_code*

Reason: Unable to get common storage.

Action: There may be a problem with CSA. If the problem persists, call Customer Support.

T03TR005E DSPSERV CREATE error - R15=*ret_code* R00=*rsn_code*

Reason: Could not create a data space for trace buffers.

Action: Call Customer Support.

T03TR006E ALESERV ADD error - R15=*ret_code* R00=*rsn_code*

Reason: Could not modify the primary address space access list.

Action: Call Customer Support.

T03TR007E CTRACE DEFINE error - R15=*ret_code* R00=*rsn_code*

Reason: Could not define our trace address space to the IBM component trace facility.

Action: Call Customer Support.

T03TR008E STORAGE OBTAIN error for TIV - R15=*ret_code* R00=*rsn_code*

Reason: Unable to get private storage.

Action: Check the REGION size. If the problem persists, call Customer Support.

T03TR101E UNALLOWABLE FILTER KEYWORD

Reason: A filter keyword was specified but is not allowed.

Action: Verify the filters for the specified group_id.

T03TR102E HOST IP NAME/ADDRESS UNRESOLVABLE

Reason: This is from the HOST(*host*) specification in the filter data. The host IP name was not able to be resolved using the specified DNR.

Action: Either use a dotted IP address or specify DNRSSID(*ssid*).

T03TR103E PORT NAME OR NUMBER UNRESOLVABLE

Reason: This is from the PORT(*port*) specification in the filter data. The name could not be converted or the number was invalid.

Action: Specify a well-known port name or use a valid number.

T03TR104E PROTOCOL UNRESOLVABLE

Reason: This is from the PROTOCOL(*proto*) specification in the filter data. The name could not be converted or the number was invalid.

Action: Specify a valid name or number.

T03TR105E MAX DATA VALUE OUT OF RANGE

Reason: The filter parameter to limit data was out of range.

Action: Respecify using correct value.

T03TR901E Abend xxxx occurred processing TRACE CT (+xxxx) Base(xxxxxxxx) PSW(xxxxxxxx)

Reason: An abend occurred processing the reply.

Action: Contact Customer Support.

T03TR902E Abend xxxx during TRACEUP/TRACEDN PC call

Reason: An abend occurred in the trace address space processing request.

Action: View the T01LOG, JESMSGLG and JESYSMSG to determine the error. Contact Customer Support with the logs available.

T03TR903E Invalid OPTIONS specified

Reason: A request was made to start, modify or stop a trace instance but the specified parameters where invalid.

Action: Review the OPTIONS= specification and reissue the command with the necessary corrections.

T03TR904E Instance(xx) is not an active trace

Reason: A request was made to modify or stop a trace instance but the specified instance is not active.

Action: Verify that you are using the correct Instance ID. Issue **R *nn*,OPTIONS=(STATUS),END** to get a list of active instances. An instance could have been stopped by another operator.

T03TR905E HALT specification requires Instance

Reason: A request was made to stop a trace instance but the trace instance was not specified.

Action: Reissue the command with INSTANCE(*x*) specified.

T03TR906E Writer connection change requires Instance

Reason: A request was made to change the connection status (connect or disconnect) but the WTR= keyword was specified without INSTANCE(*nn*).

Action: Reissue the command with INSTANCE(*nn*) specified.

T03TR907E Writer Disconnect invalid. Instance(xx) is not connected to writer

Reason: A request was made to disconnect an external writer from a trace instance that was not connected to an external writer.

Action: Retry the command without requesting the writer be disconnected. Issue **R *nn*,OPTIONS=(STATUS),END** to get a list of active instances. An instance could have been stopped by another operator.

T03TR908E Invalid GROUP name of *gr_id*

Reason: GROUPS((group_id [,'filter')) was entered but the group ID *gr_id* is not valid.

Action: Use one of the documented group names. See the *Administrator* or *System Management Guide* for more information.

T03TR909I Trace start successful Instance(xx)

Reason: The shown trace instance was successfully started. Remember *xx* for any request that require INSTANCE(*xx*) in the specification.)

Action: None. This is an Informational message.

T03TR910I Trace shutdown successful Instance(xx)

Reason: The trace instance stop processing for the shown trace instance was successful.

Action: None. This is an Informational message.

T03TR911I Trace modify successful Instance(xx)

Reason: The trace instance modify processing for the shown trace instance was successful.

Action: None. This is an Informational message.

T03TR912E Trace start error RC(xx) Reason(xx) Info(xxxx)

Reason: The trace instance start processing failed. The return code and reason code are displayed.

Return Code (RC) Information:

RC=16 Indicates T03CTSUP was entered without an ITCB, REGION size too small.

RC=12 Implies the Trace Address space is not active.

RC=08 Indicates an error occurred. The Reason Codes are:

For Invalid *xxxx*, the control block ID is not valid.

01 Invalid Trace Instance.

02 Invalid Trace Group ID see the *System Management Guide* for valid names.

03 DSPSERV Error—information is return code from DSPSERV.

04 TFE Storage Obtain Error—information is return code from STORAGE.

05 ITIM ALLOC Error—information is return code from ITIME.

06 Trace address space not active.

07 Max trace limit exceeded.

09 Invalid TFPL.

10 TFPL Storage Obtain Error—information is return code from STORAGE.

11 Invalid TAB.

12 Invalid TIV.

13 TSKB Storage Obtain/Release Error—information is return code from STORAGE.

14 Invalid TDSB.

15 ILOAD Error for Sink Exit—information is return code from ILOAD.

16 T03STM2Q address error—address is zero.

17 ITIME SET Error—information is return code from ITIME.

18 Invalid TSCB.

19 TGV area size error—internal error with TGV allocation.

20 Buffer request exceeds segment size.

21 No free TSCB control blocks—internal error.

22 Segment not on page boundary—internal error.

23 DataSpace ALET error—information is return code from TESTART.

24 SFRR error—information is return code from SFRR

25 ILOAD error for T03STBFF—information is return code from ILOAD.

30 Abend.

The info(*xxx*) field is variable. For example, on RC=08 with "TFPL Storage Obtain Error —the INFO field is the return code from STORAGE". So, INFO will contain the return code from the MVS STORAGE macro. Customers or Customer Support can refer to the IBM manuals for the return code of the STORAGE macros. Sometimes, the INFO is blank, other times it will contain the return code from internal macros, while others are return codes from network macros. This is primarily for Customer Support use.

Action: Contact Customer Support.

T03TR913E Trace shutdown error for Instance(xx) RC(xx) Reason(xx) Info(xxxx)

Reason: The trace instance stop processing failed. The return code and reason code are displayed.

Return Codes and Reason Codes:

RC=16 Indicates T03CTTDN was entered without an ITCB, REGION size too small.

RC=12 Implies the Trace Address space is not active.

RC=04 Indicates an error occurred. The reason codes are:

For Invalid *xxxx*, the control block ID is not valid

01 Invalid trace instance.

02 TSKB Storage Release Error—information is return code from STORAGE.

03 ITIM Error—information is return code from ITIME.

04 DSPSERV Release error—information is return code from DSPSERV.

05 Abend.

06 Trace address space not active.

07 Invalid TAB.

08 Invalid TIV.

09 Invalid TGV.

10 Invalid TFE.

11 Invalid TFPL.

12 Invalid TSKB.

13 Invalid TSCB.

14 Invalid TDSB.

15 TGV area size error—internal error with TGV allocation.

16 TFPL Storage release error—information is return code from STORAGE.

17 ECB Post error—information is return code from POST.

18 Flush Exit error —error from routine to flush buffers.

19 ECB Storage Obtain/Release error—information is return code from STORAGE.

20 Buffer Flush error—information is return code from ITIME.

21 Invalid TBUF.

22 TRTM Parm Storage Obtain/Release error—information is return code from STORAGE.

23 DataSpace ALET error—information is return code from TESTART.

24 ILOAD for T03STRTM failed—information is return code from ILOAD.

25 Tear Down already running—requested instance already being stopped.

Action: Contact Customer Support.

T03TR914E Trace modify error for Instance(xx) RC(xx) Reason(xx) Info(xxxx)

Reason: A request to modify an instance failed. The return code and reason code are displayed. See T03TR912E for more information.

Action: Contact Customer Support.

T03TR915E Error processing Filter specification RC(xx) Reason(xx)

Reason: An error occurred processing the filter parameter in a GROUPS specification. The return code and reason code are displayed. See T03TR912E for more information.

Action: Verify that the filter applies to the specified group_id.

T03TR916I Instance(xx) Active, records=*records*

Reason: The results of STATUS show the active trace instances and the number of trace records that were captured for the instance.

Action: None. This is an Informational message.

T03TR917E Disconnect Writer before halting Instance

Reason: A request was made to HALT a Trace Instance but the instance is still connected to an external writer.

Action: Either disconnect the writer, or specify WTR=DISCONNECT with the HALT request.

T03TR918I No active Trace Instances

Reason: In response to a STATUS request, there were no active traces.

Action: None. This is an Informational message.

T03TR919E Writer Connect invalid for existing WRAP trace

Reason: A WRAP type trace (default if WTR=writer is not specified) was started and a modify request with WTR=writer was attempted. A writer cannot be started for a WRAP type trace.

Action: Either start the trace with WTR=writer or start it specifying option NOWRAP in OPTIONS=(*options*).

T03TR920E IKJPARS R15=*nnnnnnnn*

Reason: The message indicates that IKJPARS returned an error condition.

Action: If the parameters are correct, collect the messages from the SYSLOG and contact Customer Support.

Appendix

A error *rc* Mapping

Use the *string* located below and the SSL function from the message to determine the error as described in the IBM manual *System Secure Socket Layer Programming Guide and Reference.*

Mapping Error Strings to Error Return Codes

error *string*	**error *rc***
GSK_ERROR_NO_CIPHERS	-1
GSK_ERROR_NO_CERTIFICATE	-2
GSK_ERROR_BAD_CERTIFICATE	-4
GSK_ERROR_UNSUPPORTED_CERTIFICATE_TYPE	-6
GSK_ERROR_IO	-10
GSK_ERROR_BAD_MESSAGE	-11
GSK_ERROR_BAD_MAC	-12
GSK_ERROR_UNSUPPORTED	-13
GSK_ERROR_BAD_CERT_SIG	-14
GSK_ERROR_BAD_CERT	-15
GSK_ERROR_BAD_PEER	-16
GSK_ERROR_PERMISSION_DENIED	-17
GSK_ERROR_SELF_SIGNED	-18
GSK_ERROR_BAD_MALLOC	-20
GSK_ERROR_BAD_STATE	-21
GSK_ERROR_SOCKET_CLOSED	-22
GSK_ERROR_GSK_INITIALIZATION_FAILED	-23
GSK_ERROR_HANDLE_CREATION_FAILED	-24
GSK_ERROR_BAD_DATE	-25
GSK_ERROR_BAD_KEY_LEN_FOR_EXPORT	-26

error *string*	**error *rc***
GSK_ERROR_NO_PRIVATE_KEY	-27
GSK_BAD_PARAMETER	-28
GSK_ERROR_INTERNAL	-29
GSK_ERROR_WOULD_BLOCK	-30
GSK_ERROR_LOAD_GSKLIB	-31
GSK_ERROR_NO_INITIALIZE	-32
GSK_ERROR_SRB	-33
GSK_SOC_BAD_V2_CIPHER	-40
GSK_SOC_BAD_V3_CIPHER	-41
GSK_SOC_BAD_SEC_TYPE	-42
GSK_SOC_NO_READ_FUNCTION	-43
GSK_SOC_NO_WRITE_FUNCTION	-44
GSK_ERROR_UNKNOWN_ERROR	-99
GSK_SOC_BAD_SEC_TYPE_COMBINATION	-102
GSK_INITIALIZE_OK	0
GSK_KEYFILE_IO_ERROR	1
GSK_KEYFILE_OPEN_FAILED	2
GSK_KEYFILE_BAD_FORMAT	3
GSK_KEYFILE_BAD_PASSWORD	4
GSK_KEYFILE_BAD_MALLOC	5
GSK_KEYFILE_NOTHING_TO_WRITE	6
GSK_KEYFILE_WRITE_FAILED	7
GSK_KEYFILE_NOT_FOUND	8
GSK_KEYFILE_BAD_DNAME	9
GSK_KEYFILE_BAD_KEY	10
GSK_KEYFILE_KEY_EXISTS	11
GSK_KEYFILE_BAD_LABEL	12
GSK_KEYFILE_DUPLICATE_NAME	13
GSK_KEYFILE_DUPLICATE_KEY	14
GSK_KEYFILE_DUPLICATE_LABEL	15
GSK_BAD_FORMAT_OR_INVALID_PW	16
GSK_WARNING_INVALID_SERVER_CERT	98
GSK_WARNING_INVALID_SERVER_PRIV_KEY	99

error *string*	**error *rc***
GSK_ERR_INIT_PARM_NOT_VALID	100
GSK_INIT_SEC_TYPE_NOT_VALID	102
GSK_INIT_V2_TIMEOUT_NOT_VALID	103
GSK_IGSK_KEYFILE_CERT_EXPIREDNIT_V3_TIMEOUT_NOT_VALID	104
	105

Computer Associates™