

Chapter 19. AMAPDUPL: Problem Documentation Upload Utility

The IBM z/OS Problem Documentation Upload Utility (PDUU) is a utility that sends large amounts of documentation in a more efficient manner than sending one large data set to IBM sites. This utility sections the input data set (such as stand-alone dump data set) into smaller data sets that are compressed and sent in parallel using multiple, simultaneous transfer sessions. This results in shorter transmission time for very large data sets. You can also encrypt the data sets. These sessions can send diagnostic documentation to IBM using File Transfer Protocol (FTP) or Secured Hypertext Transfer Protocol (HTTPS).

There are two work buffers for each transfer session (the "A" buffer and the "B" buffer). Each "A" work buffer is filled by copying records from the input data set. When the "A" buffer is full, the sessions are started in parallel. At the same time, each "B" work buffer is filled by copying records from the input data set. When the "B" buffer is full and the transfer of the "A" buffer is complete, transfer of the next "B" buffer starts. This process continues between the "A" and the "B" buffers, until everything in the input data set is sent.

You can have up to 20 transfer sessions running simultaneously, specifiable by the CC_FTP or CC_HTTPS parameter.

For FTP sessions, data is buffered into work data sets. The work data sets are dynamically allocated and can range in size from 1 MB to 9,999 MB. You can experiment to see what works best in your environment, but here are some guidelines:

- Start with three or four parallel FTP sessions. Too many parallel FTP sessions can saturate the network link.
- Use medium size work data sets.

For HTTPS sessions, data is buffered into 31-bit storage. When choosing a WORK_SIZE value, note that you may have limited private storage available (managed on an installation basis) and this number will be used as the size of the buffer.

Each WORK_SIZE buffer sent to IBM results in the creation of a numbered file that IBM uses to recreate the original data set for diagnosis. If the WORK_SIZE is very small in relationship to the input data set, you can end up with too many files on the IBM sites. For example, if you are sending a 100 GB z/OS stand-alone dump and make the work data set size 1 MB, PDUU will attempt to create 100,000 files on the IBM site, which exceeds the IBM limit of 99,999 files. This also causes a lot of delay by starting and stopping the transfer sessions for each file.

If the work buffers are very large in relationship to the input data set size, the amount of overlap time is decreased. When the program first starts, it must fill the "A" work buffer before it starts transmitting any data, which means the copy time is not overlapping with data that needs to be sent. For example, if you were sending a 1 GB dump and you set the work data set size to 1 GB (1,000 MB), there is no overlap between copying the records and sending the work files.

If the input data set is a partitioned data set (PDS/PDSE), PDUU unloads it first into a sequential data set using the IEBCOPY utility.

PDUU typically compresses the input data before it is written to the work buffer; therefore, it is counterproductive to use a tool such as AMATERSE or TRSMMAIN to compress the input data set before using PDUU to send it to the IBM site. If a file is tersed, PDUU will not perform further compression. Overall performance of using AMATERSE with PDUU to send the file takes longer than if an untersed file is compressed and sent using PDUU.

Planning to use PDUU

Use PDUU as the primary utility for sending large volumes of documentation, such as stand-alone dumps, to the IBM site.

When using HTTPS mode, PDUU uses virtual storage to buffer requests instead of DASD data sets. The requesting AMAPDUPL job must have access to enough private virtual storage to satisfy the request. Please refer to the discussion of the WORK_SIZE parameter for details.

If you have previously used the MTFTPS stand-alone tool program (before z/OS V1R13), you must understand the following changes made to package the PDUU utility as part of z/OS:

- The PDUU utility name is AMAPDUPL; however, MTFTPS ships as an alias entry point to AMAPDUPL.
- AMAPDUPL resides in SYS1.MIGLIB (which must be a data set in the LNKST concatenation), so a STEPLIB DDNAME is not necessary to invoke AMAPDUPL.
- AMA messages are described in [z/OS MVS System Messages, Vol 1 \(ABA-AOM\)](#).

Prerequisites and restrictions for PDUU

With APAR OA55959, PDUU now supports two transmission protocols: FTP and HTTPS. Both require proper configuration of the z/OS® Communications Server to use the preferred protocol.

For HTTPS mode, specify HTTPS mode using the USE_HTTPS=Y SYSIN parameter. You will need a valid configured key store containing the necessary certificates to access the IBM documentation sites. This can be specified by the HTTPS_KEYRING or HTTPS_KEYFILE parameters.

For FTP mode, see the topic on [Transferring files using FTP in z/OS Communications Server: IP Configuration Guide](#). The PDUU uses active FTP mode as the default, unless another mode is requested with the corresponding FTP subcommands defined in FTPCMDSD data set.

The PDUU supports the following types of input data sets:

- Members of partitioned data sets (PDS) and partitioned data sets extended (PDSE)
- Large format (DSNTYPE=LARGE) and traditional sequential data sets
- Extended format sequential data sets
- Fixed, variable, and undefined-length, blocked and unblocked, spanned and unspanned record formats (RECFM) = F, FB, FBS, V, VB, VS, VBS, U)
- Data sets with records containing ISO/ANSI or machine code control characters
- Data sets in cylinder-managed space.
- Partitioned data sets (PDS) and partitioned data sets extended (PDSE).

PDUU does not support the following types of input data sets:

- Large block interface (LBI) (no BLKSIZE value).
- VSAM and direct (DSORG=DA) data sets
- Data sets with keys (KEYLEN)
- z/OS UNIX files
- Concatenated data sets of any type

JCL statements for PDUU

The JCL statements for the PDUU are:

SYSPRINT

The data set can be either SYSOUT or a sequential data set. The data set must be RECFM=FB, LRECL=134. For additional details, see [“Prerequisites and restrictions for PDUU” on page 630](#).

SYSUT1

The sequential or partitioned data set to transfer to IBM. For additional details, see [“Prerequisites and restrictions for PDUU”](#) on page 630.

SYSUT2

This data set is optional and can be used only when transferring partitioned data sets (PDS/PDSE). It defines a sequential unload data set for IEBCOPY output produced during the unload operation. If the SYSUT2 statement is omitted the unload data set will be allocated dynamically. This parameter can be used if you want to directly control the allocation of the unload data set, for example, to specify a particular volume or certain amount of volume space. For additional details about usage and allocation parameters of the unload data set see the topic on Unloading (Backing up) Data Sets in [z/OS DFSMSdfp Utilities](#). See also [“Example 9: Using SYSUT2 to allocate an unload data set”](#) on page 639.

SYSIN

A sequential data set that uses the following control statements. The data set must be RECFM=FB, LRECL=80. For additional details, see [“Prerequisites and restrictions for PDUU”](#) on page 630.

PDUU is managed through the following SYSIN statements with these guidelines:

- Use an asterisk (*) in the first column of each comment line to indicate comments.
- Keywords must start in column one.
- Use control statements that are in form VERB=OPERAND.
- Mixed case verbs and operands are allowed.
- The operand starts in the column after the equal sign and goes to the first blank column except TARGET_SYS, DIRECTORY, CIPHER_KEY, ACCOUNT, HTTPS_KEYFILE, HTTPS_KEYRING, HTTPS_KEYSTASH, HTTPS_PROXY, HTTPS_LOCALIPADDR, USERID, and PASSWORD, which can contain blanks.
- Anything after the first blank is ignored except for any operands that can contain blanks. In those cases, do not use blanks from column one to the end of the operand.
- Control statements can be coded on one or more (up to 6) consecutive records. Control statements with operands that allow blanks must not extend beyond column 71, but can continue on the following record in columns 16 through 71. Columns 1 through 15 of the continuation record must be blank. See [“Example 10: Using a multiple record control statement in SYSIN”](#) on page 640. Control statements with operands that do not allow blanks can occupy columns 1 through 80.
- When specifying a control statement twice, the last specification is used.

USE_HTTPS

An optional parameter that when specified with a value of 'Y' enables HTTPS mode and indicates that PDUU use the HTTPS protocol to transfer data to IBM.

Omitting this parameter results in PDUU using the default FTP protocol.

TARGET_SYS

The name of the TCP/IP system to transfer the files to. One through 256 characters, dotted decimal format is allowed, no default value, can not contain blanks, and it must be specified.

For FTP mode, if using a proxy server, this should be the name of the proxy server.

You can include additional FTP command parameters on the TARGET_SYS parameter by using the z/OS UNIX specifications as shown in the topic [FTP command -- Entering the FTP environment in z/OS Communications Server: IP User's Guide and Commands](#). For example, to trace output (-d) and use a specific ftpdata_filename (-f'/'WES.MYFTP.DATA"):

```
TARGET_SYS=-d -f'/'WES.MYFTP.DATA' " testcase.boulder.ibm.com
```

Use the -p parameter to specify an alternate IP stack.

For HTTPS mode, does not allow for the specification of a config file or proxy information through the TARGET_SYS parameter.

USERID

The user ID on the target system that is used to send the files. One through 64 characters, no default value, does not have to be specified, and can contain imbedded blanks.

For FTP mode only, if USERID and PASSWORD are not supplied and NETRCLEVEL=2, the values from the NETRC data set is used for the FTP sessions.

If using a proxy server, this can be the full login to the remote system in the format *userid@remote.system.name*.

PASSWORD

The password for the USERID on the target system. One through 64 characters and the default value is blanks.

For FTP mode only, if using a proxy server, this can be the USERID and PASSWORD for the proxy server in the format *userid@password*.

ACCOUNT

For FTP mode, the account data that is sent when an FTP session is started. One through 64 characters with no default value.

This parameter is ignored for HTTPS mode.

TARGET_DSN

The variable (descriptive) portion for the file names on the target system. One through 64 characters, no default value, and it must be specified.

WORK_DSN

For FTP mode, the prefix for the data set names of work files on the sending system. One through 40 characters, no default value, and it must be specified. The work data sets are large format, sequential, data sets and cannot have the compaction attribute.

Note: Because work files are dynamically allocated with large format and do not support compressed format, if you specify data class for work files with the compaction attribute or N, the following message is issued for all work files:

```
IGD17163I COMPRESSION REQUEST NOT HONORED FOR DATA SET
work_file_dsnname BECAUSE DATA SET CHARACTERISTICS
DO NOT MEET COMPRESSION CRITERIA, ALLOCATION CONTINUES
```

This parameter is ignored for HTTPS mode.

CC_HTTPS | CC_FTP

The number of parallel FTP transfer sessions to use when transmitting the files. One or two decimal digits, the value must be between one and 20, and the default is two. CC_HTTPS is an alias for CC_FTP.

WORK_SIZE | WORK_DSN_SIZE

The maximum size of the work buffer in megabytes. One through four decimal digits. When unspecified, the default is 100. WORK_SIZE is an alias for WORK_DSN_SIZE.

For FTP mode, two DASD data sets per session (CC_FTP) are dynamically allocated with this size.

For HTTPS mode, virtual storage of the size $2 * CC_HTTPS\# * WORK_SIZE\# * 1$ MB is requested.

When choosing values for WORK_SIZE and CC_HTTPS parameters, be aware that these buffers are allocated in 31-bit storage, and this is limited to significantly less than 2 GB. For example, if you have CC_HTTPS=4 and WORK_SIZE=200, AMAPDUPL will attempt to allocate virtual storage of 1600 MB. This may fail with return code 12 and message AMA761E if requested storage is not available. Also, beware that transfers may begin, but fail due to the web enablement toolkit storage requirements. PDUU will also fail with return code 12 and message AMA761E if the web enablement toolkit is unable to obtain storage for PDUU. Consider lowering the CC_HTTPS or WORK_SIZE parameters in this case.

KEEP_WORK

For FTP mode, the parameter to save the work data sets that are dynamically allocated for each FTP session. If you omit the KEEP_WORK parameter, the program does not save the work data sets. Y is the only value for the KEEP_WORK parameter.

Note: Only specify this parameter when debugging a problem.

This parameter is ignored for HTTPS mode.

DATACLAS

For FTP mode, the data class to use when allocating the work files on the sending system. One through eight characters with no default value.

This parameter is ignored for HTTPS mode.

MGMTCLAS

For FTP mode, the management class to use when allocating the work files on the sending system. One through eight characters with no default value.

This parameter is ignored for HTTPS mode.

STORCLAS

For FTP mode, the storage class to use when allocating the work files on the sending system. One through eight characters with no default value.

This parameter is ignored for HTTPS mode.

DIRECTORY

The directory on the target system where the files will be sent with FTP or HTTPS. One through 32 characters, with no default, can contain blanks, and you must specify the directory.

For HTTPS transfer to www.secure.ecurep.ibm.com specify the destination without any leading or trailing slashes, such as 'DIRECTORY=mvs'.

CASE

The CASE id associated with the file and problem. This field must be 11 numeric or uppercase characters. Do not specify when the PMR statement is specified. Example: TS123456789

PMR

The PMR number with which this file is associated. Do not specify when the CASE statement is specified. This field must be 13 numeric or uppercase characters, specify in the form xxxxx.yyy.zzz, and define the variables as:

Field	Explanation	Example
XXXXX	PMR Number	34143
YYY	Branch office	055
ZZZ	IBM Country Code	724

CIPHER_KEY

The encryption key to use for 192-bit triple DES encryption. The 24 characters following CIPHER_KEY= are used as the key. The key can include imbedded and/or trailing blanks. For example, CIPHER_KEY=HERE IS CIPHER KEY IN 24 or CIPHER_KEY=Shortkey. If you do not specify CIPHER_KEY=, no encryption is performed. If you encrypt the data set using CIPHER_KEY, you must provide IBM with the encryption key so they can perform problem diagnosis.

Note: If CIPHER_KEY= is followed by 24 blanks, the file will be encrypted with a key of 24 blanks.

NO_FTP

A value of 'Y' specifies that PDUU compress, optionally encrypt, separate files into parts, and move the part to a local z/OS Unix Systems Services directory, without FTPing the files. PDUU uses TSO services to send data sets to USS directories. When you specify NO_FTP=Y:

- TARGET_SYS, CC_FTP, USERID, and PASSWORD are not required
- CC_FTP settings are ignored and set to 1
- FTPCMDS DD will be ignored
- The system allocates the SYSTSPRT DD to receive messages from TSO.

Once transferred, you can browse the files and extract them from the z/OS Unix Systems Service directory. The file names used depend on the PMR and TARGET_DSN input:

```
PMR.TARGET_DSN.Tdate.MTFTP.F00002 to ...Fnnnnn
```

The size and number of files you end up with depends on the WORK_DSN_SIZE SYSIN control statement and how well the input file compresses.

The date section is a random string based on the time the job is run, and PDUU creates different output file names if the same file is sent at a different time. In addition there will be a small file with suffix F001 containing the control record with information necessary to recreate the file on the receiving end.

Please package and send all generated files to IBM with the file names created. Use the NO_FTP=Y option if you have a closed data center (with no outside internet access). PDUU then prepares files you can send to IBM service on removable media as desired.

NO_FTP is ignored when USE_HTTPS is specified.

HTTPS_KEYRING

Specify a SAF key ring or PKCS #11 token containing certificates necessary to connect to the HTTPS sessions. Of the form:

- SAF key ring name, specified as userid/keyring
- PKCS #11 token, specified as *TOKEN*/token_name

HTTPS_KEYFILE

Specifies a path and file name of the key data base file created by the System SSL gskkyman utility.

HTTPS_KEYSTASH

Specifies the path and file name of the password stash file created by the System SSL gskkyman utility. This option is required when HTTPS_KEYFILE is specified.

HTTPS_PORT

An optional parameter when USE_HTTPS=Y is specified indicating the remote port number to which to connect. The default value is 443.

HTTPS_IPSTACK

An optional parameter when USE_HTTP=Y is specified indicating the local TCP/IP stack name to be used when connecting to the IBM site. 1-8 characters specifications are allowed.

HTTPS_LOCALIPADDR

An optional DD statement that gathers debug information such as messages issued to the SYSPRINT data set and the FTP protocol messages. The data set must be RECFM=FB, LRECL=134.

HTTPS_LOCALPORT

An optional parameter when USE_HTTP=Y is specified indicating the outgoing port number from which the connection is to originate.

HTTPS_PROXY

An optional parameter when USE_HTTP=Y is specified indicating the HTTP proxy to use. Must specify starting with http:// or https://. For example: http://my.proxy.com

HTTPS_PROXYPORT

An optional parameter when USE_HTTP=Y is specified indicating the proxy port to connect to.

HTTPS_PROXYUSERNAME

An optional parameter when USE_HTTP=Y is specified indicating the username to connect to the HTTP proxy. One through 64 characters and can contain embedded blanks. Must specify with HTTPS_PROXYPASSWORD.

HTTPS_PROXYPASSWORD

An optional parameter when USE_HTTP=Y is specified indicating the password to connect to the HTTP proxy. One through 64 characters and can contain embedded blanks. Must specify with HTTPS_PROXYUSERNAME.

HTTPS_VERBOSE

An optional parameter when that when specified with a value of 'Y' and when USE_HTTP=Y indicates to the web enablement toolkit to produce verbose messages for HTTPS transfers. These messages can be helpful in diagnosing connection issues and aiding in IBM problem determination. The HTTPS_VERBOSE_DD can be used to change where these messages are sent.

HTTPS_VERBOSE_DD

An optional parameter to specify a 1-8 character DD name that takes effect when the HTTPS_VERBOSE=Y parameter is used. The DD name indicates where the web enablement toolkit is to place the verbose messages. This must meet the requirements as described in *HTTP/HTTPS enabler options and values* in the HWTH_OPT_VERBOSE_OUTPUT option. The default is SYSPRINT when not specified.

DEBUG

An optional DD statement that gathers debug information such as messages issued to the SYSPRINT data set and the FTP protocol messages. The data set must be RECFM=FB, LRECL=134.

FTPCMDS

For FTP mode, an optional DD statement that provides additional flexibility for traversing firewall or proxy servers. When this DD statement is provided, after the initial USERID and PASSWORD are sent, the specified sequential data set is read by the application and the commands contained in the data set are included as FTP commands. The data set must be RECFM=FB and LRECL=80.

This DD is ignored for HTTPS mode.

SYSTSPRT

An optional DD statement that shows messages from TSO operations when you have NO_FTP=Y specified. If you receive message AMA778I with a return code of X'04' and function code of X'C', add this DD statement to your JCL to receive messages and rerun if necessary to receive messages.

JCL examples for PDUU

Use the following JCL examples as a guideline for creating your own JCL. Consider storing your user ID and password in a separate concatenated data set. Doing so provides added security because the user ID and password are not directly in the JCL. It is also makes it much easier to change the user ID and password across multiple jobs.

You can use some of the JCL examples as a starting point to traverse a firewall or proxy server. There are very few common characteristics for firewall or proxy servers with local customization. If you are able to traverse the firewall or proxy server with a plain FTP statement, modifications to the parameters USERID, PASSWORD, ACCOUNT, and TARGET_SYS, in conjunction with commands in the FTPCMDS data set, the ftp_data file, or both can permit the z/OS PDUU to traverse your firewall or proxy server.

Example 1: Simple FTP connection

The JCL example in [Figure 216 on page 636](#) invokes the AMAPDUPL program to transfer file H44IPCS.WESSAMP.TRKS055K to the testcase.boulder.ibm.com system as a set of work files stored in /toibm/mvs with the shared prefix 12345.123.123.wessamp.bigfile. Each of the three work files is 500 MB.

```
//FTP      EXEC PGM=AMAPDUPL
//SYSUDUMP DD SYSOUT=*
//SYSPRINT DD SYSOUT=*
//SYSUT1   DD DISP=SHR,DSN=H44IPCS.WESSAMP.TRKS055K
//SYSIN    DD *
USERID=anonymous
PASSWORD=anonymous
TARGET_SYS=testcase.boulder.ibm.com
TARGET_DSN=wessamp.bigfile
WORK_DSN=wes.ftpout
CC_FTP=03
WORK_DSN_SIZE=500
DIRECTORY=/toibm/mvs/
PMR=12345.123.123
//
```

Figure 216. Simple FTP connection

Example 2: FTP connection using a proxy server

In Figure 217 on page 636, the USERID control statement has the format `user@hostname`, where `hostname` is the name of the TCP/IP system to transfer files to, and `user` is the user name on the `hostname` system. The PASSWORD control statement has format `proxyuser@proxypass`, where `proxyuser` is the user name on the proxy server and `proxypass` is the user password on the proxy server. TARGET_SYS is the name of the TCP/IP proxy server.

```
//FTP      EXEC PGM=AMAPDUPL
//SYSUDUMP DD SYSOUT=*
//SYSPRINT DD SYSOUT=*
//SYSUT1   DD DISP=SHR,DSN=H44IPCS.WESSAMP.TRKS055K
//SYSIN    DD *
USERID=anonymous@testcase.boulder.ibm.com
PASSWORD=proxyuser@proxypass
TARGET_SYS=your.proxy.server.name
TARGET_DSN=wessamp.bigfile
WORK_DSN=wes.ftpout
CC_FTP=03
WORK_DSN_SIZE=500
DIRECTORY=/toibm/mvs/
PMR=12345.123.123
//
```

Figure 217. FTP connection using a proxy server

Example 3: FTP connection using a proxy server with proxy user ID

In Figure 218 on page 637, the USERID control statement has format `user@proxyuser@hostname`, where `hostname` is the name of the TCP/IP system to transfer the files to, `user` is the user name on the `hostname` system, and `proxyuser` is the user name on the proxy server. The PASSWORD control statement has format `proxyuser@proxypass`, where `proxyuser` is the user name on the proxy server and `proxypass` is the user's password on the proxy server. TARGET_SYS is the name of the TCP/IP proxy server.


```
//FTP      EXEC PGM=AMAPDUPL
//SYSUDUMP DD SYSOUT=*
//SYSPRINT DD SYSOUT=*
//SYSUT1   DD DISP=SHR,DSN=H44IPCS.WESSAMP.TRKS055K
//SYSIN    DD *
USERID=anonymous@proxyuser@testcase.boulder.ibm.com
PASSWORD=proxyuser@proxypass
TARGET_SYS=proxy.server.name
TARGET_DSN=wessamp.bigfile
WORK_DSN=wes.ftpout
CC_FTP=03
WORK_DSN_SIZE=500
DIRECTORY=/toibm/mvs/
PMR=99999.000.000
//
```

Figure 218. FTP with a proxy user ID

Example 4: Using a proxy server with the FTPCMDS DD statement

In Figure 219 on page 637, the USERID control statement has format `proxyuser@hostname`, where `hostname` is the name of the TCP/IP system to transfer the files to, and `proxyuser` is the user name on the proxy server. The PASSWORD control statement defines the user password on the proxy server. The data set name `WES.FTPCMDS.DATA` contains an additional user command with an anonymous user name and password on the hostname system.

```
//FTP      EXEC PGM=AMAPDUPL
//SYSUDUMP DD SYSOUT=*
//SYSPRINT DD SYSOUT=*
//SYSUT1   DD DISP=SHR,DSN=H44IPCS.WESSAMP.TRKS055K
//FTPCMDS  DD DISP=SHR,DSN=WES.FTPCMDS.DATA
//SYSIN    DD *
USERID=proxyid@testcase.boulder.ibm.com
PASSWORD=proxypass
TARGET_SYS=proxy.server.name
TARGET_DSN=SVCD
WORK_DSN=HLQ.FTPOUT
CC_FTP=03
WORK_DSN_SIZE=500
DIRECTORY=/toibm/mvs/
PMR=99999.000.000
//
```

Figure 219. FTP using the FTPCMDS DD statement

The example in Figure 220 on page 637 shows the typical format of the FTPCMDS data set.

```
user anonymous pw userid@company.com
```

Figure 220. FTPCMDS data set example

Example 5: Using a proxy server with a port specification on the TARGET_SYS parameter

The example in Figure 221 on page 638 uses a proxy server with a port specification of 2121 on the TARGET_SYS parameter and inline FTPCMDS DD statement. This example is similar to the previous one, the only difference are the FTPCMDS is an in-stream data set and the port specification is included on the TARGET_SYS parameter.

```
//FTP EXEC PGM=AMAPDUPL
//SYSUDUMP DD SYSOUT=*
//SYSPRINT DD SYSOUT=*
//SYSUT1 DD DISP=SHR,DSN=H44IPCS.WESSAMP.TRKS055K
//FTPCMDS DD *
user anonymous pw userid@company.com
//SYSIN DD *
USERID=proxyuser@testcase.boulder.ibm.com
PASSWORD=proxypass
TARGET_SYS=proxy.server.name 2121
TARGET_DSN=SVCD
WORK_DSN=HLQ.FTPOUT
CC_FTP=03
WORK_DSN_SIZE=500
DIRECTORY=/toibm/mvs/
PMR=99999.000.000
CIPHER_KEY=PMR99999sad
//
```

Figure 221. FTP specifying port 2121 on TARGET_SYS

Example 6: Forcing PASSIVE mode using the FTPCMDS inline DD statement

Figure 222 on page 638 shows the FTP connection set up from the FTP client to the FTP server using the FTP locsite fwfriendly command.

The FWfriendly parameter specifies that the FTP client is firewall-friendly. For additional details, see the topic about [LOCSite subcommand--Specify site information to the local host in z/OS Communications Server: IP User's Guide and Commands](#).

Note: When the FTP server has an IPv6 address, data connections are always set up from the FTP client to the FTP server without reference to the FWfriendly setting.

```
//FTP EXEC PGM=AMAPDUPL
//SYSUDUMP DD SYSOUT=*
//SYSPRINT DD SYSOUT=*
//SYSUT1 DD DISP=SHR,DSN=H44IPCS.WESSAMP.TRKS055K
//FTPCMDS DD *
LOCSite Fwfriendly
//SYSIN DD *
USERID=proxyuser@testcase.boulder.ibm.com
PASSWORD=proxypass
TARGET_SYS=proxy.server.name
TARGET_DSN=SVCD
WORK_DSN=HLQ.FTPOUT
CC_FTP=03
WORK_DSN_SIZE=500
DIRECTORY=/toibm/mvs/
PMR=99999.000.000
CIPHER_KEY=PMR99999sad
//
```

Figure 222. FTP forcing PASSIVE mode

Example 7: Using a userid.NETRC data set

The example in Figure 223 on page 639 uses the proxy login and password stored in the userid.NETRC data set (you can submit this as a surrogate job where the userid.NETRC data set is invisible to the job originator). Use of the userid.NETRC data set requires NETRCLEVEL=2, which is set in the FTP.DATA data set. Using the -f parameter on TARGET_SYS control statement specifies which FTP.DATA data set to use.

- Find information about the use of the NETRC data set in [z/OS Communications Server: IP User's Guide and Commands](#).
- Find information about the use of the FTP.DATA data set in [z/OS Communications Server: IP Configuration Reference](#).

```
//FTP EXEC PGM=AMAPDUPL
//SYSUDUMP DD SYSOUT=*
//SYSPRINT DD SYSOUT=*
//SYSUT1 DD DISP=SHR,DSN=H44IPCS.WESSAMP.TRKS055K
//SYSIN DD *
TARGET_SYS=-f"'WES.MYFTP.DATA'" testcase.boulder.ibm.com
PMR=99999.999.000
DIRECTORY=/toibm/mvs/
TARGET_DSN=wes.gb01tst
work_dsn=wes.ftpout
cc_ftp=03
WORK_DSN_SIZE=500
//

For this instance, the userid.NETRC data set consists of one line:
machine testcase.boulder.ibm.com login anonymous password ibmusi@ibm.com

The FTP.DATA data set contains:
;*****
;*
NETRCLEVEL 2
;*
```

Figure 223. FTP using a userid.NETRC data set

Example 8: Using the DEBUG statement

The example in Figure 224 on page 639 adds the DEBUG DD statement, which creates a data set that contains the message data, as described in [“DEBUG” on page 635](#).

```
//FTP EXEC PGM=AMAPDUPL
//SENDSTP EXEC PGM=AMAPDUPL
//SYSPRINT DD SYSOUT=*
//DEBUG DD DSN=PDUU.DEBUG,DISP=(,CATLG),
// UNIT=SYSALLDA,SPACE=(CYL,(1,1),RLSE)
//SYSUT1 DD DISP=SHR,DSN=H44IPCS.WESSAMP.TRKS055K
//SYSIN DD *
USERID=anonymous
PASSWORD=anonymous
TARGET_SYS=testcase.boulder.ibm.com
TARGET_DSN=wessamp.bigfile
WORK_DSN=wes.ftpout
WORK_DSN_SIZE=500
DIRECTORY=/toibm/mvs
PMR=12345.123.123
//
```

Figure 224. FTP connection with the DEBUG DD statement

Example 9: Using SYSUT2 to allocate an unload data set

The JCL example in [Figure 225 on page 640](#) invokes the AMAPDUPL program to transfer partitioned data set H44IPCS.PDS.DATA to the testcase.boulder.ibm.com. The optional SYSUT2 statement is used to allocate an unload sequential data set for the IEBCOPY utility invoked by the PDUU.

```

//FTP EXEC PGM=AMAPDUPL
//SYSUDUMP DD SYSOUT=*
//SYSPRINT DD SYSOUT=*
//SYSUT1 DD DSN=H44IPCS.PDS.DATA,DISP=SHR
//SYSUT2 DD DSN=H44IPCS.UNLOAD.DASD,DISP=(NEW,CATLG),
// DCB=(RECFM=VS),
// SPACE=(CYL,(1,1),RLSE),UNIT=SYSDA
//SYSIN DD *
USERID=anonymous
PASSWORD=anonymous
TARGET_SYS=testcase.boulder.ibm.com
TARGET_DSN=wessamp.bigfile
WORK_DSN=wes.ftpout
CC_FTP=03
WORK_DSN_SIZE=500
DIRECTORY=/toibm/mvs/
PMR=12345.123.123
//

```

Figure 225. Using SYSUT2 statement for allocating an unload data set

Example 10: Using a multiple record control statement in SYSIN

The following JCL example in Figure 226 on page 640 invokes the AMAPDUPL program with FTP.DATA data set specified by the `-f` parameter on TARGET_SYS control statement coded on 3 consecutive SYSIN records.

```

-----1-----2-----3-----4-----5-----6-----7--
//FTP EXEC PGM=AMAPDUPL
//SYSUDUMP DD SYSOUT=*
//SYSPRINT DD SYSOUT=*
//SYSUT1 DD DSN=H44IPCS.SEQ.DATA,DISP=SHR
//SYSIN DD *
USERID=anonymous
PASSWORD=anonymous
TARGET_SYS=-d -f /u/directoryForFtpData/subdirectoryForFtpData1/subdire
          ctoryForFtpData2/subdirectoryForFtpData3/subdirectoryFor
          FtpData4/ftpDataFile testcase.boulder.ibm.com

TARGET_DSN=wessamp.bigfile
WORK_DSN=wes.ftpout
CC_FTP=03
WORK_DSN_SIZE=500
DIRECTORY=/toibm/mvs/
PMR=12345.123.123
//

```

Figure 226. Using a multiple record control statement

Example 11: Using the NO_FTP option

The following JCL invokes the AMAPDUPL program to compress and separate the file into chunks that can be brought to IBM for documentation. File H44IPCS.WESSAMP.TRKS055K will be compressed and stored into several files at:

```

/u/nickj/pduu/12345.123.123.WESSAMP.BIGFILE.Txxxxx.MTFTP.Fnnnnn

```

```
//FTP      EXEC PGM=AMAPDUPL
//SYSPRINT DD  SYSOUT=*
//SYSTSPRT DD  SYSOUT=*
//SYSUT1   DD  DISP=SHR,DSN=H44IPCS.WESSAMP.TRKS055K
//SYSIN    DD  *
NO_FTP=Y
TARGET_DSN=WESSAMP.BIGFILE
WORK_DSN=WES.FTPOUT
WORK_DSN_SIZE=500
DIRECTORY=/u/nickj/pduu/
PMR=12345.123.123
```

Figure 227. Using the NO_FTP option

The following files will be stored on your local systems directory:

```
/u/nickj/pduu/:
12345.123.123.WESSAMP.BIGFILE.TE3246.MTFTP.F00002
12345.123.123.WESSAMP.BIGFILE.TE3246.MTFTP.F00003
12345.123.123.WESSAMP.BIGFILE.TE3246.MTFTP.F00004
. . .
12345.123.123.WESSAMP.BIGFILE.TE3246.MTFTP.F001
```

Example 12: Using a proxy server with multiple FTPCMDS DD statements

The example in [Figure 228 on page 641](#) uses a proxy server with multiple statements in an inline FTPCMDS DD. In this example, FTPCMDS DD runs the USER and PASS commands, and the USERID and PASSWORD SYSIN statements are omitted. The user also specifies PASSIVE mode.

```
//FTP      EXEC PGM=AMAPDUPL
//SYSUDUMP DD  SYSOUT=*
//SYSPRINT DD  SYSOUT=*
//SYSUT1   DD  DISP=SHR,DSN=H44IPCS.WESSAMP.TRKS055K
//FTPCMDS DD  *
USER 'anonymous@testcase.boulder.ibm.com proxyuser'
PASS userid@company.com
ACCT proxypassword
LOCSite FwFriendly
//SYSIN    DD  *
TARGET_SYS=proxy.server.name
TARGET_DSN=SVCD
WORK_DSN=HLQ.FTPOUT
CC_FTP=03
WORK_DSN_SIZE=500
DIRECTORY=/toibm/mvs/
PMR=99999.000.000
CIPHER_KEY=PMR99999sd
//
```

Figure 228. FTP specifying multiple statements in an inline FTPCMDS DD

Example 13: Simple HTTPS connection to testcase

The JCL example in [Figure 229 on page 642](#) invokes the AMAPDUPL program to transfer file H44IPCS.WESSAMP.TRKS055K to the testcase.boulder.ibm.com system as a set of work files stored in /toibm/mvs with the shared prefix 12345.123.123.wessamp.bigfile. Each of the three work files will be 100 MB. This connection uses the RACF keyring for user TSOUSER named 'pduu'.

PDS entry TSOUSER.FTPINFO(TESTCASE) contains the USERID and PASSWORD in a RACF protected and encrypted PDS with contents described in [Figure 230 on page 642](#).

```
//FTP      EXEC PGM=AMAPDUPL
//SYSUDUMP DD SYSOUT=*
//SYSPRINT DD SYSOUT=*
//DEBUG    DD SYSOUT=*
//SYSUT1   DD DISP=SHR,DSN=H44IPCS.WESSAMP.TRKS055K
//SYSIN    DD DISP=SHR,DSN=TSOUSER.FTPINFO(TESTCASE)
//          DD *
TARGET_SYS=testcase.boulder.ibm.com
TARGET_DSN=wessamp.bigfile
CC_HTTPS=03
WORK_SIZE=100
DIRECTORY=/toibm/mvs/
PMR=12345.123.123
USE_HTTPS=Y
HTTPS_KEYRING=TSOUSER/pduu
```

Figure 229. Simple HTTPS connection to testcase

```
USERID=user@domain.com
PASSWORD=userpassword
```

Figure 230. Simple HTTPS connection contents of TSOUSER.FTPINFO(TESTCASE)

Example 14: Simple HTTPS connection with verbose HTTPS messages

The JCL example in [Figure 231 on page 642](#) is similar to “[Example 13: Simple HTTPS connection to testcase](#)” on [page 641](#) but puts verbose messages into the HTTPDEBG DD. This connection uses a key database file created by the gskkyman and a matching key database password stored in a key stash.

```
//FTP      EXEC PGM=AMAPDUPL
//SYSUDUMP DD SYSOUT=*
//SYSPRINT DD SYSOUT=*
//DEBUG    DD SYSOUT=*
//SYSUT1   DD DISP=SHR,DSN=H44IPCS.WESSAMP.TRKS055K
//HTTPDEBG DD DISP=(NEW,CATLG,KEEP),
//          DSN=TSOUSER.HTTPDEBG,DCB=(RECFM=V,DSORG=PS),
//          SPACE=(CYL,(1,1))
//SYSIN    DD DISP=SHR,DSN=TSOUSER.FTPINFO(TESTCASE)
//          DD *
TARGET_SYS=testcase.boulder.ibm.com
TARGET_DSN=wessamp.bigfile
CC_FTP=03
WORK_SIZE=50
DIRECTORY=/toibm/mvs/
PMR=12345.123.123
USE_HTTPS=Y
HTTPS_KEYFILE=/etc/websiv1/mykeys.kdb
HTTPS_KEYSTASH=/etc/websiv1/mykeys.sth
HTTPS_VERBOSE=Y
HTTPS_VERBOSE_DD=HTTPDEBG
```

Figure 231. FTP specifying multiple statements in an inline FTPCMDS DD

Example 15: Simple HTTPS connection to Ecurep

The JCL example in [Figure 232 on page 643](#) invokes the AMAPDUPL program to transfer file H44IPCS.WESSAMP.TRKS055K to the testcase.boulder.ibm.com system as a set of work files. These will be sent to ecurep and directed to the mvs directories with the shared prefix 12345.123.123.wessamp.bigfile. Each of the three work files will be 100 MB. This connection uses the RACF keyring for user TSOUSER named 'pduu'.

PDS entry TSOUSER.FTPINFO(TESTCASE) contains the USERID and PASSWORD in a RACF protected and encrypted PDS with contents described in [Figure 233 on page 643](#).

```

///FTP      EXEC PGM=AMAPDUPL
//SYSUDUMP DD  SYSOUT=*
//SYSPRINT DD  SYSOUT=*
//DEBUG     DD  SYSOUT=*
//SYSUT1    DD  DISP=SHR,DSN=H44IPCS.WESSAMP.TRKS055K
//SYSIN     DD  DISP=SHR,DSN=TSOUSER.FTPINFO(ECUREP)
//          DD  *
TARGET_SYS=www.secure.ecurep.ibm.com
TARGET_DSN=wessamp.bigfile
CC_FTP=03
WORK_SIZE=100
DIRECTORY=mvs
PMR=12345.123.123
USE_HTTPS=Y
HTTPS_KEYRING=TSOUSER/pduu

```

Figure 232. Simple HTTPS connection to Ecurep

```

USERID=user@domain.com
PASSWORD=userpassword

```

Figure 233. Simple HTTPS connection contents of TSOUSER.FTPINFO(ECUREP)

Return codes for PDUU

Upon completion, PDUU places one of the return codes listed in [Table 72 on page 643](#) in general purpose register (GPR) 15.

Table 72. Return codes for z/OS Problem Documentation Upload Utility

Return Code	Explanation
0	Successful completion
4	Potential successful completion. If not, investigate messages.
8	Invalid parameters in control statement
10	Unsupported data set format
12	Storage obtain failure
16	Required input parameters are missing
20	Invalid input data set specified
24	Severe error occurred during dictionary building
28	Severe error occurred during file open
32	Severe error occurred during compression process
36	Error in FTP or HTTPS operation
40	Severe error during IKJTSOEV operation for NO_FTP=Y invocations, see message AMA777I
44	Severe error during IKJEFTSR operation for NO_FTP=Y invocations, see message AMA778I and SYSTSPRT DD
64	Severe error in file operation

Table 72. Return codes for z/OS Problem Documentation Upload Utility (continued)

Return Code	Explanation
68	Error unloading a PDS or PDSE
69	Failure to attach a subtask
70	Failure in pause element service
71	Failure in BPX1SDD service
98	System or user abend occurred during subtask operation
99	System or user abend occurred

Descriptor code

N/A

AMA731I**EFFECTIVE THROUGHPUT = *thrput* BYTES/SECOND****Explanation**

This message displays the throughput value FTP channel used to transmit the data. In the message text:

thrput

The throughput value.

System action

None.

System programmer response

None.

User response

None.

Problem determination

None.

Source

PDUU

Module

AMAPDUPL

Routing code

N/A

Descriptor code

N/A

AMA732I**START *number type* FTP SESSION****Explanation**

This message indicates the beginning of FTP or HTTPS session. In the message text:

number

The order number of starting FTP or HTTPS session.

type

The type of session, FTP or HTTPS session.

System action

None.

User response

None.

Problem determination

None.

Source

PDUU

Module

AMAPDUPL

Routing code

N/A

Descriptor code

N/A

AMA733I **SEND FILE A|B *filenumber*****Explanation**

This message is displayed when corresponding work file is sent. In the message text:

filenumber

The number of the work file to send.

System action

None.

System programmer response

None.

User response

None.

Problem determination

None.

Source

PDUU

Module

AMAPDUPL

Routing code

N/A

Descriptor code

N/A

AMA734I **UNABLE TO OPEN SYSIN DATASET**

Explanation

The Problem Documentation Upload Utility (PDUU) encountered an error while attempting to open the specified SYSIN data set.

System action

PDUU processing ends.

System programmer response

None.

User response

None.

Problem determination

None.

Source

PDUU

Module

AMAPDUPL

Routing code

N/A

Descriptor code

N/A

AMA735I**SYSIN STATEMENTS:****Explanation**

This message indicates the start of reading and interpreting data from SYSIN data set. All SYSIN statements except PASSWORD get printed to corresponding SYSPRINT data set

System action

None.

System programmer response

None.

User response

None.

Problem determination

None.

Source

PDUU

Module

AMAPDUPL

Routing code

N/A

Descriptor code

N/A

AMA736E**THE LENGTH OF ACCOUNT INFORMATION SPECIFIED IS NOT
BETWEEN 1 AND 64****Explanation**

The Problem Documentation Upload Utility (PDUU) encountered an error while reading and interpreting SYSIN data set. The ACCOUNT is too long.

System action

None.

System programmer response

None.

User response

Correct ACCOUNT parameter in SYSIN data set and rerun PDUU.

Problem determination

Return code 8 is issued.

Source

PDUU

Module

AMAPDUPL

Routing code

N/A

Descriptor code

N/A

AMA737E**INCORRECT SYSIN CONTROL STATEMENT, EQUAL SIGN OMITTED:
*statement***

Explanation

The Problem Documentation Upload Utility (PDUU) encountered an error while reading and interpreting SYSIN data set. No equal sign was found in the control statement.

statement

The incorrect control statement.

System action

None.

System programmer response

None.

User response

None.

Problem determination

None.

Source

PDUU

Module

AMAPDUPL

Routing code

N/A

Descriptor code

N/A

AMA738E**THE LENGTH OF USERID SPECIFIED IS NOT BETWEEN 1 AND 64
CHARACTERS****Explanation**

The Problem Documentation Upload Utility (PDUU) encountered an error while reading and interpreting SYSIN data set. USERID name is too long.

System action

PDUU processing ends.

System programmer response

None.

User response

Correct USERID in SYSIN data set and rerun PDUU.

Problem determination

None.

Source

PDUU

Module

AMAPDUPL

Routing code

N/A

Descriptor code

N/A

AMA739E**THE LENGTH OF PASSWORD SPECIFIED IS NOT BETWEEN 1 AND 64
CHARACTERS****Explanation**

The Problem Documentation Upload Utility (PDUU) encountered an error while reading and interpreting SYSIN data set. PASSWORD is too long

System action

PDUU processing ends.

System programmer response

None.

User response

Correct password in SYSIN data set and rerun PDUU

Problem determination

None.

Source

PDUU

Module

AMAPDUPL

Routing code

N/A

Descriptor code

N/A

AMA740E**THE LENGTH OF TARGET SYSTEM NAME SPECIFIED IS NOT BETWEEN 1
AND 256 CHARACTERS**

Explanation

The Problem Documentation Upload Utility (PDUU) encountered an error while reading and interpreting SYSIN data set. TARGET_SYS name is too long.

System action

PDUU processing ends.

System programmer response

None.

User response

Correct TARGET_SYS statement in SYSIN data set and rerun PDUU.

Problem determination

None.

Source

PDUU

Module

AMAPDUPL

Routing code

N/A

Descriptor code

N/A

AMA741E**THE LENGTH OF TARGET_DSN SPECIFIED IS NOT BETWEEN 1 AND 50 CHARACTERS****Explanation**

The Problem Documentation Upload Utility (PDUU) encountered an error while reading and interpreting SYSIN data set. TARGET_DSN name is too long.

System action

PDUU processing ends.

System programmer response

None.

User response

Correct TARGET_DSN statement in SYSIN data set and rerun PDUU.

Problem determination

None.

Source

PDUU

Module

AMAPDUPL

Routing code

N/A

Descriptor code

N/A

AMA742E THE LENGTH OF DIRECTORY NAME SPECIFIED IS NOT BETWEEN 1 AND 32 CHARACTERS**Explanation**

The Problem Documentation Upload Utility (PDUU) encountered an error while reading and interpreting SYSIN data set. Directory name is too long.

System action

PDUU processing ends.

System programmer response

None.

User response

Correct DIRECTORY statement in SYSIN data set and rerun PDUU. For the DIRECTORY data set format, see the topic about [JCL statements for z/OS Problem Documentation Upload Utility in *z/OS MVS Diagnosis: Tools and Service Aids*](#).

Problem determination

None.

Source

PDUU

Module

AMAPDUPL

Routing code

N/A

Descriptor code

N/A

AMA743E THE LENGTH OF WORK DATASET PREFIX NAME SPECIFIED IS NOT BETWEEN 1 AND 40 CHARACTERS

Explanation

The Problem Documentation Upload Utility (PDUU) encountered an error while reading and interpreting SYSIN data set. WORK_DSN prefix name is too long.

System action

PDUU processing ends.

System programmer response

None.

User response

Correct WORK_DSN statement in SYSIN data set and rerun PDUU. For the correct WORK_DSN statement in SYSIN, see the topic about [JCL statements for z/OS Problem Documentation Upload Utility in z/OS MVS Diagnosis: Tools and Service Aids](#).

Problem determination

This message is accompanied by return code 28. See the topic about [Return codes for the z/OS Problem Documentation Upload Utility in z/OS MVS Diagnosis: Tools and Service Aids](#).

Source

PDUU

Module

AMAPDUPL

Routing code

N/A

Descriptor code

N/A

AMA744E**WORK DATASET NAME HAS INCORRECT CHARACTERS****Explanation**

The Problem Documentation Upload Utility (PDUU) encountered an error while reading and interpreting SYSIN data set. Invalid characters were found in WORK_DSN name.

System action

PDUU processing ends.

System programmer response

None.

User response

Correct WORK_DSN in SYSIN data set and rerun PDUU. For the correct WORK_DSN statement in SYSIN, see the topic about [JCL statements for z/OS Problem Documentation Upload Utility in z/OS MVS Diagnosis: Tools and Service Aids](#).

Problem determination

None.

Source

PDUU

Module

AMAPDUPL

Routing code

N/A

Descriptor code

N/A

AMA745E THE LENGTH OF DATACLAS NAME SPECIFIED IS NOT BETWEEN 1 AND 8 CHARACTERS

Explanation

The Problem Documentation Upload Utility (PDUU) encountered an error while reading and interpreting SYSIN data set. DataClas name is too long.

System action

PDUU processing ends.

System programmer response

None.

User response

Correct DATACLAS statement in SYSIN data set and rerun PDUU. For the correct DATACLAS statement in SYSIN, see the topic about [JCL statements for z/OS Problem Documentation Upload Utility in z/OS MVS Diagnosis: Tools and Service Aids](#).

Problem determination

This message is accompanied by return code 28. See the topic about [Return codes for the z/OS Problem Documentation Upload Utility in z/OS MVS Diagnosis: Tools and Service Aids](#).

Source

PDUU

Module

AMAPDUPL

Routing code

N/A

Descriptor code

N/A

AMA746E**THE LENGTH OF STORCLAS NAME SPECIFIED IS NOT BETWEEN 1 AND 8 CHARACTERS****Explanation**

The Problem Documentation Upload Utility (PDUU) encountered an error while reading and interpreting SYSIN data set. The STORCLAS name is too long.

System action

PDUU processing ends.

System programmer response

None.

User response

Correct STORCLAS statement in SYSIN data set and rerun PDUU. For the correct STORCLAS statement in SYSIN, see the topic about [JCL statements for z/OS Problem Documentation Upload Utility in z/OS MVS Diagnosis: Tools and Service Aids](#).

Problem determination

This message is accompanied by return code 8 documented in the topic about [Return codes for the z/OS Problem Documentation Upload Utility in z/OS MVS Diagnosis: Tools and Service Aids](#).

Source

PDUU

Module

AMAPDUPL

Routing code

N/A

Descriptor code

N/A

AMA747I**PARAMETER *parameter* MUST ONLY CONTAIN NUMBERS****Explanation**

The Problem Documentation Upload Utility (PDUU) encountered an error while reading and interpreting SYSIN data set. The value specified for the *parameter* is invalid.

In the message text:

parameter

The parameter that has characters other than numbers in it.

System action

PDUU processing ends.

System programmer response

None.

User response

Correct the *parameter* statement in SYSIN data set and rerun PDUU. For the correct values for the *parameter* statement in SYSIN, see the topic about [JCL statements for z/OS Problem Documentation Upload Utility in z/OS MVS Diagnosis: Tools and Service Aids](#).

Problem determination

This message is accompanied by return code 8 documented in the topic about [Return codes for the z/OS Problem Documentation Upload Utility in z/OS MVS Diagnosis: Tools and Service Aids](#).

Source

PDUU

Module

AMAPDUPL

Routing code

N/A

Descriptor code

N/A

AMA748I

THE SPECIFIED *parameter* VALUE IS NOT BETWEEN *lowValue* AND *highValue*

Explanation

The Problem Documentation Upload Utility (PDUU) encountered an error while reading and interpreting SYSIN data set. The specified *parameter* value is not within allowable numerical interval.

In the message text:

parameter

The parameter that has an incorrect numeric value.

lowValue

The minimum numeric value for the *parameter*.

highValue

The maximum numeric value for the *parameter*.

System action

PDUU processing ends.

System programmer response

None.

User response

Correct the *parameter* statement in SYSIN data set and rerun PDUU. For more information, see the topic about [JCL statements for z/OS Problem Documentation Upload Utility in z/OS MVS Diagnosis: Tools and Service Aids](#).

Problem determination

This message is accompanied by return code 8 documented in the topic about [Return codes for the z/OS Problem Documentation Upload Utility in z/OS MVS Diagnosis: Tools and Service Aids](#).

Source

PDUU

Module

AMAPDUPL

Routing code

N/A

Descriptor code

N/A

AMA749E**SEVERE ERROR OCCURRED WHILE WRITING TO WORK FILES****Explanation**

The Problem Documentation Upload Utility (PDUU) encountered an error while writing to one of the work files.

System action

PDUU processing ends.

System programmer response

None.

User response

Check PDUU job output and SYSIN control statements then rerun PDUU. If problem occurs again contact your system programmer. For more information, see the topic about [JCL statements for z/OS Problem Documentation Upload Utility in *z/OS MVS Diagnosis: Tools and Service Aids*](#).

Problem determination

This message is accompanied by return code 8 documented in the topic about [Return codes for the z/OS Problem Documentation Upload Utility in *z/OS MVS Diagnosis: Tools and Service Aids*](#).

Source

PDUU

Module

AMAPDUPL

Routing code

N/A

Descriptor code

N/A

AMA750E**INCORRECT NUMERICS IN WORK_DSN_SIZE CONTROL STATEMENT**

Explanation

The Problem Documentation Upload Utility (PDUU) encountered an error while reading and interpreting SYSIN data set. The specified WORK_DSN_SIZE parameter contains invalid numeric characters.

System action

PDUU processing ends.

System programmer response

None.

User response

Correct the CC_FTP statement in SYSIN data set and rerun PDUU. For more information, see the topic about [JCL statements for z/OS Problem Documentation Upload Utility in *z/OS MVS Diagnosis: Tools and Service Aids*](#).

Problem determination

This message is accompanied by return code 8 documented in the topic about [Return codes for the z/OS Problem Documentation Upload Utility in *z/OS MVS Diagnosis: Tools and Service Aids*](#).

Source

PDUU

Module

AMAPDUPL

Routing code

N/A

Descriptor code

N/A

AMA751E**THE WORK DATASET SIZE IS NOT BETWEEN 1 AND 9999****Explanation**

The Problem Documentation Upload Utility (PDUU) encountered an error while reading and interpreting SYSIN data set. Specified WORK_DSN_SIZE parameter is not within allowable numerical interval. PDUU work files must have at least 1 MG but no more than 9999 MG.

System action

PDUU processing ends.

System programmer response

None.

User response

Correct the WORK_DSN_SIZE statement in SYSIN data set and rerun PDUU. For more information, see the topic about [JCL statements for z/OS Problem Documentation Upload Utility in *z/OS MVS Diagnosis: Tools and Service Aids*](#).

Problem determination

This message is accompanied by return code 8 documented in the topic about [Return codes for the z/OS Problem Documentation Upload Utility in *z/OS MVS Diagnosis: Tools and Service Aids*](#).

Source

PDUU

Module

AMAPDUPL

Routing code

N/A

Descriptor code

N/A

AMA752E**KEEP_WORK OPERAND WAS NOT RECOGNIZED****Explanation**

The Problem Documentation Upload Utility (PDUU) encountered an error while reading and interpreting SYSIN data set. The KEEP_WORK operand must either have a Y value specified or not be specified at all.

System action

None.

System programmer response

None.

User response

Correct KEEP_WORK statement in SYSIN data set and rerun PDUU. For more information, see the topic about [JCL statements for z/OS Problem Documentation Upload Utility in *z/OS MVS Diagnosis: Tools and Service Aids*](#).

Problem determination

This message is accompanied by return code 8 documented in the topic about [Return codes for the z/OS Problem Documentation Upload Utility in *z/OS MVS Diagnosis: Tools and Service Aids*](#).

Source

PDUU

Module

AMAPDUPL

Routing code

N/A

Descriptor code

N/A

AMA753I**THE LENGTH OF PARAMETER *parameter* IS NOT BETWEEN *lowValue* AND *highValue* CHARACTERS****Explanation**

The Problem Documentation Upload Utility (PDUU) encountered an error while reading and interpreting SYSIN data set. *parameter* is either too long or too short.

In the message text:

parameter

The parameter that is either too long or too short.

lowValue

The minimum numeric value for the *parameter*.

highValue

The maximum numeric value for the *parameter*.

System action

None.

System programmer response

None.

User response

Correct *parameter* statement in SYSIN data set and rerun PDUU. For more information, see the topic about [JCL statements for z/OS Problem Documentation Upload Utility in *z/OS MVS Diagnosis: Tools and Service Aids*](#).

Problem determination

This message is accompanied by return code 8 documented in the topic about [Return codes for the z/OS Problem Documentation Upload Utility in *z/OS MVS Diagnosis: Tools and Service Aids*](#).

Source

PDUU

Module

AMAPDUPL

Routing code

N/A

Descriptor code

N/A

AMA754E**CRYPTOGRAPHIC ASSIST IS NOT AVAILABLE ON THIS PROCESSOR****Explanation**

This message indicates that this processor cannot execute crypto instructions, so that cryptographic function of Problem Documentation Upload Utility (PDUU) is unavailable on this processor.

System action

PDUU processing ends.

System programmer response

None.

User response

Exclude CIPHER_KEY parameter from the SYSIN if it is affordable by your security policy and rerun PDUU, For more information about the CIPHER_KEY parameter, see the topic about [JCL statements for z/OS Problem Documentation Upload Utility in *z/OS MVS Diagnosis: Tools and Service Aids*](#).

Problem determination

This message is accompanied by return code 8 documented in the topic about [Return codes for the z/OS Problem Documentation Upload Utility in *z/OS MVS Diagnosis: Tools and Service Aids*](#).

Source

PDUU

Module

AMAPDUPL

Routing code

N/A

Descriptor code

N/A

AMA755E**192 BIT TRIPLE DES IS NOT AVAILABLE ON THIS PROCESSOR****Explanation**

This message indicates that this processor cannot perform 192 bit triple DES encryption, so that cryptographic function of PDUU is unavailable on this processor.

System action

PDUU processing ends.

System programmer response

None.

User response

Exclude CIPHER_KEY parameter and rerun PDUU.

- For more information about the CIPHER_KEY parameter, see the topic about [JCL statements for z/OS Problem Documentation Upload Utility in *z/OS MVS Diagnosis: Tools and Service Aids*](#).
- For more information and for Triple DES encryption see [z/OS Cryptographic Services ICSF Application Programmer's Guide](#) and [z/OS Communications Server: IP Configuration Guide](#).

Problem determination

This message is accompanied by return code 8 documented in the topic about [Return codes for the z/OS Problem Documentation Upload Utility in *z/OS MVS Diagnosis: Tools and Service Aids*](#).

Source

PDUU

Module

AMAPDUPL

Routing code

N/A

Descriptor code

N/A

AMA756E**THE PMR SPECIFIED IS NOT 13 CHARACTERS IN LENGTH****Explanation**

The Problem Documentation Upload Utility (PDUU) encountered an error while reading and interpreting SYSIN data set. The specified PMR parameter has an invalid length.

System action

PDUU processing ends.

System programmer response

None.

User response

Correct the PMR statement in SYSIN data set and rerun PDUU. For more information about the PMR statement, see the topic about [JCL statements for z/OS Problem Documentation Upload Utility in *z/OS MVS Diagnosis: Tools and Service Aids*](#).

Problem determination

This message is accompanied by return code 8 documented in the topic about [Return codes for the z/OS Problem Documentation Upload Utility in *z/OS MVS Diagnosis: Tools and Service Aids*](#).

Source

PDUU

Module

AMAPDUPL

Routing code

N/A

Descriptor code

N/A

AMA757E**INCORRECT CHARACTERS IN PMR CONTROL STATEMENT**

Explanation

The Problem Documentation Upload Utility (PDUU) encountered an error while reading and interpreting SYSIN dataset. Specified PMR parameter contains invalid characters

System action

None.

System programmer response

None.

User response

Correct PMR statement in SYSIN data set and rerun PDUU. For more information about the SYSIN statement, see the topic about [JCL statements for z/OS Problem Documentation Upload Utility in *z/OS MVS Diagnosis: Tools and Service Aids*](#).

Problem determination

None.

Source

PDUU

Module

AMAPDUPL

Routing code

N/A

Descriptor code

N/A

AMA758E**ERROR IN SYSIN STATEMENT. UNRECOGNIZED VERB: *verb*****Explanation**

The Problem Documentation Upload Utility (PDUU) encountered an error while reading and interpreting SYSIN data set. The specified verb is unknown for PDUU.

verb

The verb that PDUU failed to interpret

System action

None.

System programmer response

None.

User response

Correct statement with illegal verb in SYSIN data set and rerun PDUU. For more information about the SYSIN statement, see the topic about [JCL statements for z/OS Problem Documentation Upload Utility in *z/OS MVS Diagnosis: Tools and Service Aids*](#).

Problem determination

This message is accompanied by return code 8 documented in the topic about [Return codes for the z/OS Problem Documentation Upload Utility in *z/OS MVS Diagnosis: Tools and Service Aids*](#).

Source

PDUU

Module

AMAPDUPL

Routing code

N/A

Descriptor code

N/A

AMA759E**UNABLE TO OPEN FTPCMDS DATASET****Explanation**

The Problem Documentation Upload Utility (PDUU) encountered an error while attempting to open the specified FTPCMDS data set.

System action

None.

System programmer response

None.

User response

Check FTPCMDS data set and rerun PDUU. For more information, see the topic about [JCL statements for z/OS Problem Documentation Upload Utility in *z/OS MVS Diagnosis: Tools and Service Aids*](#).

Problem determination

This message is accompanied by return code 12 documented in the topic about [Return codes for the z/OS Problem Documentation Upload Utility in *z/OS MVS Diagnosis: Tools and Service Aids*](#).

Source

PDUU

Module

AMAPDUPL

Routing code

N/A

Descriptor code

N/A

AMA760I

***module ENDED ABNORMALLY. SYSTEM COMPLETION CODE = scode,
REASON CODE = reasc***

Explanation

The Problem Documentation Upload Utility (PDUU) encountered an error while executing *module* and ended abnormally. This message displays completion and reason codes accompanying the abnormal end.

In the message text:

module

The PDUU module that ended abnormally.

scode

The completion code describing the reason for the abnormal end.

reasc

The reason code describing the nature of the error. A value of 0 indicates PDUU could not tell the reason code due to the SDWA not being active.

System action

PDUU processing ends.

System programmer response

If the error recurs, search problem reporting databases for a fix for the problem. If no fix exists, contact the IBM Support Center.

User response

Look at the messages in the job log for more information, check correctness of the parameters passed in the SYSIN data set and rerun PDUU. If the abend recurs, notify the system programmer. For correct parameters, see the topic about [JCL statements for z/OS Problem Documentation Upload Utility in z/OS MVS Diagnosis: Tools and Service Aids](#).

Problem determination

This message is accompanied by return code 99 documented in the topic about [Return codes for the z/OS Problem Documentation Upload Utility in z/OS MVS Diagnosis: Tools and Service Aids](#).

Source

PDUU

Module

AMAPDUPL

Routing code

N/A

Descriptor code

N/A

AMA761E

UNABLE TO OBTAIN NECESSARY STORAGE

Explanation

The Problem Documentation Upload Utility (PDUU) was unable to obtain storage for the internal work areas and buffers.

System action

PDUU processing ends.

System programmer response

None.

User response

Increase the region size parameter in the JCL and rerun PDUU. For more information about the region size see [z/OS MVS JCL Reference](#).

In USE_HTTPS=Y mode, lower the values for the number of sessions or size of each session, with the CC_HTTPS or WORK_SIZE parameters, to reduce the amount of storage needed to successfully complete the processing. When adjusting these parameters, make sure the total file size can be divided into 99,999 parts or less.

Problem determination

This message is accompanied by return code 28. See the topic about [Return codes for the z/OS Problem Documentation Upload Utility](#) in [z/OS MVS Diagnosis: Tools and Service Aids](#).

Source

PDUU

Module

AMAPDUPL

Routing code

N/A

Descriptor code

N/A

AMA762I**UNABLE TO OPEN DEBUG DATASET****Explanation**

The Problem Documentation Upload Utility (PDUU) encountered an error while attempting to open specified DEBUG data set.

System action

PDUU processing continues but no data will be printed to the DEBUG data set.

System programmer response

None.

User response

Check the DEBUG data set characteristics and rerun PDUU. See the topic about [the DEBUG DD statement](#) in [z/OS MVS Diagnosis: Tools and Service Aids](#).

Source

PDUU

Module

AMAPDUPL

Routing code

N/A

Descriptor code

N/A

AMA763I THE LENGTH OF MGMTCLAS NAME SPECIFIED IS NOT BETWEEN 1 AND 8 CHARACTERS**Explanation**

The Problem Documentation Upload Utility (PDUU) encountered an error while reading and interpreting SYSIN data set. MgmtClas name is too long.

System action

PDUU processing ends.

System programmer response

None.

User response

Correct MGMTCLAS statement in SYSIN data set and rerun PDUU. For the correct MGMTCLAS statement in SYSIN, see the topic about JCL statements for z/OS Problem Documentation Upload Utility in [z/OS MVS Diagnosis: Tools and Service Aids](#).

Source

PDUU

Module

AMAPDUPL

Routing code

N/A

Descriptor code

N/A

AMA764I WAITING FOR COMPLETION OF FTP THREAD: nnn REMOTE FILE: ...suffix**Explanation**

AMAPDUPL has prepared the next file for transfer and is waiting for the FTP thread nnn to complete transfer of the previous file with resulting suffix. In the DEBUG DD, FTP messages indicating transfer rate may be shown depending on values set in the FTP.DATA PROGRESS parameter.

System action

PDUU processing continues.

System programmer response

None.

User response

Use the DEBUG dd to show in progress transfer rates for the FTP session. If the resulting messages indicate the transfer is significantly delayed, consult your network administrator to determine the reason for the delay, or if the PDUU files can be sent on a faster connection. When DEBUG output is directed to SYSOUT, refresh the job log to get the latest messages.

In the message text:,*nnn* is the thread that is waiting for completion. Work data set *Annn* or *Bnnn* is currently being sent, and the other will be sent when the previous data set transfer completes. The remote file that is being created looks like *Fnnnnn* where *n* is a number.

Source

PDUU

Module

AMAPDUPL

Routing code

N/A

Descriptor code

N/A

AMA770I**THE LENGTH OF SYSIN CONTROL STATEMENT EXCEEDS ALLOWABLE
LIMIT****Explanation**

The total length of the multiple record statement is more than the maximum allowable statement length.

System action

The Problem Documentation Upload Utility (PDUU) stops reading the SYSIN data set and ends processing.

System programmer response

None.

User response

Look for error statement in the PDUU job listing. Fix it and rerun the job.

Problem determination

None.

Source

PDUU

Module

AMAPDUPL, MTFTPR

Routing code

N/A

Descriptor code

N/A

AMA771I**ERROR IN SYSIN STATEMENT. THE LENGTH OF VERB SPECIFIED
EXCEEDS ALLOWABLE LIMIT.****Explanation**

The length of the verb specified in the SYSIN control statement is longer than the maximum possible length.

System action

PDUU execution continues.

System programmer response

None.

User response

Use the correct verb for the desired function and rerun PDUU.

Source

Problem documentation upload utility (PDUU)

Module

AMAPDUPL

AMA772I**THE NUMBER OF FILES TRANSFERRED TO THE TARGET SYSTEM
EXCEEDS THE LIMIT****Explanation**

The Problem Documentation Upload Utility PDUU can not complete the transmission of the entire input data set. Based upon the WORK_DSN_SIZE specified, the number of work files needed to transmit the data exceeds the 999 file number limit.

System action

The Problem Documentation Upload Utility (PDUU) execution ended.

Operator response

None.

System programmer response

None.

User response

Increase the maximum size of the work files by specifying a larger value for the WORK_DSN_SIZE control statement when rerunning PDUU.

Problem determination

None.

Source

PDUU

Module

AMAPDUPL

AMA773I **AMATERSE INVOCATION FAILED, PROCESSING OPTION = *optionName*****Explanation**

In the message text:

optionName

The name of the Amaterse processing option specified via PARM keyword.

Attempt to invoke Amaterse utility with specified *optionName* failed.

System action

MTFTPR processing ends.

Operator response

None.

System programmer response

None.

User response

Check job output for error messages describing the situation and try to fix the problem.

Problem determination

None.

Source

PDUU

Module

MTFTPR

AMA774I **UNABLE TO *actionName datasetType* DATASET WITH AMATERSE****Explanation**

In the message text:

actionName

The name of Amaterse processing_action - PACK or UNPACK.

datasetType

The type of dataset processing with Amaterse - INPUT or OUTPUT.

Amaterse failed to process specified data set.

System action

MTFTPR processing ends.

Operator response

None.

System programmer response

None.

User response

Check job output for Amaterse messages describing the situation and try to fix the problem..

Problem determination

None.

Source

PDUU

Module

MTFTPR

AMA775I**UNABLE TO DYNAMICALLY ALLOCATE PARTITIONED DATASET FOR
AMATERSE****Explanation**

MTFTPR can not dynamically allocate partitioned output data set for Amaterse and can not unpack received data set packed with Amaterse.

System action

MTFTPR processing ends.

Operator response

None.

System programmer response

None.

User response

Specify partitioned output data set for Amaterse by using SYSUT2 optional DD statement.

Problem determination

None.

Source

PDUU

Module

MTFTPR

AMA776I*parameter* MUST BE SPECIFIED WITH A VALUE OF 'Y'**Explanation**

The Problem Documentation Upload Utility (PDUU) encountered an error while reading and interpreting SYSIN data set. The value specified for the *parameter* can only be 'Y' or *parameter* can be omitted.

System action

PDUU processing ends.

System programmer response

None.

User response

Correct the *parameter* statement in SYSIN data set and rerun PDUU. For the correct values for the *parameter* statement in SYSIN, see the topic about [JCL statements for z/OS Problem Documentation Upload Utility in z/OS MVS Diagnosis: Tools and Service Aids](#).

Problem determination

This message is accompanied by return code 8 documented in the topic about [Return codes for the z/OS Problem Documentation Upload Utility in z/OS MVS Diagnosis: Tools and Service Aids](#).

Source

PDUU

Module

AMAPDUPL

Routing code

N/A

Descriptor code

N/A

AMA779I*parameter1* AND *parameter2* NOT SPECIFIED, ONE IS REQUIRED

Explanation

parameter1 and *parameter2* statements were not found in the SYSIN data.

One of *parameter1* or *parameter2* is required for PDUU operation.

For example: PMR or CASE supplies data associated with the hardware or software problem, and is necessary to report the problem and track its solution. These parameters have no default value and either PMR, or CASE, must be specified.

System action

PDUU processing ends.

System programmer response

None.

User response

Supply *parameter1* or *parameter1* in the SYSIN input and rerun PDUU. For more details about those statements, see the topic about [JCL statements for z/OS Problem Documentation Upload Utility in z/OS MVS Diagnosis: Tools and Service Aids](#).

Problem determination

This message is accompanied by return code 16 documented in the topic about [Return codes for the z/OS Problem Documentation Upload Utility for the z/OS Problem Documentation Upload Utility in z/OS MVS Diagnosis: Tools and Service Aids](#).

Source

PDUU

Module

AMAPDUPL

AMA780I**THE CASE SPECIFIED IS NOT 11 CHARACTERS****Explanation**

The Problem Documentation Upload Utility (PDUU) encountered an error while reading and interpreting SYSIN data. The specified CASE parameter has an invalid length.

System action

PDUU processing ends.

System programmer response

None.

User response

Correct the CASE statement in the SYSIN data and rerun PDUU. For more information about the CASE statement, see the topic about [JCL statements for the z/OS Problem Documentation Upload Utility in z/OS MVS Diagnosis: Tools and Service Aids](#).

Problem determination

This message is accompanied by return code 8 documented in the topic about [Return codes for the z/OS Problem Documentation Upload Utility](#) for the z/OS Problem Documentation Upload Utility in [z/OS MVS Diagnosis: Tools and Service Aids](#).

Source

PDUU

Module

AMAPDUPL

AMA781I**INCORRECT CHARACTERS IN CASE CONTROL STATEMENT****Explanation**

The Problem Documentation Upload Utility (PDUU) encountered an error while reading and interpreting SYSIN data. The value specified for the CASE parameter contains incorrect characters.

System action

None.

System programmer response

None.

User response

Correct CASE statement in the SYSIN data and rerun PDUU. For more information about the SYSIN statement, see the topic about JCL statements for z/OS Problem Documentation Upload Utility in [z/OS MVS Diagnosis: Tools and Service Aids](#).

Problem determination

None.

Source

PDUU

Module

AMAPDUPL

AMA782I***parameter1 AND parameter2 BOTH SPECIFIED, SPECIFY ONLY ONE*****Explanation**

parameter1 and *parameter2* statements were both found in the SYSIN data.

Only one of *parameter1* or *parameter2* is allowed for PDUU operation.

For example: Both PMR and CASE statements were found in the SYSIN data. PMR and CASE statements are mutually exclusive. Specify either PMR or CASE.

System action

PDUU processing ends.

Operator response:

System programmer response

None.

User response

Supply either a *parameter1* or *parameter2* statement in the SYSIN input and rerun PDUU. For more details about those statements, see the topic about JCL statements for the z/OS Problem Documentation Upload Utility in [z/OS MVS Diagnosis: Tools and Service Aids](#).

Problem determination

This message is accompanied by return code 16 documented in the topic about [Return codes for the z/OS Problem Documentation Upload Utility](#) for the z/OS Problem Documentation Upload Utility in [z/OS MVS Diagnosis: Tools and Service Aids](#).

Source

PDUU

Module

AMAPDUPL

AMA783I

**HWTH service THREAD: *thread* RETCODE: *retcode* REASON: *rsncode*
SERVID: *serviceinfo* X: *reasonText***

Explanation

Information about the HTTP Enablement Toolkit service is given to aid in problem diagnosis. This will appear in DEBUG if return code is 0, and SYSPRINT if return code is nonzero.

In the message text:

- *service* - A description of the toolkit service that was performed by PDUU.
- *thread* - The PDUU thread that issued the service.
- *retcode* - The return code for the service.
- *rsncode* - The reason code for the service.
- *serviceinfo* - Service information returned by the toolkit.
- *reasonText* - When reason text is supplied, it is added after AMA783I beginning with 'X:' to describe the return code.

System action

PDUU processing continues on successful calls to the toolkit and PDUU processing ends on failures.

Operator response:

System programmer response

None.

User response

On failures, see [The z/OS HTTP/HTTPS protocol enabler in z/OS MVS Programming: Callable Services for High-Level Languages](#) and the linked service return and reason code descriptions to determine why the toolkit service failed.

Problem determination

On failures, this message is accompanied by return code 36 documented in the topic about [Return codes for the z/OS Problem Documentation Upload Utility](#) for the z/OS Problem Documentation Upload Utility in [z/OS MVS Diagnosis: Tools and Service Aids](#).

Source

PDUU

Module

AMAPDUPL

AMA785I

BPX1SDD Failed RETVAL: *retval* RETCODE: *retcode* REASON: *rsncode*

Explanation

Service BPX1SDD failed with the stated returned codes. PDUU needs to successfully call BPX1SDD when USE_HTTPS=Y.

In the message text:

- *retval* - Return value from the BPX1SDD service.
- *retcode* - Return code from the BPX1SDD service.
- *rsncode* - Reason code from the BPX1SDD service.

System action

PDUU processing ends.

System programmer response

None.

User response

See [z/OS UNIX System Services Messages and Codes](#) for information on the failing return and reason codes on the failing return and reason codes.

Problem determination

This message is accompanied by return code 71 documented in the topic about [Return codes for the z/OS Problem Documentation Upload Utility](#) for the z/OS Problem Documentation Upload Utility in [z/OS MVS Diagnosis: Tools and Service Aids](#).

Source

PDUU

Module

AMAPDUPL

AMA787I

SUBTASK *thread* FAILURE SERVICE: *service* RETCODE: *retcode*

Explanation

A subtask of AMAPDUPL performing processing for a USE_HTTPS-Y session abended.

In the message text:

- *thread* - The number of the thread that failed.

- *service* - The service the thread was performing at the time of failure.
- *retcode* - The PDUU return code at the time of failure.

System action

PDUU processing ends.

System programmer response

None.

User response

See accompanying messages to determine the cause of the failure.

Problem determination

This message is accompanied by return code 98 documented in the topic about [Return codes for the z/OS Problem Documentation Upload Utility](#) for the z/OS Problem Documentation Upload Utility in [z/OS MVS Diagnosis: Tools and Service Aids](#).

Source

PDUU

Module

AMAPDUPL

AMA788I

exitType **EXIT. THREAD: thread STATUS: statusCode VERSION: versionCode REASON: reason Z: headerName=headerValue Y: bodyContents**

Explanation

A response header or body exit received control and is recording status. This message is followed by the header or body contents when available. This can be helpful when debugging problems.

In the message text:

- *exitType* - RESPONSE HEADER or RESPONSE BODY.
- *thread* - The number of the thread.
- *statusCode* - HTTP Status code (for example; 200, 404).
- *versionCode* - The HTTP version of the request.
- *reason* - The reason text accompanying the status code.
- *headerName=headerValue* - When *exitType* is RESPONSE HEADER, headers are provided following AMA788I prefixed with 'Z:'
- *bodyContents* - When *exitType* is RESPONSE BODY, the contents of the response body are provided prefixed with 'Y:'. Blank lines are omitted from the output.

System action

PDUU processing ends.

System programmer response

None.

User response

On failures, use the provided HTTP status information, headers, and body contents to determine the cause of the failure for the request. You might need to set HTTPS_DEBUG and HTTPS_DEBUGDD to further aid problem determination.

Problem determination

This message is accompanied by return code 16 documented in the topic about [Return codes for the z/OS Problem Documentation Upload Utility](#) for the z/OS Problem Documentation Upload Utility in [z/OS MVS Diagnosis: Tools and Service Aids](#).

Source

PDUU

Module

AMAPDUPL

AMA789I

parameter1 MUST BE SPECIFIED WHEN parameter2 IS SPECIFIED

Explanation

When *parameter1* is specified, *parameter2* must also be specified.

System action

PDUU processing ends.

Operator response:**System programmer response**

None.

User response

Specify *parameter2* or omit *parameter1* in the SYSIN input and rerun PDUU. For more details about those statements, see the topic about JCL statements for the z/OS Problem Documentation Upload Utility in [z/OS MVS Diagnosis: Tools and Service Aids](#).

Problem determination

This message is accompanied by return code 16 documented in the topic about [Return codes for the z/OS Problem Documentation Upload Utility](#) for the z/OS Problem Documentation Upload Utility in [z/OS MVS Diagnosis: Tools and Service Aids](#).

Source

PDUU

Module

AMAPDUPL