
SHADOW®

# STARTED TASK PARAMETERS

NEON SYSTEMS, INC.

This document is published by the NEON Systems, Inc. Technical Publications Department and applies to Shadow® Server™, Version 4, Release 5, and Shadow® OS/390 Web Server™, Version 4, Release 5.



Address inquiries to:

**NEON Systems, Inc.**
14100 SW Freeway, Suite 500
Sugar Land, Texas 77478

World Wide Web: http://www.neonsys.com

Phone: 1-800-505-6366
(281) 491-4200 (Corporate Sales, Technical Support)
Fax: (281) 242-3880

**December 1999**

# Contents


**About this Publication** . . . . . v

How this Publication is Organized . . . . . v
Conventions . . . . . vi
Reader's Comments . . . . . vi
NEON Systems, Inc. Products . . . . . vii
Year 2000 Compliancy Statement . . . . . vii
Working with Technical Support . . . . . viii

**SDB/SWS Started Task Parameters** . . . . . 1

Introduction . . . . . 1
PRODADABAS . . . . . 3
PRODAPPCMVS . . . . . 4
PRODBROWSE . . . . . 11
PRODCICS . . . . . 15
PRODCOMM . . . . . 19
PRODFILE . . . . . 28
PRODGLV . . . . . 38
PRODHTML . . . . . 42
PRODIMS (PRODIMCO) . . . . . 45
PRODLICENSE (PRODLICN) . . . . . 48
PRODLOGGING (PRODLOG) . . . . . 50
PRODPARM . . . . . 56
PRODREXX . . . . . 68
PRODRPC (PRODRPPA) . . . . . 69
PRODRRS . . . . . 75
PRODSECURITY (PRODSEC) . . . . . 77
PRODSEF . . . . . 89
PRODSQL (PRODSQPA) . . . . . 93
PRODSTOR . . . . . 101
PRODTOKEN . . . . . 105
PRODTRAC (PRODTRACE) . . . . . 107
PRODTSOSRV . . . . . 123
PRODUSS . . . . . 127
PRODWLM . . . . . 129
PRODWWW . . . . . 131
Obsolete . . . . . 136

**Index** . . . . . Index-1

# About this Publication

This book describes the started task parameters for Shadow Direct and Shadow OS/390 Web Server.

## How this Publication is Organized

This book contains the following:

- A section containing all the parameter groups, as well as the parameters within each group. It includes the following information for each parameter:
  - Parameter name.
  - Parameter description.
  - Default value.
  - Whether or nots it's modifiable after initiazlization.
  - Whether or not it's output only.

# Conventions

This book contains the following highlighting conventions:

This book contains the following highlighting conventions:

**BOLD CAPS**
Identifies commands. For example:

Use the **KEYS** command to ...

Text enclosed in single quotes denotes library, data set, and DD names. For example:

```
'SLDSYSIN'    'PLUSIN'    'RESLIB'
```

Monospace
Identifies code examples, screen prompts, and messages, as well as directory paths. For example:

```
//STEP010    EXEC  PGM=NDBA2400
```

*Monospace Italics*
Identifies information you must provide at a screen prompt or in a text field. For example:

PARM='PARMLIB=*your.parmlib*'

<KEY> Identifies the key to press. For example:

<ENTER>

NEON Systems, Inc. uses *Release.Version* to identify software packages. For example, *Product 4.1*, denotes the fourth release, first revision of the software.

# Reader's Comments

At NEON Systems, Inc. we are always looking for good ideas. If you have any comments or suggestions regarding any of our publications, please complete the Reader's Comment form (located at the back of this book) and return it to NEON, Attention: Technical Publications Department.

**Mailing Address:** **NEON Systems, Inc.**
14100 SW Freeway, Suite 500
Sugar Land, Texas 77478

**Fax Number:** (281) 242-3880

You can also send comments to directly to our Technical Publications department via the following email address: **documentation@neonsys.com**.

Thank you!

# NEON Systems, Inc. Products

For a comprehensive list of the products currently marketed by NEON Systems, Inc., visit our World Wide Web site at: **http://www.neonsys.com**.

***Note:***
You can also access and download all of the current NEON publications from this Web site.

# Year 2000 Compliancy Statement

The following products from NEON Systems, Inc., are Year 2000 ready.

- **Enterprise Security Management Products**
- **Enterprise Subsystem Management Product Family**
- **Shadow® Product Family and Add-On Components**

The mainframe code for the Shadow Product Family, Version 3.1 and all subsequent versions, are Y2K ready.

All versions of the client code associated with Shadow® Direct™ and Shadow Enterprise Direct® are Y2K ready.

These products use four-digit year values both internally and externally (although, in a few cases, two-digit year values are displayed while four-digit year values are maintained internally).

***Note:***
While Shadow Direct, Shadow® OS/390 Web Server™, and Shadow Enterprise Direct are Y2K ready, customers should be aware that these products can provide access to data sources that may not be Y2K ready.

# Working with Technical Support

NEON Systems, Inc. provides a number of ways for you to obtain assistance for our products. All product support inquiries are handled by the same support group, regardless if you are a trial or a licensed customer. The following are available support options:

| Support Option | How to Access | How it Works | This Option is Best for: |
|---|---|---|---|
| **E-mail** | To contact Technical Support via e-mail:<br>**support@neonsys.com**<br>Email is available for receipt 24 hours a day, 7 days a week and is answered between 9AM-7PM CST Monday through Friday. | Email goes to the support queue, which is continuously monitored by a staff of cross-functional technical experts. It is answered in the order it is received. It is logged in the support database and assigned a trouble ticket number for tracking purposes. | This type of support is excellent for low to medium priority requests. It is a proven method for providing further information on critical problems that may have been phoned in. Email is a convenient way of sending us a list of lower priority items you have collected at a time that is convenient for you. |
| **Phone** | To contact Technical Support, please call:<br>**1-800-505-6366** (U. S. and Canada)<br>**1-281-491-4200** (outside North America) | During normal working hours you will be transferred to someone who can usually answer your question on the first call. You may be required to page a support person via our phone mail system after hours. | This type of support is best for high priority requests and initial installation questions. Use this option for any obvious system errors or anytime you need the most rapid reply to your question. |
| **Internet** | To access Internet support, please visit our Web site at:<br>**www.neonsys.com** | Simply visit our Web site. NEON Systems works to keep current, relevant materials on our Web site to support our trial and licensed customers. | This option provides immediate access to documentation, updated client-side drivers, and our product Knowledge Base. The Knowledge Base is a collection of questions answered by support. Use this option to answer your own questions or to get a better understanding of what customers ask on an ongoing basis. |
| **Account Manager** | To contact your NEON Systems Sales Representative, please call:<br>**1-800-505-6366** (U. S. and Canada)<br>**1-281-491-4200** (outside North America) | Your Sales Representative is your account manager. This person is ultimately responsible for your complete satisfaction with NEON Systems, Inc. | Contact your Sales Representative for pricing information, contract details, password renewal or if you feel your needs are not being met. |

# SDB/SWS Started Task Parameters

## Introduction

*Shadow Server* and *Shadow OS/390 Web Server* are controlled using certain parameters, known as SDB/SWS Started Task parameters. These parameters can be modified with the SDBxIN00 or SWSxIN00 initialization EXEC using a **MODIFY PARM** statement. The table below provides a description of all the parameters and their default values. The table also shows if the parameters are modifiable after the servers are started and whether the parameters are output only.

***Note:***

Many of these parameters can be modified with the SDB Task Parameters window of the ISPF/SDF application.

# PRODADABAS

| Parameter Name | Parameter Description | Default Value | Modifiable After Initialization | Output Only |
|---|---|---|---|---|
| ADABAS | ADABAS SUPPORT ACTIVATED.<br>(YES, NO)<br>Set the ADABAS option to YES if ADABAS support is to be activated. The ADABAS module, ADALNK, must be present in the STEPLIB concatenation when this option is set. | **YES** | No | No |
| ADABASAUTOMAP | ADABAS AUTOMAPPING ACTIVATED.<br>(YES, NO)<br>This parameter controls whether or not a customer can turn off the automapping feature. | **YES** | No | No |
| ADABASNODB2 | ADABASE USING DB2 FEATURE.<br>(YES, NO)<br>This parameter controls whether or not adabas dot can be used without db2 available. | **NO** | No | No |
| ADABASSECURITY | ADABAS SECURITY ACTIVATED.<br>(YES, NO)<br>This parameter controls whether or not a resource rule is to be constructed consisting of dbid and file. | **NO** | No | No |
| ADABASUID | ADABAS UID ADD3 ACTIVATED.<br>(YES, NO)<br>This parameter controls whether or not the customer can see the client uid in the ADABAS control block adds 3 field. | **NO** | No | No |
| READONLY | ADABAS READONLY ACTIVATED.<br>(YES, NO)<br>This parameter controls whether or not SQL access for ADABAS allows update type requests. | **NO** | No | No |

# PRODAPPCMVS

| Parameter Name | Parameter Description | Default Value | Modifiable After Initialization | Output Only |
|---|---|---|---|---|
| CHECKCONVIDINTERVAL | CONVID TIMEOUT CHECKING INTERVAL<br>This parameter controls how often each convid is checked to see if the convid has timed out. If the convid has timed out, the conversation is deallocated and the entry in the conversation id table is removed. Do not change this number unless you identify a situation where this number is inadequate. Call technical support for more information.<br>**Minimum: 0 Maximum: 3600** | **15 SECONDS** | Yes | No |
| IMSCNVIDTBLSZ | IMS CONVERSATION ID TABLE SIZE<br>This parameter can be used to specify the size of the table used to maintain the status of active conversations.<br>Do not change this number unless you identify a situation where this number is inadequate. Call technical support for more information.<br>**Minimum: 2048 Maximum: 262144** | **32K** | No | No |
| IMSCONVTYPE | DEFAULT IMS CONVERSATION TYPE.<br>This parameter identifies the conversation type on which the service is invoked There are two possible values, however this value should be set to Basic or omitted altogether:<br>**Basic:** (default) TPs will format their data into separate records, with record length and data specified, before sending it.<br>**Mapped**: TPs will rely on APPC to format the data that the TPs send. | **BASIC** | Yes | No |
| IMSDEALLOCONVTIME | DEALLOC IMS CONV TIME VALUE<br>This parameter specifies the maximum allowable duration of inactivity for any conversation. The inactive period is defined as the time expired since the last APPC/MVS call. Do not change this number unless you identify a situation where this number is inadequate. Call technical support for more information. | **900 SECONDS** | Yes | No |
| IMSDEFAULTMAPNAME | DEFAULT IMS MAP NAME. | **'DFSDSP01'** | Yes | No |

| Parameter Name | Parameter Description | Default Value | Modifiable After Initialization | Output Only |
|---|---|---|---|---|
| IMSLOCALLU | DEFAULT IMS LOCAL LUNAME.<br>This paramter specifies the name of the local LU from which the caller's allocate request is to originate. The ability to specify the local LU name allows the caller to associate its outbound conversations with particular LUs. The caller's address space must have access to the named LU. Otherwise, a parameter_error return code is returned.<br><br>This is the new local LU Name specified in `'SYS1.PARMLIB(APPCPMxx)'` This parameter is optional; the default is to use the APPC Base LU, as is defined in `'SYS1.PARMLIB(APPCPMxx)'`.<br><br>**Note**: It is recommended that a separate Local LU be defined for each Shadow Server you have running using IMS/APPC. Application developers should be informed of which LU to use with which copy of the Shadow Server. *The APPC base LU will work in most cases, however using a separate Local LU tends to be a more reliable request.* | **NULL (SD)**<br>**'P393AIM1' (SWS)** | Yes | No |
| IMSLUEE0 | ACTIVATE DFSLUEE0 EXIT.<br>(YES, NO) | **NO (SD)**<br>**YES (SWS)** | Yes | No |
| IMSLUEE0ESCSEQ | DFSLUEE0 ESCAPE SEQUENCE. | **'<%NE02%>'** | Yes | No |

| Parameter Name | Parameter Description | Default Value | Modifiable After Initialization | Output Only |
|---|---|---|---|---|
| IMSMODENAME | DEFAULT IMS MODE NAME.<br>This parameter specifies the mode name designating the network properties for the session to be allocated for the conversation. The network properties include, for example, the class of service to be used. The mode name value of `'SNASVCMG'` is reserved for use by APPC/MVS. If a mode name of `'SNASVCMG'` is specified on the Allocate service, the request is rejected with a return code of parameter_error.<br><br>If you specify a symbolic destination name in the sym_dest_name parameter, set mode_name to blanks to obtain the mode_name from the side information.<br><br>If the partner LU is the same or on the same system as the local LU, mode_name is ignored. If the partner LU is on a different system, and you do not specify a sym_dest_name, a blank mode name defaults to any mode in effect for the local and partner LUs, or causes a return code of parameter_error. | **NULL** | Yes | No |
| IMSPARTNERLU | DEFAULT IMS PARTNER LUNAME.<br>This paramter is the name of the IMS LU as defined in `'SYS1.PARMLIB(APPCPMxx)'`. | **NULL (SD)**<br>**P390.P393AIMS' (SWS)** | Yes | No |
| IMSQUEUEKEEPTIME | DEFAULT IMS ALLOC QUEUE KEEP TIME VALUE.<br>Do not change this number unless you identify a situation where this number is inadequate. Call technical support for more information.<br>**Minimum: 0 Maximum: 3600** | **3600 SECONDS** | Yes | No |
| IMSRCVALLOCTIMEOUT | DEFAULT IMS RCVALLOC TIMEOUT VALUE.<br>Do not change this number unless you identify a situation where this number is inadequate. Call technical support for more information.<br>**Minimum: 0 Maximum: 3600** | **0 SECONDS** | Yes | No |

| Parameter Name | Parameter Description | Default Value | Modifiable After Initialization | Output Only |
|---|---|---|---|---|
| IMSRECVALLOCTYPE | DEFAULT IMS RECEIVE ALLOC TYPE.<br>This parameter can be used to specify whether to wait for an inbound allocate and, if so, for how long | **IMMEDIATE** | Yes | No |
| IMSRETURNCONTROL | DEFAULT IMS RETURN CONTROL.<br>This parameter can be used to specify when control is to returned to the local program within the context of session allocation. Possible values are:<br>• **SESSION.** (default and recommended value) Specifies to allocate a session for the conversation before returning control to the program. An error in allocating a session is reported on this call.<br>• **IMMEDIATE.** Specifies to allocate a session for the conversation if a session is immediately available, and return control to the program with a return code indicating whether a session is allocated. An error in allocating a session that is immediately available is reported on this call.<br>• **CONWINNER.** Specifies to allocate a session in which the local LU is the contention winner, before returning control to the program. As contention winner, the LU avoids having to compete with the partner LU to establish the session, thus potentially saving network traffic. An error in allocating a contention winner session for the conversation is reported on this call. | **SESSION** | Yes | No |

| Parameter Name | Parameter Description | Default Value | Modifiable After Initialization | Output Only |
|---|---|---|---|---|
| IMSSECURITYTYPE | DEFAULT IMS SECURITY TYPE.<br>This parameter can be used to specify the type of access information the partner LU uses to validate access to the partner program and its resources. Possible values are:<br>• **NONE.** Omit access security information on this allocation request.<br>• **SAME.** Use the userid and security profile (if present) from the allocation request that initiated the local program. The password (if present) is not used; instead, the userid is indicated as being already verified. If the allocation request that initiated execution of the local program contained no access security information, then access security information is omitted on this allocation request.<br>• **PROGRAM.** Use the access security information that the local program provides on the call. The local program provides the information by means of the User_id, Password, and Profile parameters. These values are passed exactly as specified, without folding to uppercase. | **NONE** | Yes | No |

| Parameter Name | Parameter Description | Default Value | Modifiable After Initialization | Output Only |
|---|---|---|---|---|
| IMSSYMDEST | DEFAULT IMS SMBOLIC DEST NAME.<br>This parameter specifies a symbolic name representing the partner LU, the partner TP_name, and the mode name for the session on which the conversation is to be carried. The symbolic destination name must match that of an entry in the side information data set. The appropriate entry in the side information is retrieved and used to initialize the characteristics for the conversation.<br><br>If you specify a symbolic destination name, the partner LU name, mode name, and TP name are obtained from the side information. If you also specify values for the Partner_LU_name, Mode_name, or TP_name parameters on the Allocate service, these values override any obtained from the side information.<br><br>The symbolic destination name in this field can be from 1 to 8 characters long, with characters from character set 01134. If the symbolic destination name is shorter than eight characters, it must be left-justified in the variable field, and padded on the right with blanks. To not specify a symbolic destination name, set the sym_dest_name parameter value to 8 blanks and provide values for the Partner_LU_name, Mode_name, and TP_name parameters. | **NULL** | Yes | No |
| IMSSYNCLEVEL | DEFAULT IMS SYNC LEVEL.<br>This parameter can be used to specify the synchronization levels of the local and partner TP.<br>The possible values are:<br>• **NONE.** (default) Program will not perform confirmation processing on this conversation. Programs will not call any services and will not recognize any returned parameters relating to confirmation.<br>• **CONFIRM.** Programs can perform confirmation processing on this conversation. The programs can call services and will recognize returned parameters relating to confirmation. | **NONE** | Yes | No |

| Parameter Name | Parameter Description | Default Value | Modifiable After Initialization | Output Only |
|---|---|---|---|---|
| IMSTXNTIMEOUT | DEFAULT IMS TXN TIMEOUT VALUE.<br>This parameter can be used to limit the wait time for the completion of a transaction. If the transaction times out, a message is placed in the communication buffer to notify the client that a timeout has occurred.<br>**Minimum Value: 0 Maximum Value: 300** | **0 SECONDS** | Yes | No |
| MONITORAPPC/MVS | MONITOR APPC/MVS.<br>(YES, NO)<br>This paramter specifies whether or not to monitor APPC/MVS conversations. Do not change this number unless you identify a situation where this number is inadequate. Call technical support for more information. | **NO** | Yes | No |
| REALTIMESUMMARY | IN MEMORY REALTIME SUMMARY COUNT.<br>This parameter controls the number of APPC/MVS real-time summary records to keep in memory at one time. If this parameter is set to zero, then no APPC/MVS real-time summary records will be retained in memory. The APPC/MVS summary records kept in memory can be interactively displayed. Do not change this number unless you identify a situation where this number is inadequate. Call technical support for more information.<br>**Minimum: 0 Maximum: 360** | **60 INTERVALS** | Yes | No |

# PRODBROWSE

| Parameter Name | Parameter Description | Default Value | Modifiable After Initialization | Output Only |
|---|---|---|---|---|
| ARCHIVEDATACLASS | ARCHIVE DEFINE CLUSTER DATACLASS.<br>This parameter defines the DATACLASS operand value used to define linear clusters for archive datasets. If not set, DATACLASS is not specified when the linear datasets are allocated. | **NULL** | Yes | No |
| ARCHIVEDEFCLPARMS | ARCHIVE DEFINE CLUSTER PARAMETERS.<br>This parameter contains additional parameter values which are passed on DEFINE CLUSTER statements generated to define archive backup datasets. | **NULL** | Yes | No |
| ARCHIVEDSNPREFIX | ARCHIVE DATASET NAME PREFIX.<br>This parameter defines the high-level qualifier which the subsystem uses to construct datasets names for trace browse archive files. The value ".Dyyyyddd.Thhmmss" is appended to the qualifier, where "yyyyddd" is the julian date, and "hhmmss" is the time of day. Trace browse archival processing cannot be performed if this prefix is not set, since there is no default value. | **NULL** | Yes | No |
| ARCHIVEMGMTCLASS | ARCHIVE DEFINE CLUSTER MGMTCLASS.<br>This parameter defines the MGMTCLASS operand value used to define linear clusters for archive datasets. If not set, MGMTCLASS is not specified when the linear datasets are allocated. | **NULL** | Yes | No |
| ARCHIVESTORCLASS | ARCHIVE DEFINE CLUSTER STORCLASS.<br>This parameter defines the STORCLASS operand value used to define linear clusters for archive datasets. If not set, STORCLASS is not specified when the linear datasets are allocated. | **NULL** | Yes | No |

| Parameter Name | Parameter Description | Default Value | Modifiable After Initialization | Output Only |
|---|---|---|---|---|
| BROWSEARCHIVE | BROWSE DATA ARCHIVING OPTION.<br>This parameter controls whether the product produces archives of the wrap-around trace and how the archival procedure is inaugurated. If set to NONE, archival of the trace is not supported and only user-requested ARCHIVE EXTRACTs are supported; explicitly requested EXTRRACT archives are not considered to be "backup" type archives. If set to AUTO, archival is triggered by automatically generating an ARCHIVE BACKUP command. If set to MESSAGE, the system generates a message when archiving should be performed, the generation of the ARCHIVE BACKUP command is not performed automatically. | **NONE** | Yes | No |
| BROWSEARCHIVECOUNT | BROWSE MESSAGES TO ARCHIVE AT A TIME.<br>This parameter is the number of messages to be written for each automated archival operation. Recommend value is no more than one-third of the BROWSEMAX value. | **0 MESSAGES** | Yes | No |
| BROWSEARCHIVECUSHION | ARCHIVE BACKUP CUSHION COUNT.<br>This parameter is the number of messages used as a threshold for automated triggering of an archive event and as a guard against archiving overwritten messages. An archive event is scheduled for each group of BROWSEARCHIVECOUNT messages. However, scheduling is deferred until BROWSEARCHIVECUSHION additional messages have been logged. This cushion is required because some messages are updated in place, and allows the system to get beyond the ACTIVE message range before actually copying the messages to a backup. The cushion value is also used if a backup is requested and overlay of previously un-backed-up message is in progress or imminent. The system begins the archive with the next un-archived message, when possible. But if overlay is imminent or in-progress, already, this many messages are skipped in order to ensure that these overlaid messages are not copied. | **0 MESSAGES** | Yes | No |
| BROWSEBLOCKS | NUMBER OF BLOCKS IN TRACE BROWSE. | **200 BLOCKS** | No | No |

| Parameter Name | Parameter Description | Default Value | Modifiable After Initialization | Output Only |
|---|---|---|---|---|
| BROWSEDDNAME | BROWSE DATA SET DDNAME. | **'SDBTRACE' (SD)**<br>**'SWSTRACE' (SWS)** | No | No |
| BROWSEINTERVAL | BROWSE CHECKPOINT INTERVAL.<br>**Minimum Value: 1 Maximum Value: 300** | **15 SECONDS** | Yes | No |
| BROWSEMAX | BROWSE MAXIMUM MESSAGE COUNT. | **12000 MESSAGES** | No | No |
| CLEARARCHIVERECOVERY | CLEAR ARCHIVE RECOVERY STATUS FIELDS.<br>This parameter, if set to YES during start-up, will cause any in-flight archive recovery and cleanup operations to be bypassed. It does so by clearing the in-flight indicators. Cleanup of an incomplete trace browse archive must be handled manually, since setting this flag causes the Server to delete all the information needed to invoke automatic cleanup at a later time. | **NO** | No | No |
| EXTRACTDATACLASS | EXTRACT DEFINE CLUSTER DATACLASS.<br>This parameter defines the DATACLASS operand value used to define linear clusters for EXTRACT datasets. If not set, DATACLASS is not specified when the linear datasets are allocated. | **NULL** | Yes | No |
| EXTRACTDEFCLPARMS | EXTRACT DEFINE CLUSTER PARAMETERS.<br>This parameter contains additional parameter values which are passed on DEFINE CLUSTER statements generated to define EXTRACT backup datasets. | **NULL** | Yes | No |
| EXTRACTDSNPREFIX | EXTRACT DATASET NAME PREFIX.<br>This parameter defines the high-level qualifier which the subsystem uses to construct datasets names for trace browse extract files. The value ".uuuuuuu.Dyyyyddd.Thhmmss" is appended to the qualifier, where "uuuuuuu" is the julian date, and "hhmmss" is the time of day. Trace browse extract processing cannot be performed if this prefix is not set, since there is no default value. | **NULL** | Yes | No |

| Parameter Name | Parameter Description | Default Value | Modifiable After Initialization | Output Only |
|---|---|---|---|---|
| EXTRACTMGMTCLASS | EXTRACT DEFINE CLUSTER MGMTCLASS.<br>This parameter defines the MGMTCLASS operand value used to define linear clusters for EXTRACT datasets. If not set, MGMTCLASS is not specified when the linear datasets are allocated. | **NULL** | Yes | No |
| EXTRACTSTORCLASS | 'EXTRACT DEFINE CLUSTER STORCLASS.<br>This parameter defines the STORCLASS operand value used to define linear clusters for EXTRACT datasets. If not set, STORCLASS is not specified when the linear datasets are allocated. | **NULL** | Yes | No |

# PRODCICS

| Parameter Name | Parameter Description | Default Value | Modifiable After Initialization | Output Only |
|---|---|---|---|---|
| CICSCONNECTRETRY | CICS CONNECT RETRY INTERVAL.<br>This parameter specifies the duration, in seconds, of the interval between attempts to connect to the target CICS region(s).<br>**Minimum Value: 15 Maximum Value: None** | **300 SECONDS** | Yes | No |
| CICSDATACONV | CONVERT NULLS TO BLANKS.<br>(YES, NO)<br>This parameter controls the conversion of null bytes to blanks for the CICS Transaction Server. | **YES** | Yes | No |
| CICSIRCSVCNO | CICS INTERREGION SVC NBR.<br>This parameter indicates the SVC number which is assigned to the Interregion SVC. | **X'D8'** | No | No |
| CICSLOADBALANCE | USE TXN QDEPTH FOR LOAD BALANCING.<br>(YES, NO)<br>This parameter controls whether or not the CICS transaction queue depth is to be used in load balancing decisions. | **NO** | Yes | No |
| CICSIRCSVCVR | CICS INTERREGION SVC VERSION.<br>This parameter indicates the version of the Interregion SVC which is active on the current system. | **NULL** | No | Yes |
| CICSMAXCONNECTIONS | MAXIMUM NUMBER OF CONNNECTIONS.<br>This parameter specifies the maximum number of connections which may be defined. This number indicates the total number of connections to all CICS regions. | **0** | Yes | No |
| CICSPROCOWNER | CICS STORED PROCEDURE OWNER.<br>This parameter allows the user to specify the procedure owner for IMS stored procedure map. | **'CICSEXCI'** | Yes | No |
| CICSSUBSYSTEM | CICS SUBSYSTEM NAME.<br>This parameter indicates the subsystem name which CICS is defined as using. | **'CICS'** | No | No |

| Parameter Name | Parameter Description | Default Value | Modifiable After Initialization | Output Only |
|---|---|---|---|---|
| CICSTXNSERVERNAME | CICS TXN SERVER NAME.<br>This parameter specifies the name which is used in CICS to define the CICS Txn Server to CICS. | **'SDBB' (SD)**<br>**'SWSS' (SWS)** | No | No |
| CICSTXNTIMEOUT | TRANSACTION TIMEOUT VALUE.<br>This parameter can be used to limit the wait time for the completion of a transaction.<br>**Minimum Value: 0 Maximum Value: 300** | **30 SECONDS** | Yes | No |
| DELETEEXCIMODULES | DELETE EXCI MODULES.<br>(YES, NO)<br>This parameter controls whether or not modules left in storage after an unsuccessful EXCI INIT_USER call should be delelted. This parameter will be set to YES by default until IBM generates a fix for this problem. | **YES** | Yes | No |
| DURETRY | SDUMP RETRY DURATION VALUE.<br>This parameter specifies the total time, in seconds, that the external CICS interface is to continue trying to obtain an MVS system dump using the SDUMP macro.<br>**Minimum Value: 0 Maximum Value: 30** | **0 SECONDS** | Yes | No |
| EXCI | INITIALIZE EXCI SUPPORT.<br>(YES, NO)<br>This parameter specifies whether or not the EXCI support is initialized. | **YES** | No | No |
| EXCIAPITYPE | EXCI DEFAULT API TYPE.<br>This parameter specifies the default APITYPE for EXCI support | **EXCI** | Yes | No |
| EXCICALLBYREF | EXCI CALL BY REFERENCE.<br>(YES, NO)<br>This parameter controls how parameters are passed via the shadow_cics interface. The default method is call by value, if the following parameter is set to YES, the the parameter passing method is call by reference. | **NO** | Yes | No |

| Parameter Name | Parameter Description | Default Value | Modifiable After Initialization | Output Only |
|---|---|---|---|---|
| EXCICONNECTIONNAME | EXCI DEFAULT CONNECTION NAME.<br>This parameter specifies the default CICS Connection Name for EXCI support. | **‘EXCS’** | Yes | No |
| EXCIDATACONV | EXCI CONVERT NULLS TO BLANKS.<br>(YES, NO)<br>This parameter controls the conversion of null bytes to blanks for the CICS Transaction Server. | **YES (SD)**<br>**NO (SWS)** | Yes | No |
| EXCIPIPEPREALLOC | PREALLOCATE EXCI PIPES.<br>(YES, NO)<br>This parameter specifies whether or not EXCI pipes are preallocated and opened for use. | **YES** | No | No |
| EXCIPIPEPREOPEN | PREOPEN EXCI PIPES.<br>(YES, NO)<br>This parameter specifies whether of not EXCI pipes are preallocated and preopened for use. | **YES** | No | No |
| EXTTRACE | EXCI EXTERNAL TRACE.<br>This parameter specifies whether you want external CICS interface internal tracing, and at what level. | **'OFF'** | Yes | No |
| GTF | EXCI GTF TRACE.<br>This parameter specifies whether all trace entries normally written to the external CICS interface internal trace table are also to be written to an MVS generalized trace facility (GTF) data set (if GTF tracing is active). | **'OFF'** | Yes | No |
| MDIVIACICS | EXECUTE MDI RSP VIA CICS.<br>(YES, NO)<br>This parameter specifies whether or not MDI RSPs are to be executed in the CICS address space. | **YES** | Yes | No |
| MSGCASE | EXCI MESSAGE CASE.<br>This parameter specifies whether the DFHEXxxxx messages are to be issued in mixed or upper case. | **'MIXED'** | Yes | No |

| Parameter Name | Parameter Description | Default Value | Modifiable After Initialization | Output Only |
|---|---|---|---|---|
| NEONMRO | INITIALIZE NEON MRO SUPPORT.<br>(YES, NO)<br>This parameter specifies whether or not the NEONMRO support is initialized. | **NO** | No | No |
| SESSIONWAITINTERVAL | SESSION WAIT INTERVAL VALUE.<br>Specifies duration of task waiting for the EXCI pipe will wait before retrying the DPL request.<br>**Minimum Value: 0 Maximum Value: 1000** | **100 MILLISECONDS** | Yes | No |
| SESSIONWAITTIME | SESSION WAIT TIME VALUE.<br>Specifies duration of time the caller requesting the EXCI pipe will wait for one to become available.<br>**Minimum Value: 0 Maximum Value: 300000** | **6000 MILLISECONDS OR 60 SECONDS** | Yes | No |
| TIMEOUT | DPL REQUEST TIMEOUT VALUE.<br>This parameter specifies the time interval, in hundredths of a second, that the external CICS interface is to wait for a DPL command to complete.<br>**Minimum Value: 1 Maximum Value: 2147483647** | **6000 HUNDREDTHS** | Yes | No |
| TRACESZE | INTERAL TRACE TABLE SIZE.<br>This parameter specifies the size in kilobytes of the internal trace table for use by the external CICS interface. This table is allocated in virtual storage above the 16MB line. You should ensure that there is enough virtual storage for the trace table by specifying a large enough region size on the MVS REGION parameter.<br>**Minimum Value: 16 Maximum Value: 1048576** | **16 KILOBYTES** | Yes | No |
| TRAP | USE EXCI SERVICE TRAP (DFHXCTRA).<br>This parameter specifies whether the service trap module, DFHXCTRA, is to be used. | **'OFF'** | Yes | No |

# PRODCOMM

| Parameter Name | Parameter Description | Default Value | Modifiable After Initialization | Output Only |
|---|---|---|---|---|
| APPLID | VTAM APPLICATION ID. | **'SDBBAPPL'** | Yes | No |
| BYPASSCOMPRESSION | BYPASS OUTBOUND DATA COMPRESSION.<br>(YES, NO) | **NO** | Yes | No |
| CONNECTRETRYINT | CONNECT RETRY INTERVAL.<br>This field controls how long the main product address space waits between attempts to connect to any of the TCP/IP subsystems. This field is specified in seconds.<br>**Minimum Value: 30 Maximum Value: None** | **300 SECONDS** | Yes | No |
| CONNECTTIMEOUT | TCP/IP CONNECT READ TIMEOUT VALUE.<br>This field is the timeout value for several host operations. Its most important use is to control how long the host will wait for a client TCP/IP (IBM, Interlink, and NSC) connection to complete. This applies only to Shadow Direct. Shadow OS/390 Web Server uses the value to control how long it will wait for an inbound URL to be sent. Interlink TCP/IP code uses this field as the timeout value for directory services requests. | **20** | Yes | No |
| IBMHOSTDOMAIN | IBM HOST DOMAIN NAME. | **NULL (SD)**<br>**'P390.NEONSYS.COM' (SWS)** | Yes | No |
| IBMPORTNUMBER | IBM TCP/IP PORT NUMBER.<br>This parameter sets the port number used to LISTEN for, and ACCEPT all inbound TCP/IP sessions. This port number should be reserved for exclusive use by the main product address space. Each copy of the main product address space will need its own separate port number if TCP/IP is being used. There is a default value for this port number if it is not set in the initialization EXEC. Note, that the port number can be set to a string of 'ANY'. This is a special value used to show that the system should assign an ephemeral port number for use by the product. | **'1200' (SD)**<br>**'80' (SWS)** | No | No |

| Parameter Name | Parameter Description | Default Value | Modifiable After Initialization | Output Only |
|---|---|---|---|---|
| IBMSSLPORTNUMBER | IBM SSL TCP/IP PORT NUMBER.<br>This parameter sets the port number used to LISTEN for, and ACCEPT all inbound encrypted TCP/IP sessions. This port number should be reserved for use only by the main product address space. Each copy of the main product address space will need its own port number if SSL over TCP/IP is being used. There is a default value for the SSL port number if the value is not set in the initialization EXEC.<br>**Minimum Value: 0 Maximum Value: 32767** | **300** | No | No |
| ITCHOSTDOMAIN | INTERLINK HOST DOMAIN NAME. | **NULL** | Yes | No |
| ITCKEEPALIVEOPTION | ITC/IP KEEPALIVE OPTION. | **NODATA/ ABORT** | Yes | No |
| ITCKEEPALIVETIME | ITC/IP KEEPALIVE TIME.<br>**Minimum Value: 15 Maximum Value: 1439** | **15 MINUTES** | Yes | No |
| ITCLISTENQDEPTH | ITC/IP LISTEN QUEUE DEPTH.<br>**Minimum Value: 0 Maximum Value: 100** | **5** | No | No |
| ITCMAXBUFFERSIZE | ITC/IP MAXIMUM BUFFER SIZE.<br>This field indicates the maximum Interlink TCP/IP buffer size for TRECV/TSEND. The default value is obtained from the TIB which is returned via the TINFO call. This parameter can be set by the user to override the TIB value.<br>**Minimum Value: 512 Maximum Value: ITCMXBU** | **0** | No | No |
| ITCPORTNUMBER | INTERLINK TCP/IP PORT NUMBER.<br>**Minimum Value: 0 Maximum Value: 32767** | **1200** | No | No |

| Parameter Name | Parameter Description | Default Value | Modifiable After Initialization | Output Only |
|---|---|---|---|---|
| TCSSLPORTNUMBER | INTERLINK SSL PORT NUMBER.<br>This parameter sets the port number used to LISTEN for, and ACCEPT all inbound encrypted Interlink TCP/IP sessions. This port number should be reserved for use only by the main product address space. Each copy of the main product address space will need its own port number if SSL over Interlink is being used. There is a default value for the SSL port number if the value is not set in the initialization EXEC.<br>**Minimum Value: 0 Maximum Value: 32767** | **443** | No | No |
| ITCSUBSYSTEM | LOCAL ITC/IP SUBSYSTEM NAME. | **NULL** | Yes | No |
| ITCTCLOSETIMEOUT | TCLOSEOUT TIME VALUE.<br>This parameter specifies whehter or not the main Interlink TCP/IP listener task waits on a timeout ECB to be posted. This parameter is a circumvention for an invalid CMTC problem which causes the attached task to go to EOT very early and, as a result, the mother task waits indefinitely for a TCLOSE OPTCD=OLD to occur.<br>**Minimum Value: 0 Maximum Value: 60000** | **0 MILLISECONDS** | Yes | No |
| KEEPALIVE | HTTP PERSISTENT SESSION REUSE SUPPORT.<br>(YES, NO)<br>The KEEPALIVE parameter determines whether the Server will honor 'Connection:' and 'Keep-alive:' headers for in-bound HTTP/1.0 requests.<br>When set to YES, the Server will attempt to honor in-bound headers which request persistent session support. When set to NO, the Server ignores such headers for all HTTP/1.0 requests. | **NO** | Yes | No |

| Parameter Name | Parameter Description | Default Value | Modifiable After Initialization | Output Only |
|---|---|---|---|---|
| KEEPALIVELIMIT | HTTP PERSISTENT SESSION REUSE LIMIT.<br>This parameter sets a limit on how many times an HTTP persistent session is left open for immediate re-use by the downstream user-agent. A small number is recommended when most downstream user-agents are desktop web browsers. A larger number is recommended when the downstream user agent is known to be a proxy server. A value in the range 1 to 512 may be specified.<br>**Minimum Value: 1 Maximum Value: 512** | **4** | Yes | No |
| KEEPALIVETIMEOUT | HTTP PERSISTENT SESSION REUSE TIMEOUT.<br>This parameter specifies how long to let persistent sessions wait for another HTTP request to arrive on a session kept open for reuse. | **400 MILLISECONDS** | Yes | No |
| MAXUDPSIZE | MAXIMUM UDP DATAGRAM SIZE.<br>This field determines the maximum size of any UDP datagrams sent from the host to a client. Any buffers larger than this value will be broken into multiple pieces. This value includes the size of the UDP prefix. This means that the actual amount of data sent will be somewhat smaller than the maximum value.<br>**Minimum Value: 4096 Maximum Value: 65536** | **8192 BYTES** | Yes | No |
| MSGROUTEFROM1 | MESSAGE ROUTE FROM CONNECTION 1.<br>This parameter specifies names of output device connections that are to be re-routed to other devices. | **NULL** | No | No |
| MSGROUTEFROM2 | MESSAGE ROUTE FROM CONNECTION 2. | **NULL** | No | No |
| MSGROUTETO1_1 | MESSAGE ROUTE TO CONNECTION 1. | **NULL** | No | No |
| MSGROUTETO1_2 | MESSAGE ROUTE TO CONNECTION 2. | **NULL** | No | No |
| MSGROUTETO1_3 | MESSAGE ROUTE TO CONNECTION 3. | **NULL** | No | No |
| MSGROUTETO1_4 | MESSAGE ROUTE TO CONNECTION 4. | **NULL** | No | No |
| MSGROUTETO1_5 | MESSAGE ROUTE TO CONNECTION 5. | **NULL** | No | No |

| Parameter Name | Parameter Description | Default Value | Modifiable After Initialization | Output Only |
|---|---|---|---|---|
| MSGROUTETO2_1 | MESSAGE ROUTE TO CONNECTION 2. | **NULL** | No | No |
| MSGROUTETO2_2 | MESSAGE ROUTE TO CONNECTION 2. | **NULL** | No | No |
| MSGROUTETO2_3 | MESSAGE ROUTE TO CONNECTION 3. | **NULL** | No | No |
| MSGROUTETO2_4 | MESSAGE ROUTE TO CONNECTION 4. | **NULL** | No | No |
| MSGROUTETO2_5 | MESSAGE ROUTE TO CONNECTION 5. | **NULL** | No | No |
| NETMODE | NETWORK EXECUTION MODE.<br>This parameter controls how UDP and TCP/IP are used. The modes control if the main address space handles UDP or TCP/IP sessions and how many tasks are used to accept inbound sessions. | **NONE** | No | No |
| NETWORKADJUST | NETWORK BUFFER ADJUSTMENT FACTOR.<br>This parameter controls what fraction of the communication buffer should be reserved to allow for buffer overflow. If the field is set to 20, then 1/20th of the buffer will be reserved. If it is set to 5, 1/5th of the buffer will be reserved. This value should be reduced if buffer overflow errors occur.<br>**Minimum Value: 3 Maximum Value: 100** | **20** | Yes | No |
| NETWORKBUFFERSIZE | MAXIMUM NETWORK I/O BUFFER SIZE.<br>This field controls the size of the buffer used to receive blocks of data from the network. A failure will occur if a client application sends a buffer larger than the maximum size. This value should be raised to allow larger blocks of data to be sent to and from the client.<br>**Minimum Value: 4096 Maximum Value: 4194304** | **87K** | No | No |
| NTCFORCEACCEPT | NTC/IP ACCEPT FIX. | | Yes | No |
| NTCINTERFACE | LOCAL NTC/IP INTERFACE NAME. | **NONE** | Yes | No |
| NTCLISTENQDEPTH | ITC/IP LISTEN QUEUE DEPTH.<br>**Minimum Value: 0 Maximum Value: 100** | **5** | No | No |

| Parameter Name | Parameter Description | Default Value | Modifiable After Initialization | Output Only |
|---|---|---|---|---|
| NTCPORTNUMBER | NETWORK SYSTEMS TCP/IP PORT NUMBER.<br>**Minimum Value: 0 Maximum Value: 32767** | **1200** | No | No |
| NTCSUBSYSTEM | LOCAL NTC/IP SUBSYSTEM NAME. | **NONE** | Yes | No |
| OEASYNCIO | OE SOCKETS ASYNC I/O.<br>(YES, NO)<br>This parameter controls if Async OE sockets calls should be used or not. If this parameter is set to YES, then Async I/O will be used with OE sockets. Async I/O is faster than synchronous I/O, but there are bugs in OE sockets that sometimes prevent Async I/O from working. if this parameter is set to NO, Async I/O will not be used with OE sockets. | **NO** | No | No |
| OEHOSTDOMAIN | OE SOCKETS HOST DOMAIN NAME.<br>This parameter specifies the fully qualified internet host domain name to be used by this Server when constructing fully-qualified HTTP URLs and domain settings for HTTP cookies. The OEHOSTDOMAIN parameter is used only for OE Sockets TCP/IP Connections. The IBMHOSTDOMAIN and ITCHOSTDOMAIN parameters set the MVS TCP/IP and Interlink TCP/IP host domains, respectively.<br>The setting of this parameter can have a significant impact on whether web browsers correctly store and later re-transmit HTTP cookie values sent to it from this Server.<br>Many web browsers will not store HTTP cookies when the domain name is set unless the name contains at least 3 embedded periods (2 periods if the name ends with .com, .edu, .net, .org, .gov, .mil, or .int). Other browsers may fail to transmit cookies properly unless this name is entirely lower case.<br>For this reason, the server will automatically convert any value you specify for this parameter to lower case, and will issue a warning message if it does not contain sufficient qualification. | **NULL** | Yes | No |

| Parameter Name | Parameter Description | Default Value | Modifiable After Initialization | Output Only |
|---|---|---|---|---|
| OEPORTNUMBER | OE SOCKETS PORT NUMBER.<br>This parameter sets the port number used to LISTEN for, and ACCEPT all inbound OE Sockets TCP/IP sessions. This port number should be reserved for exclusive use by the main product address space. Each copy of the main product address space will need its own separate port number if TCP/IP is being used. There is a default value for this port number if it is not set in the initialization EXEC. Note, that the port number can be set to a string of 'ANY'. This is a special value used to show that the system should assign an ephemeral port number for use by the product. | **NULL** | No | No |
| OESSLPORTNUMBER | OE SOCKETS SSL PORT NUMBER.<br>This parameter sets the port number used to LISTEN for, and ACCEPT all inbound encrypted OE Sockets TCP/IP sessions. This port number should be reserved for use only by the main product address space. Each copy of the main product address space will need its own port number if SSL over OE Sockets is being used. There is a default value for the SSL port number if the value is not set in the initialization EXEC.<br>**Minimum Value: 0 Maximum Value: 32767** | **0** | No | No |
| OESTACK | OE SOCKETS TCP/IP STACK NAME.<br>This parameter is used to specify the name of the OE TCP/IP stack that should be used. For OE TCP/IP, this parameter is optional. If this parameter is not set, then the default OI stack will be used. If this parameter is used to select an OE TCP/IP stack, then the value must be one of the SUBFILESYSTYPE values specified in the PBXPRMxx PARMLIB member. | **NULL** | No | No |

| Parameter Name | Parameter Description | Default Value | Modifiable After Initialization | Output Only |
|---|---|---|---|---|
| SSLACCEPTTIMEOUT | SSL ACCEPT WAIT TIMEOUT.<br>This value determines how long the Server will wait on receives during SSL acceptance processing. This timeout value is used during ACCEPT processing (instead of CONNECTTIMEOUT) because the end-user may need to verify the server certificate, or perform other human-speed operations.<br>**Minimum Value: 20 Maximum Value: 3600** | **180 SECONDS** | Yes | No |
| SOCKETLINGER | SOCKET LINGER TIME.<br>This parameter controls the socket linger time for IBM TCP/IP and IBM OE Sockets. If set to 0, socket linger is turned off. If set to a non-zero value, the socket linger is turned on and set for the number of seconds specified by this parameter.<br>**Minimum Value: 0 Maximum Value: 120** | **20 SECONDS (SD)**<br>**0 SECONDS (SWS)** | No | No |
| TCPMAXSESSIONS | IBM AND INTERLINK MAXIMUM SESSIONS.<br>**Minimum Value: 0 Maximum Value: 25000** | **2500 (SD)**<br>**200 (SWS)** | No | No |
| TCPMSGLIM | TCP/IP IUCV MESSAGE LIMIT.<br>This field displays the final number of IUCV messages that can be concurrently outstanding on each IUCV path. The value will be 10 for IBM MVS TCP/IP API type 2 and 255 for API type 3. This value should not be set and is actually output only at this time.<br>**Minimum Value: 1 Maximum Value: 255** | **255** | No | No |
| TCPNAME | LOCAL TCP/IP STARTED TASK NAME. | **'TCPIPMVS'** | Yes | No |
| UDPTIMEOUT | UDP SESSION TIMEOUT.<br>This field determines how long UDP session information is kept in memory before it is released. The value should be long enough to allow for some number I/O errors and retries. If the value is too high, large amounts of storage will be used. If the value is too small, retry operations may fail. This value is not related to FAILWAITTIME and does not limit how long an application can wait between sending separate requests to the host.<br>**Minimum Value: 60 Maximum Value: 3600** | **300 SECONDS** | No | No |

| Parameter Name | Parameter Description | Default Value | Modifiable After Initialization | Output Only |
|---|---|---|---|---|
| VTAMEXITS | ENABLE VTAM SCIP/LOGON EXITS.<br>(YES, NO) | **NO** | Yes | No |

# PRODFILE

| Parameter Name | Parameter Description | Default Value | Modifiable After Initialization | Output Only |
|---|---|---|---|---|
| DFHSM | DFHSM SUPPORT ENABLED WITHIN SERVER. (YES, NO)<br>This parameter specifies whether the server should pre-initialize DFHSM support during start-up. If the parameter is to NO, no pre-initialization is performed and authorized DFHSM services will be unavailable within the server. If set to YES, initialization is attempted, and if successful authorized DFHSM processing can be performed once start-up has completed. If errors are detected during initialization, warning message(s) are issued and DFHSM support is disabled. If disabled, no additional DFHSM processing of any kind, including clean-up of outstanding DFHSM MWE control blocks remaining after the last product shutdown is performed. | **NO (SD)**<br>**YES (SWS)** | No | No |
| DFHSMCLEANUPINTERVAL | DFHSM PENDING REQUEST CLEANUP INTERVAL."<br>The DFHSMCLEANUPINTERVAL controls how often a check for pending inflight HRECALL requests is made. Requests which time out are abandoned by transaction subtasks but must be cleaned up. Failure to free the DFHSM MWE ECB's can leave below-the-line CSA storage areas permanently allocated. The interval value is specified in seconds and should be a factor of one hour. In other words, the value should divide evently into 3600. The interval is automatically set to 3600 (1 hour) if DFHSM support is not enabled during start-up. | **120 SECONDS** | Yes | No |

| Parameter Name | Parameter Description | Default Value | Modifiable After Initialization | Output Only |
|---|---|---|---|---|
| DFHSMDRAIN | DFHSM DRAIN MODE IS IN EFFECT.<br>(YES, NO)<br>This parameter can be can be set manually to prevent the Server from scheduling new HRECALL requests. The Server continues to monitor already inflight requests for completion, and free the associated MWE control blocks. The Server, itself, will set DFHSMDRAIN(YES) into effect if more than 125 pending HRECALL requests are outstanding. It will then restore DFHSMDRAIN(NO) once the number of pending requests drops below 100, PROVIDING no manual change to DFHSMDRAIN or DFHSMSTATUS are made. Any manual change prevents the Server from automatically restoring full non-drain processing.<br>**NOTE** that DFHSMDRAIN(NO) is always put into effect by the Server ANY time you manually set the DFHSMSTATUS parameter. | **NO** | Yes | No |
| DFHSMDRAINAUTO | SERVER SHOULD AUTO-RESET DFHSMDRAIN.<br>(YES, NO)<br>This parameter is an output-only fields which is set to YES only after the Server, itself, has change DFHSMDRAIN to YES. While this flag remains set to YES, the Server is responsible for resetting DFHSMDRAIN(NO) once sufficient HRECALL completions have been detected to allow new requests to be scheduled. Manually changing either DFHSMSTATUS or DFHSMDRAIN causes this field to be set to NO, and prevents the Server from resetting DFHSMDRAIN automatically. | **NO** | No | Yes |
| DFHSMSHUTDOWNWAIT | SHUTDOWN WAIT FOR PENDING HRECALL REQUESTS. This parameter can be set to the number of seconds the product's main task should suspend if outstanding DFHSM HRECALL requests are still outstanding. Shutdown is delayed while waiting for DFHSM to post outstanding requests completed. if set to zero, or if the DFHSMSTATUS parameter is set to OFFLINE, no HRECALL completion handling is performed during shutdown. | **120 SECONDS** | Yes | No |

| Parameter Name | Parameter Description | Default Value | Modifiable After Initialization | Output Only |
|---|---|---|---|---|
| DFHSMSTATUS | DFHSM SERVICES ARE OFFLINE/ONLINE.<br>This parameter can be manually set during normal Server operations to temporarily suspend all Server interactions with DFHSM. The Server continues to remember all pending HRECALL requests and will attempt to complete them and free the associated MWE blocks once this parameter is restored to DFHSMSTATUS(ONLINE). Administrators can use this option to temporarily suspend DFHSM processing during times when DFHSM services are unavailable, or DFHSM is being restarted. *<br>**NOTE** that whenever this parameter is manually altered, the DFHSMDRAIN parameter is automatically reset to DFHSMDRAIN(NO). | **ONLINE** | Yes | No |
| FILEAPIRECALL | SWSFILE RECALL PROCESSING.<br>This parameter determines whether dataset recall is used when processing SWSFILE application programming interface requests. The parameter applies only to those requests which specify a DSNAME explicitly and ONLY when the dataset is not shared (i.e. DEFINE'd during start-up).<br>If set to AUTO, the values specified for the FILERECALL, FILEHRECALL, and HRECALLWAIT parameters are used. This option is strongly recommended for new customers.<br>If set to ALLOCATE, dataset recall for SWSFILE is handled by dynamic allocation processing. Existing customers may wish to set this option to maintain operational compatibility with previous released of the product. If set to FAIL, dataset migration is not allowed and the SWSFILE request fails when dataset migration is required. | **AUTO** | Yes | No |
| FILECACHE | DYNAMIC FILE CACHE OPTION.<br>This parameter allows the user to control whether or not to cache data retrieved from files. This will improve performance, however, the file must be closed and re-opened in order to refresh the cache. Valid options are ALL (to cache all data), NONE (to inhibit caching) and DIR (to only cache PDS directories. | **ALL** | Yes | No |

| Parameter Name | Parameter Description | Default Value | Modifiable After Initialization | Output Only |
|---|---|---|---|---|
| FILECLOSEAFTER | QUIESCE FILE AFTER TIME LIMIT.<br>This parameter allows the user to control the amount of time (in seconds) that files may remain open with access. The range is 0 to 32767 seconds. A setting of 0 indicates that there is no inactivity expiration time.<br>**Minimum Value: 0 Maximum Value: 32767** | **5** | Yes | No |

| Parameter Name | Parameter Description | Default Value | Modifiable After Initialization | Output Only |
|---|---|---|---|---|
| FILEHRECALL | DYNALLOC_TO_DFHSM RECALL CONVERSION.<br>This parameter determines whether or not to internally convert dynamic allocation dataset recall requests to asynchronous DFHSM HRECALL operations. Conversion of these requests can prevent system hangs upon the SYSZTIOT resource. When the DYNALLOC SVC handles dataset recalls internally, long-term enqueues can be generated upon SYSZTIOT if a migrated dataset cannot be recalled quickly. All other DYNALLOC requests stack up behind this enqueue.<br>This parameter is ignored if DFHSM support is not enabled or is currently suspended. This parameter controls recall operations whenever dataset allocation is performed for the following Server API interfaces:<br>• SWSALLOC operations operating with RECALL(YES) specified, or using system-wide default action of FILERE-CALL(YES).<br>• SWSFILE operations against a non-shared, DSNAME-based requests when FILEAPIRECALL(DEFAULT) is in effect.<br>• WWW rule process sectons, such as /*FILE, /*EXECSQL, *EXECIMS, etc. while processing a non-shared, DSNAME-based MVS dataset when FILERULERECALL(DEFAULT) is in effect.<br>The default setting is ALLOCATE which indicates tha DYNALLOC- to-DFHSM recall conversion is not performed. When dataset recall is necessary (and allowed) the DYNALLOC SVC handles dataset in-migration.<br>If set to CONVERT, dynamic allocation requests are issued with the "no-migration" flag set on. If DYNALLOC fails with an indication that data recall is required (i.e. SVC 99 error x'278'), DFHSM HRECALL is issued internally as a timed asynchronous request. If the HRECALL completes in the allowed time period, the dynamic allocation request is retried. The dynamic allocation request fails if HRECALL fails or the time period allowed expires. | **CONVERT** | Yes | No |

| Parameter Name | Parameter Description | Default Value | Modifiable After Initialization | Output Only |
|---|---|---|---|---|
| FILEMESSAGES | CONSOLE MESSAGES FROM DYNAMIC ALLOCATION. (YES, NO) This SEF parameter determines whether or not to allow a dynamic allocation error messages to be displayed upon the system console. This parameter only affects dynamic allocation requests made through the SWSALLOC application programming interface. | **YES** | Yes | No |
| FILEIOMODE | FILE I/O ADDRESSING MODE. This parameter allows the user to control whether to use the 31 bit addressing mode for file I/O or use the 24-bit addressing mode for file I/O. The 31-bit file I/O processing is limited to systems with DFSMS enabled. | **31** | Yes | No |
| FILEMOUNT | MOUNT VOLUMES FOR DYNAMIC ALLOCATION. (YES, NO) This SEF parameter determines whether or not to allow a volume to be mounted to satisfy a dynamic allocation request. This parameter only affects dynamic allocation requests made through the SWSALLOC application programming interface. | **YES** | Yes | No |
| FILERECALL | RECALL FILES FOR DYNAMIC ALLOCATION. (YES, NO) This SEF parameter determines whether or not to allow a dataset to be recalled by HSM to satisfy a dynamic allocation request. This parameter only affects dynamic allocation requests made through the SWSALLOC application programming interface. | **YES** | Yes | No |

| Parameter Name | Parameter Description | Default Value | Modifiable After Initialization | Output Only |
|---|---|---|---|---|
| FILEREXXTOOLRECALL | FILE REXXTOOL RECALL PROCESSING.<br>This parameter determines whether dataset recall is used when processing Shadow/REXXTools dynamic allocation requests. It specifies how migrated datasets are handled when dynamic allocation is requested. If set to AUTO, recall processing is handled as specified by the FILEHRECALL, and HRECALLWAIT parameters. Use of this option is recommended for all new customers. If set to ALLOCATE, dataset in-migration for requests are handled by dynamic allocation processing. Existing customers may wish to set this option to maintain operational compatibility with previous release of the product (this allows for no time out on recall requests, and may lead to hangs within SVC99 upon the SYSZTIOT queue name). If set to FAIL, dataset recall is not allowed and if a migrated dataset is requested, the dynamic allocation request fails. | **AUTO** | Yes | No |
| FILERULERECALL | FILE RULE RECALL PROCESSING.<br>This parameter determines whether dataset recall is used when processing /*FILE rules, or to any other WWW rule process section which processes files as part of it's operation (for instance, the input/output forms used by /*EXECSQL sections). It specifically applies ONLY to those rules which specify a DSNAME explicitly, and ONLY when the dataset is not shared (i.e. DEFINE'd during start-up). If set to AUTO, recall processing is handled as specified by the FILEHRECALL, and HRECALLWAIT parameters. Use of this option is recommended for all new customers. If set to ALLOCATE, dataset in-migration for rules is handled by dynamic allocation processing. Existing customers may wish to set this option to maintain operational compatibility with previous release of the product (this allows for no time out on recalls requests, but may lead to hangs within SVC99 upon the SYSZTIOT queue name). If set to FAIL, dataset recall is not allowed and if a migrated dataset is requested, the rule operation fails. | **AUTO** | Yes | No |

| Parameter Name | Parameter Description | Default Value | Modifiable After Initialization | Output Only |
|---|---|---|---|---|
| FILESECURITY | DYNAMIC FILE SECURITY OPTION.<br>This parameter allows the user to control the userid to use when validating access to a file. The valid operations are SUBSYS (to use the userid assigned to the Shadow Server) or USERID (to use the userid assigned to the client). | **SUBSYS** | Yes | No |
| FILESHAREDDN | DEFINE NEW DDNAMES DYNAMICALLY.<br>(YES, NO)<br>This parameter allows the user to control whether or not to share DDNames whenever possible. If a DDName is already open due to a previous allocation, parameter controls whether or not the DDName can be accessed by multiple users or does the DDName need to be re-allocated to another DDName for a subsequent user. | **YES** | Yes | No |
| FILESHAREDSN | DEFINE NEW DSNAMES DYNAMICALLY.<br>(YES, NO)<br>This parameter allows the user to control whether or not to share dataset names when possible. If a dataset is already open due to a previous allocation, parameter controls whether or not the dataset can be accessed by multiple user's or does the dataset need to be re-allocated to another DDName for a subsequent user. | **NO** | Yes | No |

| Parameter Name | Parameter Description | Default Value | Modifiable After Initialization | Output Only |
|---|---|---|---|---|
| FILESTAGINGSIZELIMIT | FILE STAGING SIZE LIMIT**.**<br>When data files are processed for transmission by the server, all native MVS files must be pre-staged before actual outbound transmission can be performed. This is done in order to correctly calculate the outbound HTTP Content-length header, and to process HTML extension statements within the source text. Set this limit to specify a maximum file size for pre-staging data. The minimum size allowed is 64K (65536).<br>This limit protects the system from overcommitting processor virtual storage while handling any single file service request. If the limit value is not an exact multiple of 64k, it is rounded to the next higher multiple. The maximum allowable limit size is 16MB minus one.<br>For files which exceed this limit, the Server aborts the processing of HTML extension statements and signals an oversize file condition. If possible, the original file request is re-driven using an alternate run-time processing algorithm. The alternate procedure re-opens the dataset using a thread-owned DCB (to avoid holding a shared file for a long period of time). It then reads and transmits the file data to the client in 64k segments, up to the limit imposed by the MAXHTTPRESPBYTES.<br>The server will not attempt to re-drive oversize file requests if they were originally requested by a REXX caller using DD name. This is because often REXX file allocations are temporary and freed at close, therefore, the file cannot be re-opened. Also, re-drive requests are only honored for SEND-to-client operations,<br>and cannot be handled if HTML extension statements are present within the input data.<br>**Minimum Value: 65536 Maximum Value: 16777215** | **2097152** | Yes | No |

| Parameter Name | Parameter Description | Default Value | Modifiable After Initialization | Output Only |
|---|---|---|---|---|
| HRECALLWAIT | WAIT TIME LIMIT FOR HRECALL.<br>When DFHSM HRECALL is used for dataset in-migration, this parameter determines how long the server suspends task execution to await recall completion. If set to zero, HRECALL requests are issued without waiting on completion. Dataset recall is scheduled using DFHSM, but the Server does not wait on completion. The dataset access operation fails and must be retried later.<br>**Note** that when a zero time limit is specified, the Server does not track HRECALL requests in any way. Any positive number in the range 1 to 32767 determines the number of seconds to await recall completion. If HRECALL does not complete within the alloted time, the original request fails and must be retried. | **5 SECONDS** | Yes | No |
| HRECALLWAITMAX | MAX HRECALL WAIT TIME FOR SWSALLOC.<br>When DFHSMHRECALL is used for dataset in-migration, this parameter determines the maximum HRECALL wait time which may be specified explicitly by an SWSALLOC application programming interface request using the RECALLWAIT() keyword.<br>If an individual SWSALLOC request attempts to specify a longer wait time limit than is imposed by this parameter, the value specified by this parameter is substituted. See HRECALLWAIT for a description of the HRECALL wait time limits. The maximum value allowed for this parameter is 32,767. | **45 SECONDS** | Yes | No |

# PRODGLV

| Parameter Name | Parameter Description | Default Value | Modifiable After Initialization | Output Only |
|---|---|---|---|---|
| GLOBALADDR | GLOBAL WORKSPACE BLOCK ADDRESS. | **X'0E9CA000' (SD)**<br>**X'0973800' (SWS)** | No | Yes |
| GLOBALALLOC | NUMBER OF ALLOCATED GLOBAL VARIABLE BLOCKS. | **0 (SD)**<br>**5 (SWS)** | No | Yes |
| GLOBALBACKUPCOUNT | GLOBAL VARIABLE BACKUP COUNT. | **0** | No | Yes |
| GLOBALBACKUPDSN (PARAMETER ON HOLD) | GLOBAL VARIABLE BACKUP DATASET NAME. | **ON HOLD** | Yes | No |
| GLOBALBACKUPEND | GLOBAL LAST BACKUP END TIME. | **NONE** | No | Yes |
| GLOBALBACKUPINTVAL (PARAMETER ON HOLD) | INTERVAL BETWEEN GLOBAL VARIABLE BACKUPS.<br>**Minimum Value: 0 Maximum Value: 32767** | **ON HOLD** | Yes | No |
| GLOBALBACKUPMDSCB (PARAMETER ON HOLD) | GLOBAL VARIABLE BACKUP DATASET GDG PATTERN. | **ON HOLD** | Yes | No |
| GLOBALBACKUPNEXT | GLOBAL BACKUP NEXT START TIME. | **NONE** | No | Yes |
| GLOBALBACKUPPROC | GLOBAL VARIABLE BACKUP PROC NAME. | **SDBBGVBK** | Yes | No |
| GLOBALBACKUPSTART | GLOBAL LAST BACKUP START TIME. | **NONE** | No | Yes |
| GLOBALBACKUPUNIT (PARAMETER ON HOLD) | GLOBAL VARIABLE BACKUP DATASET UNIT. | **ON HOLD** | Yes | No |
| GLOBALBLOCKS | GLOBAL CHECKPOINT BLOCK COUNT. | **313 PAGES (SD)**<br>**626 PAGES (SWS)** | No | Yes |
| GLOBALBLOCKSUSED | NUMBER OF GLOBAL VARIABLE BLOCKS IN USE. | **0 (SD)**<br>**5 (SWS)** | No | Yes |
| GLOBALCHECKCOUNT | GLOBAL CHECKPOINT COUNT. | **1 CHECKPOINT (SD)**<br>**2 CHECKPOINTS (SWS)** | No | Yes |

| Parameter Name | Parameter Description | Default Value | Modifiable After Initialization | Output Only |
|---|---|---|---|---|
| GLOBALCONTROL | GLOBAL WORKSPACE CONTROL LENGTH. | | No | Yes |
| GLOBALDATE | GLOBAL LAST CHECKPOINT DATE. | **NONE** | No | Yes |
| GLOBALDIV | GLOBAL VARIABLES SHOULD USE DIV. (YES, NO). | **YES** | No | No |
| GLOBALFREE | NUMBER OF FREE GLOBAL VARIABLE BLOCKS. | **0** | No | Yes |
| GLOBALFREEAREAS | NUMBER OF FREE AREAS IN GLOBAL WORKSPACE. | **0** | No | Yes |
| GLOBALINTERVAL | GLOBAL VARIABLES CHECKPOINT INTERVAL. **Minimum Value: 1 Maximum Value: 300** | **15 SECONDS** | Yes | No |
| GLOBALLENGTH | GLOBAL WORKSPACE BLOCK LENGTH. | **256 BYTES** | No | Yes |
| GLOBALMAX | MAXIMUM NUMBER OF GLOBAL VARIABLES. **Minimum Value: 1 Maximum Value: None** | **5000** | No | No |
| GLOBALMSGS | GLOBAL ERROR MESSAGE COUNT. | **0** | No | Yes |
| GLOBALNEXT | GLOBAL WORKSPACE NEXT FREE OFFSET. | **X’00000100’ (SD) X’00000600’ (SWS)** | No | Yes |
| GLOBALPAGES | GLOBAL WORKSPACE AREA SIZE IN PAGES. | **313 PAGES** | No | Yes |
| GLOBALPOOL | GLOBAL WORKSPACE FREE POOL OFFSET. | **X'00000000'** | No | Yes |
| GLOBALREBUILD | REBUILD GLOBAL VARIABLE DATABASE. | **NONE** | Yes | No |
| GLOBALRESTORETIME | LAST GLOBAL VARIABLE RESTORE TIME. | **ON HOLD** | No | Yes |
| GLOBALRETRY | GLOBAL CHECKPOINT RETRY COUNT. | **0 CHECKPOINTS** | No | Yes |
| GLOBALSIZE | GLOBAL WORKSPACE BLOCK SIZE. | **1250K** | No | Yes |
| GLOBALSUBPOOL | GLOBAL VARIABLES SUBPOOL NUMBER. | **TWO** | No | No |
| GLOBALTCB | GLOBAL WORKSPACE TCB ADDRESS. | **X'007F0748' (SD) X’0071E2A0’ (SWS)** | No | Yes |

| Parameter Name | Parameter Description | Default Value | Modifiable After Initialization | Output Only |
|---|---|---|---|---|
| GLOBALTEMPADDR | TEMPORARY GLOBAL WORKSPACE BLOCK ADDRESS. | **X’0EB0C000’ (SSD)<br>X’09874000 (SWS)** | No | Yes |
| GLOBALTEMPALLOC | NUMBER OF ALLOCATED TEMP GLOBAL VARIABLE BLOCKS. | | No | Yes |
| GLOBALTEMPBLKSUSED | NUMBER OF TEMP GLOBAL VARIABLE BLOCKS IN USE. | | No | Yes |
| GLOBALTEMPFREE | NUMBER OF FREE TEMP GLOBAL VARIABLE BLOCKS. | | No | Yes |
| GLOBALTEMPFREEAREAS | NUMBER OF FREE AREAS IN TEMP GLOBAL WORKSPACE. | | No | Yes |
| GLOBALTEMPMAX | MAXIMUM NUMBER OF TEMPORARY GLOBAL VARIABLES.<br>**Minimum Value: 1 Maximum Value: None** | **5000** | No | No |
| GLOBALTEMPUPDATE | TEMPORARY GLOBAL VARIABLES UPDATE COUNT. | **0** | No | Yes |
| GLOBALTEMPUSED | NUMBER OF TEMPORARY GLOBAL VARIABLES IN USE. | | No | Yes |
| GLOBALTEMPWARNIV | INTERVAL BETWEEN TEMP GLV BLOCKS USED WARNINGS.<br>**Minimum Value: 1 Maximum Value: 32767** | **5 MINUTES** | Yes | No |
| GLOBALTEMPWARNTH (PARAMETER ON HOLD) | TEMP GLOBAL BLOCKS USED WARNING THRESHOLD.<br>**Minimum Value: 1 Maximum Value: 100** | **ON HOLD** | Yes | No |
| GLOBALTIME | GLOBAL LAST CHECKPOINT TIME. | **NONE** | No | Yes |
| GLOBALTOKEN | GLOBAL WORKSPACE TOKEN ID. | **X'FF60FB5800000000' (SD)<br>X’FF6B9A3000000000’ (SWS)** | No | Yes |
| GLOBALUPDATE | GLOBAL VARIABLES UPDATE COUNT. | **0** | No | Yes |
| GLOBALUPDATECHECK | GLOBAL CHECKPOINT UPDATE COUNT. | **0 (SD)<br>1 (SWS)** | No | Yes |

| Parameter Name | Parameter Description | Default Value | Modifiable After Initialization | Output Only |
|---|---|---|---|---|
| GLOBALUSED | NUMBER OF GLOBAL VARIABLES IN USE. | **0 (SD)**<br>**5 (SWS)** | No | Yes |
| GLOBALWARNINTVAL | INTERVAL BETWEEN GLOBAL BLOCKS USED WARNINGS.<br>**Minimum Value: 1 Maximum Value: 32767** | **5 MINUTES** | Yes | No |
| GLOBALWARNTH | TEMP GLOBAL BLOCKS USED WARNING THRESHOLD.<br>**Minimum Value: 1 Maximum Value: 100** | **80%** | Yes | No |
| GLOBALWARNTHRESH | GLOBAL BLOCKS USED WARNING THRESHOLD.<br>**Minimum Value: 1 Maximum Value: 100** | **80%** | Yes | No |
| GLVCHAINMAX | MAXIMUM NUMBER OF CHAINED GLV UPDATES.<br>**Minimum Value: 1 Maximum Value: 32767** | **1000** | Yes | No |
| GLVPENDINGHIGH (PARAMETER ON HOLD) | HIGHEST NUMBER OF PENDING GLV EPROS. | **ON HOLD** | No | Yes |
| GLVPENDINGMAX | MAXIMUM NUMBER OF PENDING GLV EPROCS.<br>**Minimum Value: 1 Maximum Value: 32767** | **100** | No | No |
| GLVSHAREDFILE (PARAMETER ON HOLD) | SHARED VARIABLE VSAM DATASET NAME. | **ON HOLD** | Yes | No |

# PRODHTML

| Parameter Name | Parameter Description | Default Value | Modifiable After Initialization | Output Only |
|---|---|---|---|---|
| DEFAULTEXCIFORMURL | CICS/EXCI HTM DEFAULT FORM URL<br>This parameter specifies a default FORM URL to be used during the generation of HTML through the Shadow Server Mapping Facility. This form will be used to process the data on the HTML form. This parameter only controls HTML generated through the EXCI HTML generation option. | **/NEON/EXCINTRY** | Yes | No |
| DEFAULTGENFORMURL | GENERIC HTML DEFAULT FORM URL<br>This parameter specifies a default FORM URL to be used during the generation of HTML through the Shadow Server Mapping Facility. This form will be used to process the data on the HTML form. This parameter only controls HTML generated through the GEN HTML generation option. | **<%WWW. CURRENTURL%>** | Yes | No |
| DEFAULTIMSFORMURL | IMS HTML DEFAULT FORM URL<br>This parameter specifies a default FORM URL to be used during the generation of HTML through the Shadow Server Mapping Facility. This form will be used to process the data on the HTML form. This parameter only controls HTML generated through the IMS HTML generation option. | **/NEON/IMSENTRY** | Yes | No |
| GENERATEFONTORDERS | GENERATE HTML FONT COLOR ORDERS.<br>(YES, NO)<br>This parameter specifies that, when generating HTML and the color of the text or data can be determined, a corresponding HTML FONT color should be generated. This is used when generating IMS HTML for MFS source extracted through the Shadow Server Mapping Facility. | **YES** | Yes | No |

| Parameter Name | Parameter Description | Default Value | Modifiable After Initialization | Output Only |
|---|---|---|---|---|
| GENERATESTYLECOLOR | GENERATE STYLE COLORS FOR INPUT FIELDS.<br>(YES, NO)<br>This parameter specifies that, when generating HTML and the color of the text or data can be determined, a corresponding HTML FONT color order should be generated. This isused when generating IMS HTML for MFS source extracted through the Shadow Server Mapping Facility. | **YES** | Yes | No |
| HTMLBACKGROUNDCOLOR | HTML DEFAULT BACKGROUND COLOR.<br>This parameter specifies a default background color when generating HTML using the Shadow Server Mapping Facility. | **'000000'** | Yes | No |
| HTMLDEFAULTFONTCOLOR | HTML DEFAULT FONT COLOR.<br>This parameter specifies a default font color when generating HTML using the Shadow Server Mapping Facility. | **'00F500'** | Yes | No |
| HTMLFONTBLUE | REPLACE BLUE WITH THIS COLOR.<br>This parameter specifies the HTML color that is to replace 3270 display station BLUE. This is generally used to translate IMS Extended Attribute colors. | **'0000F5'** | Yes | No |
| HTMLFONTGREEN | REPLACE GREEN WITH THIS COLOR.<br>This parameter specifies the HTML color that is to replace 3270 display station GREEN. This is generally used to translate IMS Extended Attribute colors. | **'00F500'** | Yes | No |
| HTMLFONTHILIGHT | REPLACE HILIGHT FIELDS WITH THIS COLOR.<br>This parameter specifies the HTML color that is to replace highlighted fields on a 3270 display station. This value is generally used to translate IMS dynamic attributes. | **'FFFFFF'** | Yes | No |
| HTMLFONTPINK | REPLACE PINK WITH THIS COLOR.<br>This parameter specifies the HTML color that is to replace 3270 display station PINK. This is generally used to translate IMS Extended Attribute colors. | **'FF6EC7'** | Yes | No |

| Parameter Name | Parameter Description | Default Value | Modifiable After Initialization | Output Only |
|---|---|---|---|---|
| HTMLFONTRED | REPLACE RED WITH THIS COLOR.<br>This parameter specifies the HTML color that is to replace 3270 display station RED. This is generally used to translate IMS Extended Attribute colors. | **'F50000'** | Yes | No |
| HTMLFONTTURQUOISE | REPLACE TURQUOISE WITH THIS COLOR.<br>This parameter specifies the HTML color that is to replace 3270 display station TURQUOISE. This is generally used to translate IMS Extended Attribute colors. | **'ADEAEA'** | Yes | No |
| HTMLFONTWHITE | REPLACE WHITE WITH THIS COLOR.<br>This parameter specifies the HTML color that is to replace 3270 display station WHITE. This is generally used to translate IMS Extended Attribute colors. | **'FFFFFF'** | Yes | No |
| HTMLFONTYELLOW | REPLACE YELLOW WITH THIS COLOR.<br>This parameter specifies the HTML color that is to replace 3270 display station YELLOW. This is generally used to translate IMS Extended Attribute colors. | **'FFFF00'** | Yes | No |
| HTMLINPUTHILIGHT | REPLACE HILIGHT INPUT FONT WITH THIS COLOR.<br>This parameter specfies the HTML color that is to replace highlighted font input fields on a 3270 display station. Input fields on a Web Browser have a white background. This value is generally used to translate IMS dynamic attributes. | **'F50000'** | Yes | No |
| HTMLINPUTWHITE | REPLACE WHITE INPUT FONT WITH THIS COLOR.<br>This parameter specifies the HTML color that is to replace white font input fields on a 3250 display station. Input fields on a Web Browser have a white background. This value is generally used to translate IMS dynamic attributes. | **'F50000'** | Yes | No |

# PRODIMS (PRODIMCO)

| Parameter Name | Parameter Description | Defaultl Value | Modifiable After Initialization | Output Only |
|---|---|---|---|---|
| APPC/IMS | INITIALIZE APPC/IMS SUPPORT.<br>(YES, NO) | **YES** | No | No |
| BMP | INITIALIZE BMP SUPPORT.<br>(YES, NO). | **NO** | No | No |
| BMPALTTCB | STORE THE ALTERNATE TCB ADDRESS.<br>(YES, NO). | **NO** | Yes | No |
| BMPNAME | IMS BMP REGION CONTROLLER<br>(Used with Shadow Direct) | **DFSRRC00** | Yes | No |
| BMPPARM | IMS BMP PARAMETER STRING<br>(Used with Shadow Direct) | **NONE** | Yes | No |
| CONVERTNULLS | CONVERT NULLS TO BLANKS.<br>This parameter controls the conversion of null bytes to blanks for the IMS Transaction Server. IMS messages may contain MID/MOD indicators which need to be converted to blanks. | **NO** | Yes | No |
| DBCTL | INITIALIZE DBCTL SUPPORT.<br>(YES, NO) | **YES** | No | No |
| IMSLTERMTABLESEQ | LTERM ASSIGNMENT TABLE SEQUENCE<br>This parameter allows the user to control LTERM assignments based upon userid or TCP/IP address when initiating transactions to IMS. Valid options are: userid (userid match will determine the LTERM name), IP address (TCP/IP address will determine the LTERM name), none (do not assign an LTERM) | **NONE** | Yes | No |
| IMSBMPTIMEOUT | IMS BMP READ TIMEOUT VALUE IN SECONDS<br>(Used with Shadow Direct) | **60** | Yes | No |
| IMSBMPWAITTIME | IMS BMP WAIT TIME VALUE.<br>**Minimum Value: 0 Maximum Value: 10000** | **100 MILLISECONDS** | Yes | No |

| Parameter Name | Parameter Description | Defaultl Value | Modifiable After Initialization | Output Only |
|---|---|---|---|---|
| IMSCLASS | SNAP DUMP SYSOUT OUTPUT CLASS. | **'A'** | Yes | No |
| IMSDDNAME | DDNAME used to allocate RESLIB. | **'CCTLDD'** | Yes | No |
| IMSDLIPRMLOC | IMS DLI PARAMETER LIST LOCATION. | **ABOVE** | Yes | No |
| IMSDSNAME | DSNAME OF THE DRA RESLIB. | **'IMS.RESLIB'** | Yes | No |
| IMSFPBUFFERS | FAST PATH BUFFERS PER THREAD. | **0** | Yes | No |
| IMSFPOVERFLOW | FAST PATH OVERFLOW BUFFERS. | **0** | Yes | No |
| IMSFUNCLEVEL | FUNCTION LEVEL OF PRODUCT REGION. | **X'01'** | Yes | No |
| IMSGROUPNAME | APPLICATION GROUP NAME. | **'NONE'** | Yes | No |
| IMSID | IMSID OF THE DBCTL REGION. | **'IMSP'** | No | No |
| IMSMAXTHREADS | MAXIMUM NUMBER OF THREADS.<br>This parameter is the maximum number of allowed DTB threads to be active at one time. | **30 (SD)**<br>**5 (SWS)** | Yes | No |
| IMSMINTHREADS | MINIMUM NUMBER OF THREADS<br>This parameter is set to the desired amount of DBT threads to open initially when Shadow Connects to IMS. | **1** | Yes | No |
| IMSNBABUFFERS | TOTAL NUMBER OF NBA BUFFERS. | **0** | Yes | No |
| IMSOTMA | INITIALIZE IMS/OTMA SUPPORT.<br>(YES, NO)<br>This parameter is used to control the initialization of ISM/OTMA support. | **NO** | No | No |
| IMSPROCOWNER | IMS STORED PROCEDURE OWNER.<br>This parameter allows the user to specify the procedure owner for IMS stored procedure map. | **'IMSTM'** | Yes | No |
| IMSSUFFIX | SUFFIX OF THE DFSPRPXX MODULE. | **'00'** | Yes | No |

| Parameter Name | Parameter Description | Defaultl Value | Modifiable After Initialization | Output Only |
|---|---|---|---|---|
| IMSTIMEOUT | DRA TERM TIMEOUT VALUE. | **10** | Yes | No |
| IMSUSERID | USERID OF THE PRODUCT REGION. | **NULL** | Yes | No |
| IMSWAITTIME | IDENTIFY RETRY WAIT TIME. | **60** | Yes | No |

# PRODLICENSE (PRODLICN)

| Parameter Name | Parameter Description | Default Value | Modifiable After Initialization | Output Only |
|---|---|---|---|---|
| CONNECTIONTEXT | HOST CONNECTION TEXT STRING. | **NULL (SD)**<br>**'SHADOW OS/390 WEB SERVER I' (SWS)** | Yes | No |
| CURRENTCPU | CURRENT CPU ID. | **'00069'** | No | No |
| DB2CONCURRENTCN | CONCURRENT DB2 USER COUNT. | **0** | No | No |
| DB2CONCURRENTHW | CONCURRENT DB2 USER HI-WATER MARK. | **5 (SD)**<br>**0 (SWS)** | No | No |
| DB2CONCURRENTMX | MAXIMUM CONCURRENT DB2 USERS.<br>**Minimum Value: 0 Maximum Value: 2000** | **2000** | Yes | No |
| DB2LICCONCURMX | MAXIMUM LICENSED DB2 USERS. | **2000** | No | No |
| EXPIRATIONDATE | PRODUCT EXPIRATION DATE. | **2020/12/31** | No | No |
| EXPIRATIONDAYS | DAYS PRIOR TO EXPIRATION. | **8121 (SD)**<br>**8086 (SWS)** | No | No |
| FIRSTCPU | FIRST LICENSED CPU ID. | **'00069' (SD)**<br>**'41230' (SWS)** | No | No |
| LICENSECODE | PRODUCT LICENSE CODE STRING. | **45R07S83Y3UA9272BYX (SD)**<br>**45D14UA4X1WCOPI3FBZ (SWS)** | No | No |

| Parameter Name | Parameter Description | Default Value | Modifiable After Initialization | Output Only |
|---|---|---|---|---|
| MAXLICENSEDCLIENTS | MAXIMUM NUMBER OF LICENSED CLIENTS.<br>This parameter is used to display the maximum number of client systems that the client software can be installed on. This parameter cannot be set using the initialization REXX program and is obtained from the product license string. If the maximum number of licensed clients has been reached, then any attempt to connect from a new client system will be rejected with an error message. If this field is zero, then there is no limit on the number of client systems that can connect. | **0** | No | No |
| OEMVENDOR | OEM VENDOR NAME STRING. | **'OEM VENDOR'** | Yes | No |
| PRODFAMILY | PRODUCT FAMILY CODE PREFIX. | **'45'** | No | No |
| PRODFEATURES | PRODUCT FEATURE CODE STRING. | **'A C E G I K M'** | No | No |

# PRODLOGGING (PRODLOG)

| Parameter Name | Parameter Description | Default Value | Modifiable After Initialization | Output Only |
|---|---|---|---|---|
| LOGAPMVSSUM | LOG APPC/MVS SUMMARY INFO IN A TABLE.<br>This parameter controls if APPC/MVS interval summary information should be logged odr not. APPC/MVS interval summary information is logged by inserting rows into a DB2 table. One row is inserted at the end of each recording level. | **NO** | Yes | No |
| LOGAPMVSSUMTABLE | TABLE NAME FOR APPC/MVS SUMMARY LOGGING.<br>This parameter is used to set the name of the DB2 table used to log APPC/MVS interval summary information. A row is inserted into this table at the end of each recording interval, if APPC/MVS interval summary recording is active. | **'SHADOW.APMVSSUM'** | Yes | No |
| LOGDB2PLNAME | DB2 PLAN NAME FOR LOGGING OPERATION.<br>This parameter controls the plan name used for all SQL operations initiated by Shadow to log performance data. If this parameter is set, then all logging operations will use the specified name. If this field is not set, then each logging operation will use the default DB2 planname, set by DEFAULTDB2PLAN. | **'SDBUDAD0' (SD)**<br>**NULL (SWS)** | Yes | No |
| LOGDB2SUBSYS | DB2 SUBSYSTEM FOR LOGGING OPERATIONS.<br>This parameter controls the DB2 subsystem used for all SQL operations. If this parameter is set, then all logging operations will be routed to the specified DB2 subsystem. If this field is not set, then each logging operation will be routed to the DB2 subsystem that the operation was associated with or the default DB2 subsystem if the operation was not associated with any DB2 subsystem. | **'DB2B'** | No | No |
| LOGDELAY | LOG DELAY TIME INTERVAL.<br>This parameter controls how long the logging task will delay after it completes processing some set of logging requests. This is done to avoid too many starts and stops with the associated overhead of connecting to DB2 and then releasing the DB2 connection.<br>**Minimum Value: 1 Maximum Value: 300** | **10 SECONDS (SD)**<br>**30 SECONDS (SWS)** | Yes | No |

| Parameter Name | Parameter Description | Default Value | Modifiable After Initialization | Output Only |
|---|---|---|---|---|
| LOGEXCEPTIONSTABLE | TABLE NAME FOR SQL EXCEPTIONS.<br>This parameter is used to set the name of the DB2 table used to log SQL operations that need to be logged. An SQL operation may need to be logged because it used too much CPU time of because it failed. | **'SHADOW. SQLEXCEPTION'** | Yes | No |
| LOGFAILURELIMIT | LOGGING FAILURE LIMIT.<br>This parameter controls how many logging requests can be pending before a failure exception will occur. Failure exceptions are passed to SEF (if enabled) for processing. If SEF is not enabled, or if there are no SEF rules for the logging failure exception, or if the SEF rules take no action, the default action will be taken. The defaul action is sto clear the queue of pending logging requests and discard all of them.<br>**Minimum: 0 Maximum: 100000** | **10,000 REQUESTS** | Yes | No |
| LOGINTERVALS | LOG EACH INTERVAL IN A TABLE.<br>(YES, NO)<br>This parameter controls if session interval information should be logged or not. Session interval information is logged by inserting rows into a DB2 table. One row is inserted for each session at the end of each recording interval and at session termination time. | **YES (SD)**<br>**NO (SWS)** | Yes | No |
| LOGINTERVALSTABLE | TABLE NAME FOR INTERVAL LOGGING.<br>This parameter is used to set the name of the DB2 table used to log interval information. A row is inserted into this table at the end of each recording interval, if interval recording is active. | **'SHADOW. INTERVALS'** | Yes | No |
| LOGMEMORYAPPC/MVS | IN MEMORY APPC/MVS INTERVAL COUNT.<br>This parameter controls the number of APPC/MVS summary records to keep in memory at one time. If this parameter is set to zero, then no APPC/MVS summary records will be retained in memory. Setting this parameter to zero will not prevent APPC/MVS interval recording from being performed. The APPC/MVS summary records kept in memory can be interactively displayed.<br>**Minimum Value: 0 Maximum Value: 1000** | **500 INTERVALS** | Yes | No |

| Parameter Name | Parameter Description | Default Value | Modifiable After Initialization | Output Only |
|---|---|---|---|---|
| LOGMEMORYINTERVALS | IN MEMORY SUMMARY INTERVAL COUNT.<br>This parameter controls the number of interval summary records to keep in memory at one time. If this parameter is set to zero, then no interval summary records will be retained in memory. Setting this parameter to zero will not prevent interval recording from being performed. The interval summary records kept in memory can be interactively displayed.<br>**Minimum Value: 0 Maximum Value: 1000** | **3 INTERVALS (SD)**<br>**200 INTERVALS (SWS)** | Yes | No |
| LOGRETAINAPMVSSUM | LOG APPC/MVS SUMMARY RETENTION PERIOD.<br>This parameter controls the number of days to wait before automatically deleting rows from the APPC/MVS summary table. In other words, all rows older than the number of days will be deleted. If this value is zero, then rows will never be automatically delelted from the APPC/MVS summary table.<br>**Minimum Value: 0 Maximum Value: 999999** | **0 DAYS** | Yes | No |
| LOGRETAININTERVALS | LOG INTERVAL RETENTION PERIOD.<br>This parameter controls the number of days to wait before automatically deleting rows from the interval summary table. In other words, all rows older than the number of days will be deleted. If this value is zero, then rows will never be automatically deleted from the interval summary table.<br>**Minimum Value: 0 Maximum Value: 999999** | **30 DAYS (SD)**<br>**0 DAYS (SWS)** | Yes | No |
| LOGRETAINSESSIONS | LOG SESSION RETENTION PERIOD.<br>This parameter controls the number of days to wait before automatically deleting rows from the sessions table. In other words, all rows older than the number of days will be deleted. If this value is zero, then rows will never be automatically deleted from the sessions table.<br>**Minimum Value: 0 Maximum Value: 999999** | **30 DAYS (SD)**<br>**0 DAYS (SWS)** | Yes | No |

| Parameter Name | Parameter Description | Default Value | Modifiable After Initialization | Output Only |
|---|---|---|---|---|
| LOGRETAINSQL | LOG SQL SOURCE RETENTION PERIOD.<br>This parameter controls the number of days to wait before automatically deleting SQL from the SQL source table. In other words, all rows older than the number of days will be deleted. If this value is zero, then rows will never be automatically deleted from the SQL source table.<br>**Minimum Value: 0 Maximum Value: 999999** | **30 DAYS (SD)**<br>**0 DAYS (SWS)** | Yes | No |
| LOGRETAINURLS | LOG URLS RETENTION PERIOD.<br>This parameter controls the number of days to wait before automatically deleting rows from the URLs table. In other words, all rows older than the number of days will be deleted. If this value is zero, then rows will never be automatically deleted from the URLs table.<br>**Minimum Value: 0 Maximum Value: 999999** | **30 DAYS** | Yes | No |
| LOGSESSIONS | LOG EACH SESSION IN A TABLE.<br>(YES, NO)<br>This parameter controls if session information should be logged or not. Session information is logged by inserting rows into a DB2 table. One row is inserted for each session at session termination time. | **YES (SD)**<br>**NO (SWS)** | Yes | No |
| LOGSESSIONSTABLE | TABLE NAME FOR SESSION LOGGING.<br>This parameter is used to set the name of the DB2 table used to log session information. A row is inserted into this table as part of session termination, if session logging is active. | **'SHADOW. SESSIONS'** | Yes | No |
| LOGSOURCETABLE | TABLE NAME FOR SQL SOURCE.<br>This parameter is used to set the name of the DB2 table used to log SQL source for conversion from dynamic SQL to static SQL. Each SQL statement is stored in one or more rows of this table. | **'SHADOW. SQLSOURCE'** | Yes | No |

| Parameter Name | Parameter Description | Default Value | Modifiable After Initialization | Output Only |
|---|---|---|---|---|
| LOGSQLSOURCE | LOG SQL SOURCE IN A TABLE.<br>(YES, NO)<br>This parameter controls if SQL source information should be logged or not. SQL source information is logged by inserting rows into a DB2 table. One row is inserted for each SQL statement when the SQL statement is processed. The logged SQL source is used to convert dynamic SQL to static SQL. | **YES (SD)**<br>**NO (SWS)** | Yes | No |
| LOGSTORAGE | LOG STORAGE USAGE IN A TABLE.<br>(YES, NO)<br>This parameter controls if storage information should be logged or not. Storage information is logged by inserting rows into a DB2 table. | **YES (SD)**<br>**NO (SWS)** | Yes | No |
| LOGSTORAGETABLE | TABLE NAME FOR STORAGE LOGGING.<br>This parameter is used to set the name of the DB2 table used to log storage information. A row is inserted into this table at the end of each recording level, if storage logging is active. | **'SHADOW.STORAGE'** | Yes | No |
| LOGURLS | LOG URLS IN A TABLE.<br>(YES, NO)<br>This parameter controls if URLs should be logged or not. URL information is logged by inserting rows into a DB2 table. One row is inserted for each URL when the URL is processed. The logged URL information can be used for any installation purpose. | **NO** | Yes | No |
| LOGURLSTABLE | TABLE NAME FOR URL LOGGING.<br>This parameter is used to set the name of the DB2 table used to log URLs. A row is inserted into this table as part of the processing of each URL, if URL logging is active. | **'SHADOW.URLS'** | Yes | No |
| LOGUSERID | USERID FOR ALL LOGGING OPERATIONS.<br>This parameters controls the DB2 userid used for all SQL operations. This userid must have enough authority to update (insert) all of the tables modified by the logging task. If this field is not set, the main product address space userid is used for all update operations. | **'SDBD' (SD)**<br>**'SWSS' (SWS)** | Yes | No |

| Parameter Name | Parameter Description | Default Value | Modifiable After Initialization | Output Only |
|---|---|---|---|---|
| LOGWAIT | LOG WAIT TIME INTERVAL.<br>This parameter controls how long the logging task will wait when there is no work to do. When this interval expires some general work (such as deleting obsolete rows) may be executed.<br>**Minimum Value: 60 Maximum Value: 43200** | **60 SECONDS (SD)**<br>**3600 SECONDS (SWS)** | Yes | No |
| LOGWARNINGLIMIT | LOGGING WARNING LIMIT.<br>This parameter controls how many logging requests can be pending before a warning exception will occur. Warning exceptions are passed to SEF (if enabled) for processing. If SSEF is not enabled, or if there are no SEF rules for the logging warning exception, or if the SEF rules take no aciton, the default action will be taken. The default action is to issue an error message describing the exception to the system console.<br>**Minimum Value: 0 Maximum Value: 100000** | **5000 REQUESTS** | Yes | No |
| RECORDINGINTERVAL | INTERVAL RECORDING PERIOD.<br>This parameter controls how often interval summary and per-client SMF and/or SQL records are created. These records show what resources where used during the current recording interval. The interval value is specified in seconds and should be a factor of one hour. In other words the value should divide evenly into 3600.<br>**Minimum Value: 1 Maximum Value: 3600** | **300 SECONDS (SD)**<br>**900 SECONDS (SWS)** | Yes | No |
| TERMINATELOGGING | TERMINATE LOGGING PROCESSING.<br>(YES, NO)<br>This parameter controls if logging processing should terminate or not. If this parameter is turned on, logging processing will end and can not be restarted. This parameter can be set at any time and will always terminate logging processing. | **NO** | Yes | No |

# PRODPARM

| Parameter Name | Parameter Description | Default Value | Modifiable After Initialization | Output Only |
|---|---|---|---|---|
| ADJUSTREGIONSIZE | AUTO-ADJUST TSO USER REGION SIZE.<br>(YES, NO)<br>This parameter allows this address space to automatically adjust the Region Size of TSO users connecting to the Shadow Server address space. | **YES** | Yes | No |
| AUTOCANCELTM | AUTOMATIC CANCEL AT PRODUCT TERMINATION.<br>(YES, NO)<br>This parameter indicates if client processing subtasks will be cancelled by the Server during shutdown, following the CLIENTQUIESCEDELAY time (if any). If this parameter is set to NO, client processing subtasks are abandoned by the server at the end of the SHUTDOWNWAIT time period and the product's main task may be terminated by the system within an SA03 abend. | **YES** | Yes | No |
| BASEINTERVAL | BASE TIME SLICE INTERVAL.<br>Used with the time slicing mechanism.<br>**Minimum Value: 0 Maximum Value: 1000000** | **0 MILLISECONDS** | Yes | No |
| CANCELWAITTIME | CLIENT CANCEL WAIT TIME VALUE.<br>This parameter controls how long the product waits between client thread termination events during product shutdown. The product automatically terminates client threads during product termination. Some IBM products cannot handle large number of thread termination events in a short period of time. To prevent problems, the product throttles client thread terminations. The CANCELWAITTIME parameter is the delay between each client thread termination initiated by the product.<br>**Minimum Value: 0 Maximum Value: 10000** | **3000 MILLISECONDS** | Yes | No |

| Parameter Name | Parameter Description | Default Value | Modifiable After Initialization | Output Only |
|---|---|---|---|---|
| CHECKLIMITSINTERVAL | CPU/WAIT LIMITS CHECKING INTERVAL.<br>This controls how often each client task is checked for a violation of any execution limit. The interval value is specified in seconds and should be a factor of one hour. In other words the value should divide evenly into 3600.<br>**Minimum Value: 1 Maximum Value: 3600** | **15 SECONDS** | Yes | No |
| CHECKDATAINTERVAL | KEY DATA CHECKING INTERVAL.<br>This parameter controls how often certain key data fields are checked for consistency and validity. If any of these fields are found to be in error, it is fixed so that normal product execution can be continued. The interval value is specified in seconds and should be a factor of one hour. In other words the value should divide evenly into 3600.<br>**Minimum Value: 1 Maximum Value: 3600** | **60 SECONDS** | Yes | No |
| CHECKSESSIONS | CHECK THE STATUS OF EACH SESSION.<br>(YES, NO)<br>Controls if communication session is checked on a periodic basis. If set to yes and it detects a session terminated because the client application terminated, the client system failed, or because of a network failure, then all work running on the host running on behalf of the client is terminated. | **NO** | Yes | No |
| CLIENTQUIESCEDELAY | CLIENT TASK QUIESCE DELAY.<br>This parameter controls how long the product waits during shutdown for client processing subtasks to end normally. This delay time value is only used when AUTOCANCELTM is set to YES. It can be used to throttle overall product shutdown processing to allow sufficient time for transaction threads to terminate normally before they are cancelled using CALLRTM. This quiesce delay occurs before the SHUTDOWNWAIT time interval begins.<br>**Minimum Value: 0 Maximum Value: 1800** | **10 SECONDS** | Yes | No |

| Parameter Name | Parameter Description | Default Value | Modifiable After Initialization | Output Only |
|---|---|---|---|---|
| DBCSTABLENAME | DEFAULT DBCS TABLE NAME.<br>This parameter allows the user to define a default DBCS Table for DBCS character translation. | **NULL** | Yes | No |
| DEBUGOPTIONS | DEBUGGING CONTROLLER OPTIONS. | **NONE** | Yes | No |
| DEFAULTCPUTIME | DEFAULT DEFAULT CPU TIME.<br>The default CPU time value (0-600), in seconds, that is used with the internal CPU time limit mechanism if a default value cannot be obtained from the security package (ACF2 or RACF). See "Setting a CPU Time Limit for ODBC Clients" in section 3.3. | **0 SECONDS** | Yes | No |
| DISPATCH | MAIN ADDRESS SPACE DISPATCH PRIORITY.<br>If this parameter is set to zero, then the product will not attempt to set its dispatch priority.<br>**Minimum Value: 0 Maximum Value: 255** | **80** | No | No |
| DLLIBDDNAME | DIRECTED LOAD DDNAME. | **NONE** | No | No |
| DSPC | INITIALIZE DSPC SUPPORT.<br>(YES, NO)<br>This parameter controls whether or not the DSPC support is initialized. | **NO** | No | No |
| ERRORCPUTIME | ERROR CPU TIME VALUE.<br>Determines the error limit (in seconds) of the external CPU time limit mechanism. | **0 SECONDS** | Yes | No |
| ERRORWAITTIME | ERROR WAIT TIME VALUE.<br>Determines the error limit (in seconds) of the external wait time limit mechanism. | **0** | Yes | No |
| EXECDSNAME | REXX EXEC DATA SET NAME. | **'CSD.AI38.<br>SV040100E'** | Yes | No |
| EXTRAINTERVAL | EXTRA TIME SLICE INTERVAL.<br>Used with the time slicing mechanism.<br>**Minimum Value: 1 Maximum Value: 10000** | **0 MILLISECONDS** | Yes | No |

| Parameter Name | Parameter Description | Default Value | Modifiable After Initialization | Output Only |
|---|---|---|---|---|
| FAILCPUTIME | FAIL CPU TIME VALUE.<br>This value determines the failure limit (in seconds) of the external CPU time limit mechanism. | **600 SECONDS** | Yes | No |
| FAILEXCLUSIVETIME | FAIL EXCLUSIVE LOCK TIME VALUE. | **0 SECONDS** | Yes | No |
| FAILSHARETIME | FAIL SHARE LOCK TIME VALUE. | **0** | Yes | No |
| FAILSQLCPUTIME | FAIL SQL CPU TIME VALUE. | **360 SECONDS** | Yes | No |
| FAILUPDATETIME | FAIL UPDATE LOCK TIME VALUE. | **0 SECONDS** | Yes | No |
| FAILWAITTIME | FAIL WAIT TIME VALUE.<br>This value determines the failure limit (in seconds) of the external wait time limit mechanism. | **0 SECONDS** | Yes | No |
| GROUPDIRECTOR | PERFORM GROUP DIRECTOR ROLE.<br>(YES, NO)<br>This parameter indicates that a member of the grop take the role of director. The director will only accept inbound connections and pass them to a member of the grop which is determined to be the most accepatable in terms of load and resource availability. The group director will not support an application execution environment. This wil lprovide for a more robust load balancing group. | **NO** | Yes | No |
| GROUPNAME | LOAD BALANCING GROUP NAME.<br>This parameter controls which group, if any, the current copy of the server belongs to. Groups are used for load balancing across multiple copies (separate subsystems) of the product. All copies that belong to the same group (i.e., have exactly the same GROUPNAME) automatically load balance between each other. If this value is not set, then the current copy does not belong to any group. | **NULL** | Yes | No |

| Parameter Name | Parameter Description | Default Value | Modifiable After Initialization | Output Only |
|---|---|---|---|---|
| HIGHMODULEDATE | HIGH MODULE ASSEMBLE DATE.<br>This field contains the assemble date of the module that was assembled latest in the product. This field is provided for technical support purposes and can not be changed. | | No | Yes |
| HIGHMODULETIME | HIGH MODULE ASSEMBLE TIME.<br>This field contains the assemble time of the module that was assembled latest in the product. This field is provided for technical support purposes and cannot be changed. | | No | Yes |
| HIGHMODULENAME | HIGH MODULE NAME.<br>This field contains the name of the module that was assembled latest in the product. This field is provided for technical support purposes and cannot be changed. | **'OPCOSK'** | No | Yes |
| HIGHMODULEVERSION | HIGH MODULE VERSION.<br>This field contains the version of the module that was assembled latest in the product. This field is provided for technical support purposes and cannot be changed. | **'04.01.00'** | No | Yes |
| ISPLLIBDSNAME | ISPLLIB DATA SET NAME. | **'CSD.AI38.SV040100.L'** | Yes | No |
| ISPMLIBDSNAME | ISPMLIB DATA SET NAME. | **'CSD.AI38.SV040100.N'** | Yes | No |
| ISPPLIBDSNAME | ISPPLIB DATA SET NAME. | **'CSD.AI38.SV040100.N'** | Yes | No |
| ISPSLIBDSNAME | ISPSLIB DATA SET NAME. | **NULL** | Yes | No |
| ISPTLIBDSNAME | ISPTLIB DATA SET NAME. | **'CSD.AI38.SV040100.N'** | Yes | No |

| Parameter Name | Parameter Description | Default Value | Modifiable After Initialization | Output Only |
|---|---|---|---|---|
| KILLWAITPOST | KILL WAITING THREADS WITH POST.<br>(YES, NO)<br>This parameter controls how threads that have exceeded a wait limit are killed. Only threads that have exceeded a wait limit are influenced by this parameter. If this parameter is set to YES, then the thread is terminated by posting the pending thread with a code that ends the pending network read operation and rolls back any database changes. If this parameter is set to NO, then the thread is killed with either a system or user abend. | **YES** | Yes | No |
| LERUNTIMEOPTS | LE/370 ENCLAVE RUN-TIME OPTIONS. | **NULL** | Yes | No |
| MAXABENDRATE | MAXIMUM ABEND RATE ALLOWED.<br>This parameter should be set to zero to turn off abend rate checking within Shadow. If it has a non-zero value, the value set will be used against the rate to determine if Shadow should terminate.<br>**Minimum: 0 Maximum: 1** | **0.1** | Yes | No |
| MAXCMDRATE | MAX COMMAND RATE ALLOWED.<br>This parameter should be set to zero to turn off the command rate checking within Shadow. If a non-zero value is used, the value set will be used against the rate to determine if Shadow should terminate.<br>**Minimum Value: 0 Maximum Value: 3** | **3.0** | Yes | No |
| MAXCPUTIME | DEFAULT MAXIMUM CPU TIME.<br>This is the maximum CPU time value, in seconds, used with the internal CPU time limit mechanism. | **0 SECONDS** | Yes | No |
| MAXLOGRATE | MAX LOGREC RATE ALLOWED.<br>This parameter should be set to zero to turn off logging rate checking within Shadow. If a non-zero value is used, the value sset will be used against the rate to determine if Shadow should terminate logrec recording dduring estae processing. | **0.01** | Yes | No |

| Parameter Name | Parameter Description | Default Value | Modifiable After Initialization | Output Only |
|---|---|---|---|---|
| MAXMSGRATE | MAX MESSAGE RATE ALLOWED.<br>This parameter should be set to zero to turn off the message rate checking within Shadow. If a non-zero value is used, the value set will be used against the rate to determine if Shadow should terminate.<br>**Minimum Value: 0 Maximum Value: 10.0** | **10.0** | Yes | No |
| MINCPUTIME | DEFAULT MINIMUM CPU TIME.<br>This is the minimum CPU time value, in seconds, used with the internal CPU time limit mechanism. | **0 SECONDS** | Yes | No |
| PROCESS | INITIAL PROCESS BLOCK COUNT.<br>This parameter needs to be equal to IMSMAXTHREADS plus the number of users that will be using the Shadow ISPF/SDF dialogs.<br>**Minimum Value: 5 Maximum Value: 250** | **10 BLOCKS** | No | No |
| PROCESSEP | PROCESS A SET OF ENTRY POINTS.<br>(YES, NO)<br>This parameter specifies whether or not a set of entry points should be processed. This option is for System Engineering use and should only be used when directed by Technical Support. | **NO** | | |
| PROCESSPC | PROCESS A SET OF PCS.<br>(YES, NO)<br>This parameter specifies whether or not a set of PCs should be processed. This option is for System Engineering use and should only be used when directed by Technical Support. | **NO** | No | No |
| PROCESSVC | PROCESS A SET OF SVCS.<br>(YES, NO)<br>This parameter specifies whether or not a set of SVSs should be processed. This option is for System Engineering use and should only be used when directed by Technical Support. | **NO** | No | No |

| Parameter Name | Parameter Description | Default Value | Modifiable After Initialization | Output Only |
|---|---|---|---|---|
| PROCESSTCB | TCB TO BE MONITORED.<br>This parameter specifies the address of a TCB that should be monitored by a set of routines. If this value is not set, then all TCBs will be monitored by these routines. If this value is set, then only one TCB will be processed by these routines. This option is for System Engineering use and should only be used when directed by Technical Support. | **X'00000000'** | Yes | No |
| QUICKREFOPTIONS | QUICKREF INVOCATION OPTIONS. | **CMD** | Yes | No |
| QUIESCESYSTEMTYPE | QUIESCE SYSTEM TYPE.<br>This parameter is used to indicate whether the termination of all client connections is to be performed immediately, or through attrition. | **ATTRITION** | Yes | No |
| RESETCONTROLINTERVAL | RESET CONTROL INTERVAL.<br>(YES, NO) | | No | No |
| REUSETHREADS | REUSE SESSION THREADS.<br>(YES, NO)<br>This parameter controls if threads should be reused or not. If this flag is set, each thread will be reused a number of times if possible. If this flag is not set, a new thread will always be created for each new inbound session. Thread reuse may reduce CPU resource utilization quite considerably when DB2 threads are used frequently and/or client userids are cached and reused for persistent session support. | **NO** | Yes | No |

| Parameter Name | Parameter Description | Default Value | Modifiable After Initialization | Output Only |
|---|---|---|---|---|
| SESSIONFAILTIME | SESSION FAILURE TIME LIMIT VALUE.<br>This parameter controls how long a remote application task (a task running on behalf of a client) can be in processing state (RPC, SQL, REXX) before the product will check if the network session is still active or not. In some cases, a remote client application will start some long running processing (for example a complex SQL statement) and then the remote application will end or the client system will fail or the network will fail. In any of these cases, the SESSIONFAILTIME parameter control how long before the product checks to see if the network session with the remote client system is still active or not. | **15 SECONDS** | Yes | No |
| SESSIONQUEUEADDRESS | SESSION TRANSFER QUEUE ADDRESS.<br>This parameter displays the address of the session transfer queue header. This parameter is used for display purposes only. | **X'0E84C000'** | No | Yes |
| SHUTDOWNWAIT | SHUTDOWN WAIT TIME VALUE.<br>This parameter controls how long the product will wait to shutdown. This is actually the number of seconds that the main product task will wait for all of its subtasks to terminate.<br>**Minimum Value: 0 Maximum Value: 3600** | **60 SECONDS** | Yes | No |
| SWIURLNAME | SHADOW WEB INTERFACE URL NAME.<br>The SWIURLNAME field is the name of the URL that will be recognized to indicate a desire to use the Shadow Web Interface. To activate the Shadow Web Interface, it is necessary to direct a browser to the port being used by a copy of the product and use the URL composed of the http://domain:port/swiurl (where swiurl is the SWIURLNAME parameter). | **SWICNTL** | Yes | No |
| TARGETTHREADCOUNT | TARGET UDP/TCP THREAD COUNT.<br>This parameter controls the target number of threads in some UDP and TCP execution modes. The value controls the number of subtasks created during product startup to handle inbound UDP datagrams and TCP sessions.<br>**Minimum Value: 1 Maximum Value: 1000** | **100 THREADS** | No | No |

| Parameter Name | Parameter Description | Default Value | Modifiable After Initialization | Output Only |
|---|---|---|---|---|
| TERMINATEINTERVAL | TERMINATE INTERVAL PROCESSING.<br>(YES, NO)<br>This parameter controls if interval processing should terminate or not. If this parameter is turned on, interval processing will end and can not be restarted. This parameter can be set at any time and will always terminate interval processing. | **NO** | Yes | No |
| THREADTIMEOUT | THREAD TIMEOUT WAIT TIME.<br>This parameter controls how long a thread will wait for new work to be assigned to it. When the time limit is reached the thread terminates. Setting too small a value will cause thread churning. Setting too high a value may leave too many idle threads.<br>**Minimum Value: 1 Maximum Value: 3600** | **300 SECONDS** | Yes | No |
| THREADREUSELIMIT | THREAD REUSE LIMIT VALUE.<br>This parameter controls how many times a thread can be used to handle a session before it terminates. Setting a value too small will cause additional CPU resources to be used. Setting a value too high may cause storage leakage. Note that a zero or one value will prevent all thread reuse.<br>**Minimum Value: 0 Maximum Value: 10000000** | **1000 SESSIONS** | Yes | No |
| TRACEBROWSECOUNT | TRACT BROWSE REVERIFY COUNT.<br>This parameter specifies the number of Trace Browse records over the maximum before the severe warning messages are reissued.<br>**Minimum Value: 1000 Maximum Value: 1000000** | **1000000** | Yes | No |
| TRACEBROWSEMAXLIMIT | MAX TRACE BROWSE RECORD COUNT.<br>This parameter specifies the maximum number of Trace Browse records allowed before the severe warning message is issued. During production initialization, this limit is checked against the current Trace Browse record count. If this number is exceeded, the Trace Browse log is cleared.<br>**Minimum Value: 2000 Maximum Value: 2000000000** | **2000000000** | Yes | No |

| Parameter Name | Parameter Description | Default Value | Modifiable After Initialization | Output Only |
|---|---|---|---|---|
| USECANCELTHREAD | USE THE DB2 CANCEL THREAD COMMAND.<br>(YES, NO)<br>This parameter controls if the DB2 CANCEL THREAD command should be used to terminate SQL operations that have exceeded installation limits. If this parameter is set to YES, then the CANCEL THREAD command is used. If this parameter is set to NO, then the TCB is terminated using CALLRTM. Note that the USERABENDKILL parameter determines the type of abend created using CALLLRTM.<br>The purpose of this parameter is to avoid possible IRLM outages caused by DB2 threads being killed with an abend.<br>Note that this parameter can only be used with releases of DB2 that support the CANCEL THREAD command (DB2 4.1 and later). | **NO** | Yes | No |
| USERABENDKILL | KILL THREADS WITH USER ABEND.<br>(YES, NO)<br>The USERABENDKILL parameter controls how connections and thus tasks or threads are terminated. When this parameter is set to yes, CALLRTM is invoked using a user abend code and the RETRY=NO option. The purpose of this parameter is to avoid possible IRLM outages due to DB2 threads killed with X22 system abend codes. The use of this parameter should coincide with the setting of the following SLIP traps.<br>SLIP SET,C=U0222,ID=U222,A=NODUMP,END<br>SLIP SET,C=U0322,ID=U322,A=NODUMP,END<br>SLIP SET,C=U0522,ID=U522,A=NODUMP,END | **YES** | Yes | No |
| WAITINTERVAL | WAIT TIME SLICE INTERVAL<br>**Minimum Value: 0 Maximum Value: 100000** | **0 MILLISECONDS** | Yes | No |
| WARNINGCPUTIME | WARNING CPU TIME VALUE.<br>The warning CPU time value determines the warning limit (in seconds) of the external CPU time limit mechanism. | **0 SECONDS** | Yes | No |

| Parameter Name | Parameter Description | Default Value | Modifiable After Initialization | Output Only |
|---|---|---|---|---|
| WARNINGWAITTIME | WARNING WAIT TIME VALUE.<br>The warning wait time value determines the warning limit (in seconds) of the external wait time limit mechanism. | **0 SECONDS** | Yes | No |

# PRODREXX

| Parameter Name | Parameter Description | Default Value | Modifiable After Initialization | Output Only |
|---|---|---|---|---|
| REXXDEFAULTADDRESS | DEFAULT HOST COMMAND ENVIRONMENT FOR REXX PGMS. | **'TSO'** | Yes | No |
| REXXMAXCLAUSES | MAXIMUM NUMBER OF REXX CLAUSES.<br>**Minimum Value:** -1 **Maximum Value:** None | **1000000** | Yes | No |
| REXXMAXCOMMANDS | MAXIMUM NUMBER OF HOST COMMANDS.<br>**Minimum Value:** -1 **Maximum Value:** None | **100000** | Yes | No |
| REXXMAXPGMSIZE | MAXIMUM REXX PROGRAM SIZE IN BYTES.<br>**Minimum Value:** 32768 **Maximum Value:** None | **1048616** | Yes | No |
| REXXMAXQUEUE | MAXIMUM EXTERNAL DATA QUEUE SIZE.<br>**Minimum Value:** 1 **Maximum Value:** 8192 | **3000** | Yes | No |
| REXXMAXSAYS | MAXIMUM NUMBER OF SAY STATEMENTS.<br>**Minimum Value:** -1 **Maximum Value:** None | **100000** | Yes | No |
| REXXMAXSECONDS | MAXIMUM SECONDS OF EXECUTION TIME.<br>**Minimum Value:** -1 **Maximum Value:** 1000000000 | **-1** | Yes | No |
| REXXMAXSTRINGLENGTH | MAXIMUM LENGTH OF ANY STRING IN A REXX PROGRAM.<br>**Minimum Value:** 128 **Maximum Value:** 32000 | **32000** | Yes | No |

# PRODRPC (PRODRPPA)

| Parameter Name | Parameter Description | Default Value | Modifiable After Initialization | Output Only |
|---|---|---|---|---|
| CALLMAXROWS | MAXIMUM NUMBER OF ROWS FROM A CALL RPC.<br>This parameter is the maximum number of rows a CALL RPC can generate. If a CALL RPC tries to generate more rows than this value, it will receive an error. If this value is set to zero, then there is no limit on the number of rows a CALL RPC can generate.<br>**Minimum Value: 0 Maximum Value: 1000000000** | **10,000 ROWS** | Yes | No |
| CALLROWSSIZE | INITIAL ROW AREA SIZE FOR A CALL RPC.<br>**Minimum Value: 0 Maximum Value: 1000000000** | **20,000 BYTES** | Yes | No |
| CHECKRPCAUTHORITY | CHECK RPC EXECUTION AUTHORITY.<br>(YES, NO)<br>This parameter controls if the SEF and ACF2/RACF should be used to check if each user has the authority to execute each RPC. If this flag is set to YES, then the SEF and ACF2/RACF will be used to verify RPC execution authority. If this flag is set to NO, then all users will be allowed to execute all RPCs. Of course, the RPC can always provide its own security. | **NO** | Yes | No |
| DEFAULTRPCPARM | DEFAULT RPC PARAMETER STRING.<br>The DEFAULTRPCPARM parameter is used to set the default parameter string passed to RPC programs. This field is only used if no parameter is specified using the Shadow Event Facility (SEF) and if this parameter is set to a non-blank value. This parameter can be used to pass runtime options to language environments such as NOSTAE and NOSPIE. | **'/NOSTAE, NOSOPIE' (SDB)**<br>**NULL (SWS)** | Yes | No |
| FAILENQHOLDTIME | FAIL ENQUEUE HOLD TIME VALUE. | **0 SECONDS** | Yes | No |
| LE370LIBKEEP | ENABLE LIBKEEP FOR LE/370.<br>(YES, NO) | **YES (SD)**<br>**NO (SWS)** | Yes | No |
| ODBCCALLRPCS | CLIENTS CAN USE ODBC CALL RPCs. | **YES** | Yes | No |
| PARAMPLIST | PASS PARAMETERS USING AN OS PLIST. | **NO** | Yes | No |

| Parameter Name | Parameter Description | Default Value | Modifiable After Initialization | Output Only |
|---|---|---|---|---|
| PBFU | ADD 1 NULL BYTE TO COLUMN FOR POWERBUILDER.<br>(YES, NO)<br>The PBFU parameter when set will cause one additional byte to be added to the precision of the column. This byte will serve as a NULL termination indicator for PowerBuilder clients. | **YES (SD)**<br>**NO (SWS)** | Yes | No |
| PRELOAD | PRELOAD REENTRANT RPC MODULES.<br>(YES, NO) | **YES** | No | No |
| PREPARECALLRPCS | CLIENTS CAN PREPARE ODBC CALL RPCS.<br>(YES, NO)<br>This parameter controls if an CALL SQL statement can be prepared or not. If this parameter is set to YES, then ODBC client applications will be allowed to prepare CALL SQL statements. Note that the CALL SQL statement will actually be executed at prepare time so that result set information can be made available after the prepare is completed. Even if this parameter is set to YES, CALL SQL statements with parameter markers can not be prepared. If this parameter is set to NO, then CALL SQL statements can not be prepared. | **YES** | Yes | No |
| ROLLBACKRPCABEND | EXECUTE ROLLBACK AFTER RPC ABEND.<br>(YES, NO)<br>This parameter specifies whether a COMMIT or a ROLLBACK should be executed after an RPC abends. If this parameter is set, then a ROLLBACK will be executed after each RPC abend. If this parameter is not set, then a COMMIT will be executed. | **NO** | Yes | No |
| RPC01SPECIALREQ | RPC01 WITH SPECIAL REQUIREMENTS.<br>(YES, NO) | **NO** | Yes | No |
| RPC02SPECIALREQ | RPC02 WITH SPECIAL REQUIREMENTS.<br>(YES, NO) | **NO** | Yes | No |
| RPC03SPECIALREQ | RPC03 WITH SPECIAL REQUIREMENTS.<br>(YES, NO) | **NO** | Yes | No |

| Parameter Name | Parameter Description | Default Value | Modifiable After Initialization | Output Only |
|---|---|---|---|---|
| RPC04SPECIALREQ | RPC04 WITH SPECIAL REQUIREMENTS. (YES, NO) | **NO** | Yes | No |
| RPC05SPECIALREQ | RPC05 WITH SPECIAL REQUIREMENTS. (YES, NO) | **NO** | Yes | No |
| RPC06SPECIALREQ | RPC06 WITH SPECIAL REQUIREMENTS. (YES, NO) | **NO** | Yes | No |
| RPC07SPECIALREQ | RPC07 WITH SPECIAL REQUIREMENTS. (YES, NO) | **NO** | Yes | No |
| RPC08SPECIALREQ | RPC08 WITH SPECIAL REQUIREMENTS. (YES, NO) | **NO** | Yes | No |
| RPC09SPECIALREQ | RPC09 WITH SPECIAL REQUIREMENTS. (YES, NO) | **NO** | Yes | No |
| RPC10SPECIALREQ | RPC10 WITH SPECIAL REQUIREMENTS. (YES, NO) | **NO** | Yes | No |
| RPC11SPECIALREQ | RPC11 WITH SPECIAL REQUIREMENTS. (YES, NO) | **NO** | Yes | No |
| RPC12SPECIALREQ | RPC12 WITH SPECIAL REQUIREMENTS. (YES, NO) | **NO** | Yes | No |
| RPC13SPECIALREQ | RPC13 WITH SPECIAL REQUIREMENTS. (YES, NO) | **NO** | Yes | No |
| RPC14SPECIALREQ | RPC14 WITH SPECIAL REQUIREMENTS. (YES, NO) | **NO** | Yes | No |
| RPC15SPECIALREQ | RPC15 WITH SPECIAL REQUIREMENTS. (YES, NO) | **NO** | Yes | No |
| RPC16SPECIALREQ | RPC16 WITH SPECIAL REQUIREMENTS. (YES, NO) | **NO** | Yes | No |

| Parameter Name | Parameter Description | Default Value | Modifiable After Initialization | Output Only |
|---|---|---|---|---|
| RPC17SPECIALREQ | RPC17 WITH SPECIAL REQUIREMENTS. (YES, NO) | **NO** | Yes | No |
| RPC18SPECIALREQ | RPC18 WITH SPECIAL REQUIREMENTS. (YES, NO) | **NO** | Yes | No |
| RPC19SPECIALREQ | RPC19 WITH SPECIAL REQUIREMENTS. (YES, NO) | **NO** | Yes | No |
| RPC20SPECIALREQ | RPC20 WITH SPECIAL REQUIREMENTS. (YES, NO) | **NO** | Yes | No |
| RPC21SPECIALREQ | RPC21 WITH SPECIAL REQUIREMENTS. (YES, NO) | **NO** | Yes | No |
| RPC22SPECIALREQ | RPC22 WITH SPECIAL REQUIREMENTS. (YES, NO) | **NO** | Yes | No |
| RPC23SPECIALREQ | RPC23 WITH SPECIAL REQUIREMENTS. (YES, NO) | **NO** | Yes | No |
| RPC24SPECIALREQ | RPC24 WITH SPECIAL REQUIREMENTS. (YES, NO) | **NO** | Yes | No |
| RPC25SPECIALREQ | RPC25 WITH SPECIAL REQUIREMENTS. (YES, NO) | **NO** | Yes | No |
| RPC26SPECIALREQ | RPC26 WITH SPECIAL REQUIREMENTS. (YES, NO) | **NO** | Yes | No |
| RPC27SPECIALREQ | RPC27 WITH SPECIAL REQUIREMENTS. (YES, NO) | **NO** | Yes | No |
| RPC28SPECIALREQ | RPC28 WITH SPECIAL REQUIREMENTS. (YES, NO) | **NO** | Yes | No |
| RPC29SPECIALREQ | RPC29 WITH SPECIAL REQUIREMENTS. (YES, NO) | **NO** | Yes | No |

| Parameter Name | Parameter Description | Default Value | Modifiable After Initialization | Output Only |
|---|---|---|---|---|
| RPC30SPECIALREQ | RPC30 WITH SPECIAL REQUIREMENTS. (YES, NO) | **NO** | Yes | No |
| RPC31SPECIALREQ | RPC31 WITH SPECIAL REQUIREMENTS. (YES, NO) | **NO** | Yes | No |
| RPC32SPECIALREQ | RPC32 WITH SPECIAL REQUIREMENTS. (YES, NO) | **NO** | Yes | No |
| RPC33SPECIALREQ | RPC33 WITH SPECIAL REQUIREMENTS. (YES, NO) | **NO** | Yes | No |
| RPC34SPECIALREQ | RPC34 WITH SPECIAL REQUIREMENTS. (YES, NO) | **NO** | Yes | No |
| RPC35SPECIALREQ | RPC35 WITH SPECIAL REQUIREMENTS. (YES, NO) | **NO** | Yes | No |
| RPC36SPECIALREQ | RPC36 WITH SPECIAL REQUIREMENTS. (YES, NO) | **NO** | Yes | No |
| RPC37SPECIALREQ | RPC37 WITH SPECIAL REQUIREMENTS. (YES, NO) | **NO** | Yes | No |
| RPC38SPECIALREQ | RPC38 WITH SPECIAL REQUIREMENTS. (YES, NO) | **NO** | Yes | No |
| RPC39SPECIALREQ | RPC39 WITH SPECIAL REQUIREMENTS. (YES, NO) | **NO** | Yes | No |
| RPC40SPECIALREQ | RPC40 WITH SPECIAL REQUIREMENTS. (YES, NO) | **NO** | Yes | No |
| RPC41SPECIALREQ | RPC41 WITH SPECIAL REQUIREMENTS. (YES, NO) | **NO** | Yes | No |
| RPC42SPECIALREQ | RPC42 WITH SPECIAL REQUIREMENTS. (YES, NO) | **NO** | Yes | No |

| Parameter Name | Parameter Description | Default Value | Modifiable After Initialization | Output Only |
|---|---|---|---|---|
| RPC43SPECIALREQ | RPC43 WITH SPECIAL REQUIREMENTS.<br>(YES, NO) | **NO** | Yes | No |
| RPC44SPECIALREQ | RPC44 WITH SPECIAL REQUIREMENTS.<br>(YES, NO) | **NO** | Yes | No |
| RPC45SPECIALREQ | RPC45 WITH SPECIAL REQUIREMENTS.<br>(YES, NO) | **NO** | Yes | No |
| RPC46SPECIALREQ | RPC46 WITH SPECIAL REQUIREMENTS.<br>(YES, NO) | **NO** | Yes | No |
| RPC47SPECIALREQ | RPC47 WITH SPECIAL REQUIREMENTS.<br>(YES, NO) | **NO** | Yes | No |
| RPC48SPECIALREQ | RPC48 WITH SPECIAL REQUIREMENTS.<br>(YES, NO) | **NO** | Yes | No |
| RPC49SPECIALREQ | RPC49 WITH SPECIAL REQUIREMENTS.<br>(YES, NO) | **NO** | Yes | No |
| RPC50SPECIALREQ | RPC50 WITH SPECIAL REQUIREMENTS.<br>(YES, NO) | **NO** | Yes | No |
| RPCSUBPOOL | EXEC CICS GETMAIN SIMULATION SUBPOOL.<br>This parameter is used to simulate the EXEC CICS GETMAIN interface for RPCs executing in the main product address space. All storage requests from RPCs are satisfied from this subpool. The entire subpool is released at the end of RPC execution. This subpool is not used to get or free storage in any actual CICS address space.<br>**Minimum Value: 0 Maximum Value: 127** | **9** | Yes | No |
| SEVERRPCABEND | SEVER SESSION IF RPC ABENDS.<br>(YES, NO)<br>This parameter specifies whether the session should be terminated upon an RPC abend. This flag is used to avoid various high-level language runtime environment problems. | **NO** | Yes | No |

# PRODRRS

| Parameter Name | Parameter Description | Default Value | Modifiable After Initialization | Output Only |
|---|---|---|---|---|
| BYPASSRRS | BYPASS RRS INTERFACE CALLS.<br>(YES, NO)<br>This parameter will bypass the actual calls to RRS, giving a zero return code instead. | **YES** | Yes | No |
| BYPASSRRSEXITS | BYPASS RRS EXITS.<br>This parameter specifies whether or not to bypass setting RRS exits. | **NO** | No | No |
| PRODRRSGROUP | PRODRRS PARAMETER GROUP.<br>(YES, NO)<br>This parameter enables or disables the display of the PRODRRS parameter group. | **VISIBLE** | Yes | No |
| RECTABLEENTRIES | RECOVERY TABLE ENTRIES.<br>This parameter specifies the maximum number of entries in the RRS recovery table. Entries are placed in the RRS recovery table when two-phase commit transactions are in doubt due to error conditions that develop during processing of the transaction. The default value is 400 entries and the minimum number of entries that will be accepted is 200. If the maximum size of the table is exceeded, information on in-doubt transactions will be lost. | **400** | No | No |

| Parameter Name | Parameter Description | Default Value | Modifiable After Initialization | Output Only |
|---|---|---|---|---|
| RESOURCEMGRNAME | RRESOURCE MANAGER NAME.<br>This parameter specifies the sysplex unique name of the RRS Resource Manager (which is an SDSRM). See the IBM Programming: Resource Recovery manual (GC28-1739) for vaild naming conventions.<br>NOTE: If the name is changed, any incomplete (in-doubt) transactions from the previous run will not be able to be completed. | If not specified, a 32-character name will be created as follows:<br>Chars 1-24: NEONRRS.RESOURCE. MANAGER<br>Chars 25-28: The Shadow Subsystem name such as SDBA, SDBB, etc.<br>Chars 29-32: System SMF ID | No | No |
| RRS | INITIALIZE RRS SUPPORT.<br>(YES, NO)<br>This parameter activates RRS support. This parameter must be set to YES to activate RRS. | **NO** | No | No |
| RRSCONTEXT | RRS CONTEXT MODE.<br>This parameter specifies whether RRS XA processing will use either a NATIVE context or a PRIVATE context. | **NATIVE** | Yes | No |

# PRODSECURITY (PRODSEC)

| Parameter Name | Parameter Description | Default Value | Modifiable After Initialization | Output Only |
|---|---|---|---|---|
| ACF2SAFCALL | ACF2 ENVIRONMENT SUPPORTS SAF CALLS.<br>(YES, NO)<br>This parameter allows the customer to control when and if they will use SAF support for ACF2. Note that this parameter only applies to resource rules, logon processing uses SAF in ACF2 releases which support it. This parameter does not apply to the Shadow OS/390 Web Server. | **NO** | Yes | No |
| ALLOCSECURITYHIGH | SECURITY BLOCKS CAN BE ALLOCATED > 16MB.<br>(YES, NO)<br>This parameter shows if ACF2 and RACF (SAF) control blocks will be allocated above or below the 16 MB line. This parameter is not read only. The value can be set. However, it is normally based on the release of the security subsystem. | **YES** | Yes | No |
| CENSORAPIDATAVALUES | CENSOR VARIOUS API DATA VALUES.<br>(YES, NO)<br>This option indicates if display of various API data should be restricted to authorized users. If off display of the data in un-restricted. | **NO** | Yes | No |
| CENSORHTTPRESP | CENSOR HTTP RESPONSE OUTPUT.<br>(YES, NO)<br>This option indicates if display of out-bound response data should be restricted to authorized users. If off display of the data in un-restricted. | **NO** | Yes | No |
| CENSORURLAUTHDATA | CENSOR AUTHORIZATION HTTP HEADER DATA.<br>(YES, NO)<br>This option indicates if display of in-bound authorization data should be restricted to authorized users. If off display of the data in un-restricted. | **YES** | Yes | No |

| Parameter Name | Parameter Description | Default Value | Modifiable After Initialization | Output Only |
|---|---|---|---|---|
| CENSORURLQUERYDATA | CENSOR URL QUERY DATA.<br>(YES, NO)<br>This option indicates if display of in-bound URL query data should be restricted to authorized users. If off display of the data in un-restricted. | **NO** | Yes | No |
| CLIENTLOGON | CLIENTS CAN BE AUTHENTICATED BY NOS.<br>(YES, NO) | **YES (SD)**<br>**NO (SWS)** | Yes | No |
| EXPOSEWWWPASSWORD | EXPOSE CLEAR-TEXT PASSWORD IN WWW.PASSWORD.<br>(YES, NO)<br>This parameter controls if client passwords provided by the HTTP request Authorization: header are instantiated in CLEAR TEXT FORM as the runtime variable WWW.PASSWORD. NO is the default and recommended setting, since otherwise ANY web transaction program will have access to client passwords. Note that WWW.PASSWORD is ONLY built across the password sent via browser userid/password prompting and is not set for any other passwords processed by the system. | **NO** | Yes | No |
| FORCESECURITYLOW | FORCE ACEE ALLOCATIONS < 16 MB.<br>(YES, NO)<br>When set on, this parameter causes ACEE blocks to be unconditionally allocated below the 16 MB line. This parameter overrides any other setting. In order to acquire ACEE blocks above the line, ALLOCSECURITYHIGH must be YES and this parameter must be set to NO. The ALLOCSECURITYHIGH parameter is normally set to the correct value based on the release level of the security subsystem being used, and therefore represents the eligibility of above-the-line ACEE blocks. However, above-the-line ACEE blocks can produce intermittent and unpredictable S0C4 ABENDS within MVS dataset OPEN and CLOSE processing. If you are using only DB2 services, you may wish to allocate ACEE blocks above the line; But if you run user-written programs which use MVS QSAM, BPAM, BSAM or VSAM datasets, you should probably set this parameter to YES. | **NO (SD)**<br>**YES (SWS)** | Yes | No |

| Parameter Name | Parameter Description | Default Value | Modifiable After Initialization | Output Only |
|---|---|---|---|---|
| GETLOGONMESSAGES | GET ALL SAF LOGON MESSAGES.<br>(YES, NO)<br>This parameter controls if all of the messages from SAF LOGON processing should be obtained or not. If this parameter is set to YES, then all of the messages will be obtained. Note, that setting this parameter to YES will force the security control blocks to be located below the 16 MB line. If this parameter is set to NO, then only a subset of the SAF LOGON messages will be obtained from the SAF interface, however it will be possible to locate the security control blocks above the 16 MB line. | **NO** | Yes | No |
| HEXIPSOURCE | USE HEXADECIMAL IP ADDRESS AS SOURCE.<br>(YES, NO)<br>This parameter is used to indicate that the 'SOURCE' for SAF calls should be set to the hexadecimal form of the IP address for clients connected using TCP/IP. This flag only applies to TCP/IP connections, of course. The four byte binary IP address is converted to an eight byte upper case hexadecimal string. This string is used as the 'SOURCE' for SAF calls. The 'SOURCE' is where the SAF request is presumed to have come from. This used to mean terminal name and now has other meanings as well. | **NO** | Yes | No |
| HFSAUTHMODE | HFS AUTHORIZATION OPERATING MODE.<br>This parameter determines how security authorization processing is to be performed when serving HFS-resident files. The default value is GLOBAL which specifies that ALL accessses to any HFS-resident file or directory paths are made using the authorizations granted to the Server's default runtime userid (i.e., the userid specified by the WWWDEFAULTRUNAUTH parameter). The Server switches to this userid before any access to any HFS resident file is made and restores the pre-existing security environment after each access.<br>**NOTE:** For Shadow Web Server Version 4.5.1, GLOBAL is the only supported authorization mode. Future versions of the Server may allow additional authorization processing options. | **GLOBAL** | No | No |

| Parameter Name | Parameter Description | Default Value | Modifiable After Initialization | Output Only |
|---|---|---|---|---|
| IMPLEMENTTLS | IMPLEMENT TRANSACTION SECURITY LEVEL.<br>(YES, NO)<br>This parameter controls whether or not the Transaction Security Level (TLS) is implemented. If this parameter is set to YES, then TLS will be used. If this paramter is set to NO, TLS will not be used. TLS establishes a security environment for RPC and SQL operations based on the current generic userid. In other words, the current generic userid is used to validate operations rather than the userid used to establish the connection to the host. No password is used to validate generic userids and this feature can only be used in a highly controlled environment. The generic userids will always be cached if this option is used. In other words, the SHARERUNAUTHACEES product parameter will be set to YES. | **NO** | No | No |
| PASSEMPTYGROUPNAME | PASS EMPTY GROUP NAME TO RACROUTE.<br>(YES, NO)<br>This parameter specifies if a SAF-based RACROUTE REQUEST=VERIFY call should pass a NULL group name on the request. Passing a NULL group name allows a user-written SAF exit routine, such as ICHRTX00, to manipulate the group name, even though Shadow does not furnish or otherwise process RACF-type group names. | **NO** | Yes | No |
| PASSIMSGROUPNAME | PASS SAF GROUP NAME TO IMS.<br>(YES, NO)<br>This paramter specifies whether or not to pass sthe SAF group name to IMS. Passing the SAF group name in the PROFILE parameter allows the group name, associated with the USERID, to appear in the I/O PCB of the IMS transaction. | **NO** | Yes | No |

| Parameter Name | Parameter Description | Default Value | Modifiable After Initialization | Output Only |
|---|---|---|---|---|
| PROVIDEPASSWORDS | PROVIDE PASSWORDS FOR LOGON RULES.<br>(YES, NO)<br>This parameter controls if passwords will be provided to LOGON rules. If this parameter is set to YES, then passwords will be provided to LOGON rules. If this value is set to NO, then passwords will not be provided to LOGON rules. If this parameter is set to CHANGE, then passwords can be changed in LOGON ATH rules. Changing a password in a LOGON ATH rule does not change the password in the security product. It only changes the password used for the current connection to the host. This parameter can not be changed after product initialization for security reasons. Note that passwords are provided as plaintext strings or they are set to blanks. | **NO** | No | No |
| RACFGROUPLIST | CHECK RACF GROUP LIST FLAG.<br>(YES, NO) | **NO** | Yes | No |
| RESOURCETYPE | RESOURCE TYPE FOR RESOURCE RULES. | **'NON'** | Yes | No |
| REUSECLIENTACEES | CACHE/REUSE CLIENT ACEE BLOCKS.<br>(YES, NO)<br>If this option is set, once a client logon has occurred, the Userid and ACEE are cached by the system at the subtask level. Up to seven userids/ACEEs may be cached per www transaction subtask. Caching of client userids is only effective when the REUSETHREADS option is also set. Use of this option may significantly decrease the overhead of WWW transaction subtasks when frequent DB2 connections are made or persistent session support is enabled. | **NO** | Yes | No |

| Parameter Name | Parameter Description | Default Value | Modifiable After Initialization | Output Only |
|---|---|---|---|---|
| RULESETSEFPROXY | RULESET SEFPROXY() OVERRIDE.<br>This parameter indicates whether the SEFPROXY() settings for individual rulesets are to to be honored, or overridden on a global basis. If NOOVERRIDE is set, then each individual ruleset's SEFPROXY() setting is honored.<br>If NONE, READ, UPDATE, or CONTROL is set, then all ruleset-level SEFPROXY settings are ignored and this setting is used instead. The ruleset SEFPROXY() setting determines whether SEF directly checks each command request to see if the end user has MVS authoriztion to the underlying ruleset before performing an operation on behalf of the user. Examples of such operations are enabling a rule, setting a rule's auto-enable flag, or putting a ruleset into offline status.<br>Note that this checking is in addition to checking the the end user's authorization to use SEF facilitities. The SEF facility check is always performed using the "SEF" resource in the Server's resource class list.<br>SEFPROXY(NONE) specifies that SEF always checks the end user's authorization for every operation.<br>SEFPROXY(READ) specifies that SEF does not check the end user's authorization when performing a read-only operation such a displaying a ruleset member list or status of an individual rule.<br>SEFPROXY(UPDATE) speficies that SEF does not check authorization for read-only and single-member-update operations, such as enabled a rule or setting a rule's auto-enable flag.<br>SEFPROXY(CONTROL) specifies that SEF does not check the end user's authorization before performing s mass member update or changing the status of an entire ruleset.<br>Note that MVS will ALWAYS perform an authorization check if an end-user attempts to browse, edit or delete a ruleset | **NOOVERRIDE** | Yes | Yes |

| Parameter Name | Parameter Description | Default Value | Modifiable After Initialization | Output Only |
|---|---|---|---|---|
| SAFVERSION | SAF PARAMETER LIST VERSION.<br>This parameter controls the version of the SAF parameter list passed to the SAF interface. Some operands such as POE (Port-Of-Entry) can only be used with later versions of the SAF parameter list. | **1.8** | Yes | No |
| SECURITYMSGSUPP | SUPPRESS MESSAGES FROM RESOURCE CHECKS.<br>(YES, NO)<br>If this parameter is set to YES, the product issues RACF security resource check requests with MSGSUPP=YES specified. If resource validation fails, a TSO user is not notified of the authorization failure. | **NO** | Yes | No |
| SECURITYPACKAGE | SECURITY PRODUCT. | **RACF** | N/A | Yes |
| SECURITYVERSION | SECURITY PRODUCT VERSION. | **‘2.3’** | No | Yes |
| SHARERUNAUTHACEES | SHARE/CACHE RUNAUTH ACEE BLOCKS.<br>(YES, NO)<br>If this option is set, all explicitly specified RUNAUTH userids and ACEE control blocks are cached and globally shared by all WWW transaction subtasks. Sharing of RUNAUTH userid control blocks in this way may significantly reduce the CPU overhead associated with the use of third-party-proxy userid processing. This option operates independently of thread re-use and Client Userid/ACEE caching operations. | **NO** | No | No |
| SSL | SSL CONNECTIONS SUPPORTED.<br>(YES, NO)<br>If the SSL parameter is set to YES, then SSL connections to the server will be supported. If not enabled SSL sessions are not supported. SSL connections require that the MVS LE/370 run-time modules be present in the LINKLIST or STEPLIB libraries, and that the SSL support modules, distributed separately, be within the STEPLIB library. | **NO** | No | No |

| Parameter Name | Parameter Description | Default Value | Modifiable After Initialization | Output Only |
|---|---|---|---|---|
| SSLAUTODETECT | AUTO-DETECT SSL CONNECTIONS.<br>(YES, NO)<br>If this option is set, the server will auto-detect SSL connections which are sent on the port normally used for clear-text connections. If this option is off, only clear-text connections can be handled on the clear-text port. Note that a separately configured SSL port will ONLY accept SSL connections. | **YES** | No | No |
| SSLENCLAVETERMINATE | TERMINATE LE ENCLAVE AT SSL CLOSE.<br>(YES, NO)<br>If this option is set, the server will terminate the transaction subtask's LE/370 enclave after any SSL connection is closed. This option is for system engineering only and should only be used when directed by technical support. | **NO** | No | No |
| SSLINITIALIZED | SSL SUPPORT HAS BEEN INITIALIZED.<br>(YES, NO)<br>This parameter is only used to show if SSL initialization was successfully completed. If this parameter is set to YES, then SSL support is ready for use. If the parameter is set to NO, then SSL can not be used. | **NO** | No | Yes |
| SSLINSTALLTYPE | SSL INSTALLED SUPPORT.<br>This parameter is a read-only value, set during product initialization. It provides the type of SSL which is installed/ supported in the system. | **NONE** | No | Yes |

| Parameter Name | Parameter Description | Default Value | Modifiable After Initialization | Output Only |
|---|---|---|---|---|
| SSLUSERID | SSL RESOURCE MANAGER TASK USERID.<br>This parameter specifies a highly-privileged userid under which the SSL resource manager subtask operates. If not specified, the SSL resource manager operates using the subsystem's address-space-level Userid. This userid must be authorized to open and read the SSL Private Key and Certificate files. Use of a separate Userid for this task prevents other transaction subtasks, and the server, itself, from accessing this HIGHLY CONFIDENTIAL information. NEON Systems STRONGLY RECOMMENDS THAT THE PRIVATE KEY AND CERTIFICATE FILES BE DEFINED TO THE SECURITY SUBSYSTEM AS HIGHLY RESTRICTED, WITH FULL AUDITING. THE SSLUSERID SHOULD BE AUTHORIZED FOR READ-ONLY ACCESS TO THESE FILES. | **NULL** | No | No |
| SSSCLIENTEXPD | SSS CLIENT EXPIRATION PERIOD.<br>This parameter specifies how many days a client is to be remembered, without being successfully contacted. The known client list will be discarded at server termination.<br>**Minimum Value: 0 Maximum Value: 999** | **0** | Yes | No |
| SSSCLIENTSTEAL | STEAL SSS CLIENT LIST ENTRIES.<br>(YES, NO)<br>This parameter controls how the known client list is managed when it is full, and a new client contacts the server. A value of YES specifies that the list entry should be stolen from the client that has been the longest without server contact. NO species that the no entry is to be stolen, and the new client will not be remembered, and it will not be notified of any password changes initiated on MVS. The maximum number of clients that can be remembered is controlled by the SSSMAXCLIENTS parameter. | **NO** | Yes | No |

| Parameter Name | Parameter Description | Default Value | Modifiable After Initialization | Output Only |
|---|---|---|---|---|
| SSSDELETEAFTER | DELETE SSS EVENTS AFTER.<br>This parameter controls how mainframe generated password change events are to be deleted from the event cell pool. A value of ANY specifies that an event can be deleted when it's associated message has been sent to any known client. ALL means that an event is deleted when all known clients have been sent the message, and LAST deletes the message when the last contacted client has been sent the message. | **LAST** | Yes | No |
| SSSEVENTSTEAL | STEAL SSS EVENT LIST ENTRIES.<br>(YES, NO)<br>This parameter controls how the password change event cell pool is to be managed when all cells are in use, and a new password change event occurs. YES specifies that the oldest event cell is to be stolen. NO specifies that no cells are to be stolen, and the new event will be discarded. NO is the default. | **NO** | Yes | No |
| SSSEXITACTIVE | ACTIVATE SSS EXITS AT STARTUP.<br>(YES, NO)<br>This parameter controls initialization of the Shadow Security Server security exits. YES specifies that the exits are to be activated by SSS initialization. NO specifies that the exits are to be installed but not activated. | **YES** | No | No |
| SSSEXITREUSE | REUSE SSS EXITS AT STARTUP.<br>(YES, NO)<br>This parameter controls initialization of the Shadow Security Server security exits. YES specifies that any exit found from a previous server execution, should be reused (.e. a warm start). NO specifies that the new copies of the exits are to be initialized. | **NO** | No | No |

| Parameter Name | Parameter Description | Default Value | Modifiable After Initialization | Output Only |
|---|---|---|---|---|
| SSSMAXCLIENTS | MAXIMUM SSS CLIENT LIST ENTRIES.<br>This parameter specifies the size of the client list. The client list is used to remember known clients across invocations of the Shadow Security Server. Password change events are sent to all clients in the known client list. Entries are kept in the list until they are stolen or their expiration period is exceeded.<br>**Minimum Value: 1 Maximum Value: 64** | **16** | No | No |
| SSSMAXEVENTS | MAXIMUM SSS PASSWORD EVENT CELLS.<br>This parameter specifies the size of the password change event cell pool. The cell pool is used by the Shadow Security Server exits, to store information about successful password change operations.<br>**Minimum Value: 10 Maximum Value: 1000** | **100** | No | No |
| SSSMAXMSGS | MAXIMUM SSS MESSAGE CELLS.<br>This parameter specifies the size of the Security Server message cell pool. The cell pool is used to contain message that are to be sent to Shadow Security Server remote.<br>**Minimum Value: 100 Maximum Value: 10000** | **100** | No | No |
| SSSMSGEXPD | SSS MESSAGE EXPIRATION PERIOD.<br>This parameter specifies how many days a password change message is to be kept without successful transmission to client(s). All message cells will be discarded at server termination.<br>**Minimum Value: 0 Maximum Value: 999** | **0** | Yes | No |
| SSSMSGSTEAL | STEAL SSS MESSAGE LIST ENTRIES.<br>(YES, NO)<br>This parameter specifies whether or not cells can be stolen from the message cell poll when all cells are in use, and a new password change event occurs. YES specifies that the oldest event cell is to be stolen. NO specifies that no cells are to be stolen, and the new event will be discarded. | **NO** | Yes | No |

| Parameter Name | Parameter Description | Default Value | Modifiable After Initialization | Output Only |
|---|---|---|---|---|
| UNCENSORZOOMONLY | UNCENSOR ZOOM VIEW ONLY.<br>(YES, NO)<br>If this parameter is NO, unauthorized users view of trace messages is censored; authorized users see the view uncensored. If set to YES, both unauthorized and authorized user's view of the trace data appears censored; However, authorized users may still view the uncensored data by displaying the underlying binary information. | **YES** | Yes | No |
| URLRESOURCETYPE | RESOURCE TYPE FOR URL MATCHING. | **'NON'** | Yes | No |
| WWWDEFAULTAUTHREQ | DEFAULT WWW RULE AUTHREQ VALUE<br>(NO, AORLWWMN, YES, AORLWWMY, LOCK, AORLWWML)<br>This parameter specifies the default WWW AUTHREQ value under which web transactions run. The AUTHREQ specification can be overridden through matching to WWW rules. | **NO** | No | No |
| WWWDEFAULTRUNAUTH | DEFAULT WWW RULE RUNAUTH USERID.<br>This parameter specifies the MVS user ID under which Web transactions, by default, run. The user ID specified is made the 'effective userid' for web transactions unless WWW rules override this value. If the parameter is set to 'NONE', then the subsystem's user ID is used. | **'DEM001'** | No | No |
| WWWRUNAUTHFORMATS | RUNAUTH OPERAND FORMATS.<br>This parameter can be used to limit the allowed operand formats. If RESTRICTED is specified, RUNAUTH can not be used to specify third-party userids. | **ALL** | No | No |
| WWWRUNAUTHLOCATION | RUNAUTH ALLOWED LOCATION.<br>This parameter specifies where the RUNAUTH parameter may be coded for /*WWW rules. It may be restricted to the master WWW ruleset only, or disabled using this parameter. | **ANYWHERE** | No | No |

# PRODSEF

| Parameter Name | Parameter Description | Default Value | Modifiable After Initialization | Output Only |
|---|---|---|---|---|
| ATHINDEX | AUTHORIZATION EPROCS INDEX POINTER. | **X'00000000'** | No | Yes |
| EPROINDEX | EPROCS SET INDEX POINTER | **X'0ED995F0' (SD)**<br>**X'0AC91A60' (SWS)** | No | Yes |
| EPROSOURCETEXT | SAVE SOURCE TEXT WITH SEF EPROCS (YES, NO). | **YES** | Yes | No |
| EPROTRACE | TRACE SEF EPROCS PROCESSING (YES, NO). | **YES** | Yes | No |
| EXCINDEX | EXCEPTION EPROCS INDEX POINTER. | **X'00000000'** | No | Yes |
| GLVINDEX | GLOBAL VARIABLE EPROCS INDEX POINTER | **X'00000000'** | No | Yes |
| MSGDRAINRATE | ADDRESS SPACE MESSAGE DRAIN RATE<br>**Minimum Value: 1 Maximum Value: 32767** | **10** | Yes | No |
| MSGTHRESHOLD | ADDRESS SPACE MESSAGE THRESHOLD.<br>**Minimum Value: 10 Maximum Value: 32767** | **1000** | Yes | No |
| NOCATCHUP | SUPPRESS TOD CATCHUP PROCESSING. (YES, NO). | **YES** | No | No |
| REVERSEMATCH | REVERSE EVENT MATCH ORDER. (YES, NO). | **NO** | No | No |
| RPCINDEX | RPC EPROCS INDEX POINTER. | **X'00000000'** | No | Yes |
| SEFACTIVE | SEF PROCESSING ACTIVE. (YES, NO). | **YES** | No | No |
| SEFCMDQUEUE | ADDRESS SEF COMMAND QUEUE SIZE.<br>**Minimum Value: 1 Maximum Value: None** | **128 ACTIONS** | No | No |
| SEFDEFAULTADDRESS | DEFAULT HOST COMMAND ENVIRONMENT FOR SEF RULES. | **SEF'** | Yes | No |

| Parameter Name | Parameter Description | Default Value | Modifiable After Initialization | Output Only |
|---|---|---|---|---|
| SEFDESC | SEF MESSAGES DESCRIPTOR CODES | **X'0000'** | Yes | No |
| SEFDEST | SEF MESSAGES DESTINATION BLOCK. | **X'C200000000000000'** | Yes | No |
| SEFEXECQUEUE | SEF EXECUTE QUEUE ADDRESS. | **X'0EE2D000' (SD)**<br>**X'0A807000' (SWS)** | No | Yes |
| SEFFIRELIMIT | SEF GLOBAL EPROCS FIRING LIMIT. | **10000** | Yes | No |
| SEFGLVEVENTS | GLV EVENTS ARE SUPPORTED.<br>(YES, NO)<br>This SEF parameter determines if GLV events are supported by the system. If on, GLV events are generated and processed. Support for GLV events has a significant impact on VIRTUAL storage used by the subsystem. It is recommended that you NOT casually enable processing GLV events. | **NO** | No | No |
| SEFINITREXX | SEF INITIALIZATION REXX PROGRAM NAME. | **'SDBBINEF' (SD)**<br>**'SWSSINEF' (SWS)** | No | No |
| SEFLIMITDISABLE | DISABLE SEF EPROCS IF FIRING LIMIT EXCEEDED.<br>(YES, NO) | **NO** | Yes | No |
| SEFMAXCLAUSES | MAXIMUM NUMBER OF SEF REXX CLAUSES.<br>**Minimum Value: 1 Maximum Value: None** | **10000** | Yes | No |
| SEFMAXCOMMANDS | MAXIMUM NUMBER OF SEF HOST COMMANDS. | **400** | Yes | No |
| SEFMAXPGMSIZE | MAXIMUM SEF PROGRAM SIZE IN BYTES.<br>**Minimum Value: 32768 Maximum Value: None** | **1048616 (SD)**<br>**100000 (SWS)** | Yes | No |
| SEFMAXQUEUE | DEFAULT EXTERNAL QUEUE SIZE.<br>**Minimum Value: 1 Maximum Value: None** | **100 (SD)**<br>**300 (SWS)** | No | No |
| SEFMAXSAYS | MAXIMUM NUMBER OF SEF SAY STATEMENTS. | **1000** | Yes | No |
| SEFMAXSECONDS | MAXIMUM SECONDS OF SEF EXECUTION TIME.<br>**Minimum Value: 1 Maximum Value: None** | **10** | Yes | No |

| Parameter Name | Parameter Description | Default Value | Modifiable After Initialization | Output Only |
|---|---|---|---|---|
| SEFROUTE | SEF MESSAGES ROUTE CODES. | **X'0000'** | Yes | No |
| SEFSIZE | SEF WORK SPACE SIZE<br>**Minimum Value: 49152 Maximum Value: None** | **262144 BYTES (SD)**<br>**10485760 (SWS)** | No | No |
| SEFSUBPOOL | SEF STORAGE SUBPOOL NUMBER<br>(ZERO, OPSGZE, TWO, OPSG02, HIGHPRIVATE, OPSGPV) | **TWO** | No | No |
| SEFV3COMPATIBLE | SEF USES V3 FORMAT CONFIGURATION PARAMETERS.<br>(YES, NO)<br>If set on, this parameter specifies that SEF should use Version 3.1.1 and below compatible configuration parameters. For version 3.1.1 and below, SEF rulesets are designated by providing the dataset name prefix and suffix values and allowing SEF to locate the rulesets using a catalog search. If set off, Version 4+ configuration parameters are used. For version 4+, "DEFINE RULESET" statements must be coded in the initialization routine, and the following product parameters are ignored: EPROPREFIX, EPROSUFFIX, EPROALTFIX, AUTHEPROSET, TYPEPROSET, and WWWEPROSET.<br>**NOTE:** Existing customers that are using version 3.1 compatible configuration to define SEF rulesets must first upgrade to use Version 4+ "DEFINE RULESET" configuration statements. HFS access is NOT provided when the Server's SEFV3COMPATIBLE startup option is set to YES. | **YES** | No | No |
| SMFEPRODISABLE | SEF EPROC DISABLEMENT SMF RECORDING.<br>(YES, NO) | **NO** | Yes | No |
| SQLINDEX | SQL EPROCS INDEX POINTER. | **X'00000000'** | No | Yes |
| TODINDEX | TIME-OF-DAY EPROCS INDEX POINTER. | **X'00000000'** | No | Yes |
| TSODESC | ADDRESS TSO MESSAGES DESCRIPTOR CODES. | **X'0000'** | Yes | No |
| TSODEST | ADDRESS TSO MESSAGES DESTINATION BLOCK. | **X'0000000000000000'** | Yes | No |

| Parameter Name | Parameter Description | Default Value | Modifiable After Initialization | Output Only |
|---|---|---|---|---|
| TSOROUTE | ADDRESS TSO MESSAGES ROUTE CODES. | **X’0000’** | Yes | No |
| TYPINDEX | TYP EPROCS INDEX POINTER. | **X’00000000’ (SD)**<br>**X’0A81A028’ (SWS)** | No | Yes |
| WWWINDEX | WWW EPROCS INDEX POINTER. | **X'00000000' (SD)**<br>**X’0A81A098’ (SWS)** | No | No |

# PRODSQL (PRODSQPA)

| Parameter Name | Parameter Description | Default Value | Modifiable After Initialization | Output Only |
|---|---|---|---|---|
| ADDITIONALSQLDATA | SEND ADDITIONAL DATA WITH SQL.<br>(YES, NO)<br>This parameter is used to control whether or not additional data should be sent to the host as part of each SQL operation. The additional data is need to support per-SQL security processing. If this flag is set to YES, additional data will be sent with all SQL operations. If this flag is set to NO, only the standard data will be sent with each SQL operation. | **NO** | Yes | No |
| AUTOCOMMITCALL | AUTO COMMIT AFTER CALL.<br>(YES, NO)<br>This parameter controls if a COMMIT should be automatically executed after a NEON or IBM DB2 Stored Procedure completes execution. The COMMIT is only done if this parameter is set to YES and if AUTO-COMMIT is active for the current host connection. The COMMIT will complete any pending database changes and release some (but not all) locks. However, the COMMIT will also destroy pending result sets for IBM DB2 Stored Procedures unless the cursors for the IBM DB2 Stored Procedure result sets are declared with HOLD. | **YES** | Yes | No |
| AUTOCOMMITCC | AUTOMATIC COMMIT AT CLOSE CURSOR.<br>(YES, NO) | **YES** | Yes | No |
| AUTOSTATICCOMMIT | COMMIT AFTER DEFERRED CLOSE FOR AUTO-STATIC SQL.<br>(YES, NO) | **NO** | Yes | No |
| AUTOSTATICDEFER | DEFER CLOSE FOR AUTO-STATIC SQL.<br>(YES, NO) | **YES** | Yes | No |
| AUTOSTATICSQL | CLIENTS CAN USE AUTO-STATIC SQL.<br>(YES, NO) | **YES** | Yes | No |

| Parameter Name | Parameter Description | Default Value | Modifiable After Initialization | Output Only |
|---|---|---|---|---|
| AUTOUSERID | AUTOMATIC USERID PROPAGATION.<br>(YES, NO)<br>Use automatic userid propagation. For more information, see the appendix in the Installation Guide entitled Link-Editing the DSN2@ATH Exit. | **YES** | Yes | No |
| BLOCKFETCH | USE BLOCK FETCH.<br>(YES, NO). | **YES** | Yes | No |
| CLOSEWITHDATA | CLOSE CURSOR EVEN WITH PENDING DATA<br>(YES, NO).<br>This field controls if the cursor of a SELECT result set should be closed before all of the rows have been sent back to the client. Setting this field to YES will allow a COMMIT to be executed before all of the result set rows have been transmittedback to the ODBC client application. Of course, the COMMIT will only be executed if COMMIT after close cursor has been requested. | **NO** | Yes | No |
| CREATEGLOBAL | CREATE GLOBAL TEMPORARY TABLES.<br>(YES, NO) | **YES** | Yes | No |
| DB2ATTACHFACILITY | DB2 ATTACH FACILITY TYPE.<br>This parameter allows the user to control which mechanism to use for the db2 interface. The options are to use the classic Call Attach Facility, or CAF, using the DSNALI interface module or, the new option of Recoverable Resource Services Attach Facility, or RRSAF, can be used for DB2 v5.1 and above systems. The new facility allows the capability of a 2-phase commit through the attachment facility. Its interface routine is DSNRLI. | **CAF**<br>**RRSAF (to use RRS).** | No | No |
| DEFAULTDB2PLAN | DEFAULT DB2 PLAN NAME. | **'SDBC1010' (SD)**<br>**'SWSC1010' (SWS)** | Yes | No |

| Parameter Name | Parameter Description | Default Value | Modifiable After Initialization | Output Only |
|---|---|---|---|---|
| DEFAULTDB2PROCNAME | DEFAULT STORED PROCEDURE TABLE NAME.<br>This parameter specifies the default for the mechanism to be used for the DB2 interface. The options are to use the classic Call Attach Facility (CAF), using the DSNALI interface module or, the new option of Recoverable Resource Services Attach Facility (RRSAF) can be used for DB2 v5.1 and above systems. The new facility allows the capability of a two-phase commit through the attachment facility. Its interface routine is DSNRLI. | **'SHADOW. PROCEDURES'** | Yes | No |
| DEFAULTDB2SUBSYS | DEFAULT DB2 SUBSYSTEM NAME. | **'DB2A'** | No | No |
| DYNAMICSQL | CLIENTS CAN USE DYNAMIC SQL.<br>(YES, NO) | **YES** | Yes | No |
| ENABLEMDIAPI | ENABLE MDI API ENTRY POINTS.<br>(YES, NO)<br>This parameter controls if the MDI API should be enabled in the host address space. If this parameter is set to YES, then all of the MDI entry points will be available for use by application programs (including COBOL programs using DYNAM). If this parameter is set to NO, then the MDI API entry points will only be available to programs that link-edit the MDI interface routines statically. | **NO** | No | No |
| EXPANDEDSQLBLOCKS | SSEND LARGER SQL CONTROL BLOCKS.<br>(YES, NO)<br>This parameter is used to control whether or not larger control blocks should be sent to the host as part of each SQL operation. The additional data is needed to support new SQL related features. If this flag is set to YES, expanded control blocks will be sent for all SQL operations (assuming the client is capable of handlling larger SQL control blocks). If this flag is set to NO, only standard control blocks will be used for SQL processing. | **YES** | Yes | No |

| Parameter Name | Parameter Description | Default Value | Modifiable After Initialization | Output Only |
|---|---|---|---|---|
| GETSECONDARYLIST | EXTRACT DB2 SECONDARY USERID LIST.<br>(YES, NO)<br>This parameter controls whether or not the secondary userid list should be extracted for each DB2 thread. If this flag is set to YES, the DB2 secondary authorization id list will be obtained just after the connection to DB2 has completed. If this flag is set to NO, then no DB2 secondary userid processing will be done. The only reason to ever set this parameter to NO is when a problem is encountered extracting the DB2 secondary userid list. | **NO** | Yes | No |
| GRANTGLOBAL | GRANT ALL TO PUBLIC ON GLOBAL TABLES<br>(YES, NO) | **YES** | Yes | No |
| HOSTFUNCTIONALLEVEL | HOST FUNCTIONAL LEVEL.<br>This parameter is only used to show what level of code the host is running. This value is passed back to the client so that the client will know what host capabilities are usable. This parameter cannot be set and is intended for Technical Support use only.<br>**Minimum Value: 0 Maximum Value: 255** | **2** | Yes | No |
| IDENTIFYDSNHLI | IDENTIFY DSNHLI2 AS DSNHLI.<br>(YES, NO) | **YES** | No | No |
| IGNOREDCODE01 | IGNORED SQLCODE NUMBER 1. | **0** | Yes | No |
| IGNOREDCODE02 | IGNORED SQLCODE NUMBER 2. | **0** | Yes | No |
| IGNOREDCODE03 | IGNORED SQLCODE NUMBER 3. | **0** | Yes | No |
| IGNOREDCODE04 | IGNORED SQLCODE NUMBER 4. | **0** | Yes | No |
| IGNOREDCODE05 | IGNORED SQLCODE NUMBER 5. | **0** | Yes | No |
| IGNOREDCODE06 | IGNORED SQLCODE NUMBER 6. | **0** | Yes | No |
| IGNOREDCODE07 | IGNORED SQLCODE NUMBER 7. | **0** | Yes | No |
| IGNOREDCODE08 | IGNORED SQLCODE NUMBER 8. | **0** | Yes | No |

| Parameter Name | Parameter Description | Default Value | Modifiable After Initialization | Output Only |
|---|---|---|---|---|
| IGNOREDCODE09 | IGNORED SQLCODE NUMBER 9. | **0** | Yes | No |
| IGNOREDCODE10 | IGNORED SQLCODE NUMBER 10. | **0** | Yes | No |
| LOOKASIDESIZE | AUTO-STATIC LOOKASIDE BUFFER SIZE.<br>**Minimum Value: 0 Maximum Value: 100000** | **400** | Yes | No |
| MAXDB2ACTIVETHREADS | MAXIMUM DB2 ACTIVE THREADS. | **1 (SD)**<br>**0 (SWS)** | No | No |
| MAXROWS | MAXIMUM NUMBER OF ROWS TO FETCH.<br>This parameter controls how many rows will be fetched. If this value is zero, then there is no limit on the number of rows in a result set. If this value is non-zero, then SQLCODE +100 will be simulated as soon as the maximum number of rows is FETCHed. Note that the actual number of rows FETCHed will be the minimum of the value below and the number of rows in the result set.<br>**Minimum Value: 0 Maximum Value: 1000000000** | **0 ROWS** | Yes | No |
| MAXTIMERONS | MAXIMUM TIMERON VALUE. | **0.0 TIMERONS** | Yes | No |
| MDIERRORCODE | USE MDI ERROR CODE AS NATIVE CODE.<br>(YES, NO)<br>This parameter controls whether or not MDI error code values should be converted to ODBC native error codes. If this flag is set to YES, the MDI error code is converted to the ODBC native error code (if possible). If this flag is set to NO, the MDI error code is traced but otherwise not used. | **NO** | Yes | No |

| Parameter Name | Parameter Description | Default Value | Modifiable After Initialization | Output Only |
|---|---|---|---|---|
| MDISQLSTATE | ADD SQLSTATE TO MDI MESSAGE TEXT<br>(NO, YES)<br>This parameter controls if the SQLSTATE value from an MDI RPC should be concatenated onto the end of the message text from the MDI RPC. If this parameter is set to YES, then the SQLSTATE string will be added to the end of the message text. If this parameter is set to NO, then the SQLSTATE string will not be included in the message text from the MDI RPC. | **NO** | Yes | No |
| MDISTORAGEVALUE | MDI INITIAL GETMAIN STORAGE VALUE.<br>This parameter controls the initial value of all storage returned from the MDI EXEC CICS GETMAIN interface. This value is used to initialize all storage obtained using this mechanism. The default is to set acquired storage to binary zeros (low values). Any other character value can be used. | **X'00'** | Yes | No |
| ODBCCATALOGLEVEL | ODBC OPTIMIZED CATALOG LEVEL.<br>**Minimum Value: 0 Maximum Value: 255** | **2** | Yes | No |
| ODBCOVERHTTP | CHECK FOR ODBC CLIENTS USING HTTP.<br>(YES, NO)<br>This parameter controls if ODBC clients can use HTTP to communicate with the host. If this flag is set to YES, then all new client TCP/IP connections will be checked for HTTP headers. Otherwise, this checking will not be done and any attempt to run ODBC over HTTP will cause serious errors. Setting this flag to YES, does add a small amount of overhead to non-HTTP session initialization overhead. | **YES** | Yes | No |

| Parameter Name | Parameter Description | Default Value | Modifiable After Initialization | Output Only |
|---|---|---|---|---|
| OPTROWS | OPTIMAL NUMBER OF ROWS TO RETURN.<br>This parameter controls how many rows will be returned each time the client application asks for rows from a result set. If this value is zero, then there is no limit on the number of rows returned to the client application (other than buffer size). If this value is non-zero, then only the specified number of rows will be returned to the client application each time the client application asks for more rows. Of course, a smaller number of rows will be returned (perhaps zero) if not enough rows are available to be returned.<br>**Minimum Value: 0 Maximum Value: 30000** | **0 ROWS** | Yes | No |
| PREFETCH | PREFETCH QUEUE BLOCK COUNT<br>This parameter The PREFETCH parameter controls how many blocks of rows should be FETCHed from DB2. These blocks of rows are used to build the compressed row buffers that are sent to an ODBC application from the server. This value should only be changed if the buffers being transmitted from the server to an ODBC client application are not full.<br>**NOTE:** This parameter value should **not** be changed unless it is recommended by NEON Technical Support.<br>**Minimum Value**: 1 **Maximum Value**: 50 | **3 BLOCKS (SD)**<br>**10 BLOCKS (SWS)** | Yes | No |
| PREFETCHROWS | PREFETCH ROWS FOR BLOCK FETCH.<br>(YES, NO)<br>This parameter r controls if additional rows should be FETCHed from DB2 while a client ODBC application is processing rows FETCHed earlier. If this parameter is set to YES, then additional rows will be FETCHed from DB2 while the ODBC client is handling previous rows. If this parameter is set to NO, the FETCH processing will not be overlapped. The default value is NO.<br>**NOTE:** This parameter value should **not** be set to YES unless it is recommended by NEON Technical Support. | **NO** | Yes | No |

| Parameter Name | Parameter Description | Default Value | Modifiable After Initialization | Output Only |
|---|---|---|---|---|
| PRESENDBLOCKS | PRESEND BLOCKS TO THE CLIENT.<br>(YES, NO)<br>The PRESENDBLOCKS parameter controls if blocks of rows should be sent from the server to the ODBC client application before the ODBC client application requests the rows. If this parameter is set to YES, then blocks of rows will be pre-sent. If this parameter is set to NO, then blocks of rows will not be pre-sent. The default is NO. This parameter is not supported at this time.<br>**NOTE:** This parameter value should **not** be set to YES unless it is recommended by NEON Technical Support. | **NO** | Yes | No |
| SPECIALTABLEPREFIX | SPECIAL TABLE PREFIX.<br>This parameter is used to specify the SQL table prefix used to identify special tables. The prefix is actually the authorization ID that designates the owner of the table. If a SQL statement that refers to a table with an authorization ID equal to this value is detected, special processing is done. The special processing includes executing a stored procedure that populates the special table with data for use by the original SQL statement. | **'NEON'** | Yes | No |
| STATICSQL | CLIENTS CAN USE STATIC SQL.<br>(YES, NO) | **YES** | Yes | No |
| UPCASEMESSAGES | UPCASE MESSAGES SENT TO A CLIENT.<br>(YES, NO)<br>This parameter is used to control whether or not all messages should be converted to upper case before they are sent back to a client application. This step is required to support the Japanese language because Japanese EBCDIC has no lower case letters. If this flag is set to YES, all messages are converted to upper case. If this flag is set to NO, the messages are not converted to upper case. | **NO** | Yes | No |

# PRODSTOR

| Parameter Name | Parameter Description | Default Value | Modifiable After Initialization | Output Only |
|---|---|---|---|---|
| CSA | CSA STORAGE UTILIZATION. | **0K** | N/A | Yes |
| CSALIMIT | CSA STORAGE UTILIZATION LIMIT.<br>**Minimum Value: 1024 Maximum Value: 2097152** | **15K** | Yes | No |
| DATASIZE | SYSTEM DATA AREA DEFAULT BLOCK SIZE.<br>The system data area default block size is the amount of storage that will be acquired for a new system data area block unless a larger block is needed. A larger block will be needed if the current object will not fit into an empty system data area block. This parameter should only be set under the specific guidance of the Technical Support group.<br>**Minimum Value: 512 Maximum Value: 65536** | **1K** | Yes | No |
| ECSA | ECSA STORAGE UTILIZATION. | **195 (SD)**<br>**187 (SWS)** | N/A | Yes |
| ECSALIMIT | ECSA STORAGE UTILIZATION LIMIT.<br>**Minimum Value: 262144 Maximum Value: 16777216** | **4096K** | Yes | No |
| EMINPRIV | EPRIVATE MINIMUM STORAGE REQUIRED.<br>This parameter is used to control the minimum amount of above the 16 MB line storage that must be available for new inbound sessions to be handled. If this much storage is not available, new inbound sessions will be rejected. If this parameter is set to zero, then the amount of above the 16 MB line storage will not be checked for each new connection.<br>**Minimum Value: 0 Maximum Value: 8388608** | **4096K** | Yes | No |
| EPRIV | EPRIVATE STORAGE UTILIZATION. | **19562K (SD)**<br>**1647K (SWS)** | N/A | Yes |

| Parameter Name | Parameter Description | Default Value | Modifiable After Initialization | Output Only |
|---|---|---|---|---|
| ERRORSTACKSIZE | ERROR STACK SIZE.<br>This parameter is the amount of storage acquired for each process for error processing. This value should be raised if stack underflow errors occur. This parameter should only be set under the specific guidance of the Technical Support group. | **16K** | Yes | No |
| MINPRIV | PRIVATE MINIMUM STORAGE REQUIRED.<br>This parameter is used to control the minimum amount of below the 16 MB line storage that must be available for new inbound sessions to be handled. If this much storage is not available, new inbound sessions will be rejected. If this parameter is set to zero, then the amount of below the 16 MB line storage will not be checked for each new connection.<br>**Minimum Value: 0 Maximum Value: 2097152** | **300K** | Yes | No |
| PRIMARYSTACKHW | PRIMARY STACK HI-WATER.<br>This primary stack hi-water mark is the maximum usage of the stack for all threads. | **0K** | N/A | Yes |
| PRIMARYSTACKMAX | PRIMARY STACK MAXIMUM.<br>This primary stack maximum sets an upper limit on the primary stack size. This parameter should only be set under the specific guidance of the Technical Support group. | **196K (SD)**<br>**416K (SWS)** | Yes | No |
| PRIMARYSTACKSIZE | PRIMARY STACK SIZE.<br>This parameter is the amount of storage acquired for each process for normal processing. This value should be raised if stack underflow errors occur. This parameter should only be set under the specific guidance of the Technical Support group. | **192K (SD)**<br>**352K (SWS)** | Yes | No |
| PRIV | PRIVATE STORAGE UTILIZATION. | **44K (SD)**<br>**37K (SWS)** | N/A | Yes |

| Parameter Name | Parameter Description | Default Value | Modifiable After Initialization | Output Only |
|---|---|---|---|---|
| RESERVEEHIGH | RESERVED EXTENDED HIGH AREA SIZE.<br>This parameter is used to determine how much extended high private should be reserved during product initialization. This storage is obtained during product startup and is released at the start of product shutdown. The storage is released to assure that the termination routines will always have enough storage to properly execute.<br>**Minimum Value: 0 Maximum Value: 4194304** | **3500K** | No | No |
| RESERVEELOW | RESERVED EXTENDED LOW AREA SIZE.<br>This parameter is used to determine how much extended low private should be reserved during product initialization. This storage is obtained during product startup and is released at the start of product shutdown. The storage is released to assure that the termination routines will always have enough storage to properly execute.<br>**Minimum Value: 0 Maximum Value: 4194304** | **3000K** | No | No |
| RESERVEELSQA | RESERVED ELSQA AREA SIZE.<br>This parameter is used to determine how much ELSQA should be reserved during product initialization. This storage is obtained during product startup and is released at the start of product shutdown. The storage is released to assure that the termination routines will always have enough storage to properly execute.<br>**Minimum Value: 0 Maximum Value: 4194304** | **3500K** | No | No |
| RESERVEHIGH | RESERVED HIGH PRIVATE AREA SIZE.<br>This parameter is used to determine how much high private should be reserved during product initialization. This storage is obtained during product startup and is released at the start of product shutdown. The storage is released to assure that the termination routines will always have enough storage to properly execute.<br>**Minimum Value: 0 Maximum Value: 1048576** | **350K** | No | No |

| Parameter Name | Parameter Description | Default Value | Modifiable After Initialization | Output Only |
|---|---|---|---|---|
| RESERVELOW | RESERVED LOW PRIVATE AREA SIZE.<br>This parameter is used to determine how much low private should be reserved during product initialization. This storage is obtained during product startup and is released at the start of product shutdown. The storage is released to assure that the termination routines will always have enough storage to properly execute.<br>**Minimum Value: 0 Maximum Value: 1048576** | **300K** | No | No |
| RESERVELSQA | RESERVED LSQA AREA SIZE.<br>This parameter is used to determine how much LSQA should be reserved during product initialization. This storage is obtained during product startup and is released at the start of product shutdown. The storage is released to assure that the termination routines will always have enough storage to properly execute.<br>**Minimum Value: 0 Maximum Value: 1048576** | **350K** | No | No |
| SHARESUBPOOLZERO | SHARE SUBPOOL ZERO STORAGE.<br>(NO, YES)<br>This parameter indicates whether subpool zero is to be shared between tasks. When subpool zero is shared, applications must explicitly free any storage allocated in subpool zero since shared subpool storage is not released at end of task. If the server is accessing VSAM files between multiple tasks under the same DDNAME, this value should be set to YES, otherwise this value should be set to NO. If this value is set to yes, the server should be recycled on a daily basis to free orphaned subpool zero storage. | **YES (SD**<br>**NO (SWS)** | Yes | No |
| STACKINCREMENTAMOUNT | PRIMARY STACK INCREMENT AMOUNT.<br>This primary stack increment amount is used to increase the default primary stack size in response to short on stack storage condition(s). This parameter should only be set under the specific guidance of the Technical Support group. | **16K** | Yes | No |

# PRODTOKEN

| Parameter Name | Parameter Description | Default Value | Modifiable After Initialization | Output Only |
|---|---|---|---|---|
| CHECKTOKENSINTERVAL | TOKEN TIMEOUT CHECKING INTERVAL.<br>This parameter controls how often each token is checked to see if the token has timed out. If the token has timed out, the token and the associated data (if any) are released. The interval value is specified in seconds and should be a factor of one hour. In other words the value should divide evenly into 3600.<br>**Minimum Value: 1 Maximum Value: 3600** | **15 SECONDS** | Yes | No |
| CURRENTTOKENADDRESS | LAST ALLOCATED TOKEN ENTRY ADDRESS.<br>This read-only parameter contains the address of the last token entry allocated by the system. | **X'00000000'** | N/A | Yes |
| CURRENTTOKENBLOCK | LAST ALLOCATED TOKEN BLOCK ADDRESS.<br>This read-only parameter contains the address of the last token control block allocated for storage of new token entries. | **X'00000000'** | N/A | Yes |
| ENABLETOKENEXC | ENABLE TOKEN EXPIRATION EXC RULE.<br>(YES, NO)<br>This paramter enables token expiration processing to fire an SEF EXC rule. | **NO** | Yes | No |
| TOKENBLOCKCOUNT | NUMBER OF TOKEN BLOCKS. | **0 BLOCKS** | No | No |
| TOKENBLOCKPTR | FIRST TOKEN BLOCK ADDRESS | **X'00000000'** | No | No |
| TOKENENTRYCOUNT | NUMBER OF TOKEN ENTRIES. | **0 TOKENS** | No | No |
| TOKENSALLOCATED | NUMBER OF TOKENS ALLOCATED. | **0 TOKENS** | No | No |
| TOKENSDELETED | NUMBER OF TOKENS DELETED. | **0 TOKENS** | No | No |
| TOKENSINUSE | NUMBER OF TOKENS IN USE. | **0 TOKENS** | No | No |
| TOKENSTIMEDOUT | NUMBER OF TOKENS TIMED OUT. | **0 TOKENS** | No | No |

| Parameter Name | Parameter Description | Default Value | Modifiable After Initialization | Output Only |
|---|---|---|---|---|
| TOKENSTORAGE | TOKEN VALUE STORAGE UTILIZATION.<br>This read-only parameter shows the amount of storage currently allocated for storage of token data values. It does not include the storage allocated for the system-managed token blocks and token entries; only the size of the data values assigned to tokens is included in this total. | **0K** | N/A | Yes |
| TOKENTIMEOUT | DEFAULT TOKEN TIMEOUT VALUE.<br>**Minimum Value: 1 Maximum Value: 2000000000** | **3600 SECONDS** | Yes | No |

# PRODTRAC (PRODTRACE)

| Parameter Name | Parameter Description | Default Value | Modifiable After Initialization | Output Only |
|---|---|---|---|---|
| DEBUGSEFWAIT | DEBUG SEF INITIALIZATION WAIT.<br>(YES, NO)<br>This parameter can be set on to debug SEF initialization wait processing. This parameter should only be set under the specific guidance of the Technical Support group. | **NO** | Yes | No |
| DEBUGSGMG | DEBUG FLAG FOR SGMG ROUTINE.<br>(ON, OFF) | **OFF** | Yes | No |
| EPROTRACE | TRACE SEF EPROCS PROCESSING.<br>(YES, NO)<br>If set to YES, this parameter causes after-execution tracing to be performed for SEF event/rule processing. If set to NO, only the before-execution trace record is logged. The default value YES is recommended for Shadow Direct, and very strongly recommended for Shadow OS/390 Web Server. | **YES** | Yes | No |
| PRECISECPUTIME | OBTAIN PRECISE CPU TIME INFORMATION.<br>(YES, NO)<br>This parameter controls how CPU time information is obtained. If this parameter is set to YES, highly accurate CPU time information is obtained at a greater CPU cost. Otherwise, a less accurate (but faster) mechanism is used to obtain CPU time. The CPU time information is used to build SMF records. | **NO** | Yes | No |
| SMFFULLSQL | TRACE FULL SQL SOURCE IN SMF.<br>(YES, NO)<br>This parameter controls how much SQL source is included in SMF records. If this parameter is set to YES, then the full SQL source will always be included in each SMF record. If this parameter is set to NO, then only the first 256 bytes of the SQL source will be included in each SMF record. Note that in practice only about 32,000 bytes of SQL source can be included in an SMF record. | **NO** | Yes | No |

| Parameter Name | Parameter Description | Default Value | Modifiable After Initialization | Output Only |
|---|---|---|---|---|
| SMFNUMBER | SMF RECORD NUMBER<br>To enable SDB SMF recording, set SMFNUMBER to desired number. If set to zero, no logging takes place. (Used with Shadow Direct ).<br>**Minimum Value: 0 Maximum Value: 255** | **0** | Yes | No |
| SMFTRACEASTEXT | TRACE SMF RECORDS AS TEXT.<br>(YES, NO)<br>This parameter controls the tracing of SMF records. If this flag is set to YES, then each SMF record is copied into Trace Browse just before it is written out to SMF. If this parameter is set to NO, then SMF records are not copied into Trace Browse as text records. SMF records are only copied into Trace Browse for debugging purposes. This flag should only be set to YES to debug SMF record problems. | **NO** | Yes | No |
| SMFTRANSACT | SMF PER-TRANSACTION RECORDING.<br>(YES, NO).<br>This parameter controls the creation of SMF transaction records. If this parameter is set to YES, then an SMF record will be created for each inbound client request. If this parameter is set to NO, then no per-transaction records will be created.<br>Each SMF transaction record contains information about all of the work done on behalf of the client. The inbound client request may have caused zero, one, or more SQL operations to be executed. | **NO** | Yes | No |
| THREADLEVELTRACE | ISOLATE MODULE TRACE TO THREAD LEVEL.<br>(YES, NO).<br>When this flag is set, TRACEENTRY, TRACEEXIT and TRACEDATA isolate tracing to one or more enabled subtask threads. When off, these routines generate tracing for all exits within the entire product. | **NO** | Yes | No |

| Parameter Name | Parameter Description | Default Value | Modifiable After Initialization | Output Only |
|---|---|---|---|---|
| TRACE | PRODUCT TRACE OPTION.<br>This parameter sets that overall level of communication (LU 6.2 and/or TCP/IP) tracing for the product. Trace messages generated using this parameter are send to the MVS log not to Trace Browse. Use of this parameter is not recommended. This parameter should only be set under the specific guidance of the Technical Support group. | **NONE** | Yes | No |
| TRACE24GETS | ONLY TRACE 24-BIT GETMAIN STR EVENTS.<br>(YES, NO) | **YES** | Yes | No |
| TRACEABENDEVENTS | TRACE ABEND EVENTS.<br>(YES, NO)<br>If abends occur either in the Shadow Server address space or in an RPC, trace the abend. | **YES** | Yes | No |
| TRACEABENDRETRYINFO | TRACE ABEND RETRY INFORMATION.<br>(YES, NO)<br>This parameter controls whether or not the retry registers and other information is traced whenever an enabled retry stack frame can be located during ESTAE recovery processing. The retry information, if any, is traced along with the original abend SDWA image, when possible, even if retry is not possible and the abend is percolated. | **YES** | Yes | No |
| TRACEABENDSDWARC1 | TRACE ABEND SDWARC1 IMAGE.<br>(YES, NO)<br>This parameter controls whether the SDWARC1 control block image is traced for ABEND events. TRACEABENDEVENTS must also be on. The SDWARC1 control block contains access and control register values at the point of an abnormal termination. | **YES** | Yes | No |

| Parameter Name | Parameter Description | Default Value | Modifiable After Initialization | Output Only |
|---|---|---|---|---|
| TRACEACEECHANGES | TRACE TASK-LEVEL ACEE CHANGES.<br>(YES, NO)<br>This parameter can set can be used to turn on an internal trace of ACEE pointer alterations. ACEE pointer alterations are used by the Shadow OS/390 Web Server to reset the effective Userid under which Web transactions are run. This parameter should only be set under the specific guidance of the Technical Support group. | **NO** | Yes | No |
| TRACEAPPCDATA | TRACE FULL APPC/MVS DATA.<br>(YES, NO)<br>This parameter controls whether the full APPC/MVS data for APPC/MVS events is traced or not. If this parameter is set to YES, then the complete APPC/MVS data for APPC/MVS events will be traced using Trace Browse. If this parameter is set to NO, then the full APPC/MVS data will not be traced. | **NO** | Yes | No |
| TRACEAPPCMVSEVENTS | TRACE APPC/MVS EVENTS.<br>(YES, NO) | **YES** | Yes | No |
| TRACEAPPCMVSMN | TRACE APPC/MVS MONITOR.<br>(YES, NO)<br>This parameter controls whether the APPC/MVS Monitor data collection APIs are to be traced. This parameter should only be turned on if the monitor is not functioning correctly. | **NO** | Yes | No |
| TRACEAPPCMVSSR | TRACE APPC/MVS SEND/RECV.<br>(YES, NO) | **YES (SD)**<br>**NO (SWS)** | Yes | No |
| TRACEATTACHEVENTS | TRACE ATTACH EVENTS.<br>(YES, NO) | **YES** | Yes | No |
| TRACEAUTHEVENTS | TRACE AUTHORIZATION EVENTS.<br>(YES, NO) | **NO (SD)**<br>**YES (SWS)** | Yes | No |
| TRACEBROWSEGROUP1 | TRACE BROWSE FLAG GROUP 1. | **X’226FB07E’ (SD)**<br>**X’227FB07E’ (SWS)** | Yes | No |

| Parameter Name | Parameter Description | Default Value | Modifiable After Initialization | Output Only |
|---|---|---|---|---|
| TRACEBROWSEGROUP2 | TRACE BROWSE FLAG GROUP 2. | **X'501FB332' (SD)**<br>**X'500DA332' (SWS)** | Yes | No |
| TRACEBROWSEGROUP3 | TRACE BROWSE FLAG GROUP 3. | **X'A8004000' (SD)**<br>**X'A8884000' (SWS)** | Yes | No |
| TRACEBROWSEGROUP4 | TRACE BROWSE FLAG GROUP 4. | **X'00000000'** | Yes | No |
| TRACECEVENTS | TRACE CLIENT PROGRAM EVENTS.<br>(YES, NO)<br>Causes events associated with C-programs running in Shadow Server's address space to be traced. | **YES** | Yes | No |
| TRACECICSEVENTS | TRACE CICS EVENTS.<br>(YES, NO) | **YES** | Yes | No |
| TRACECURSOR | TRACE CURSOR STATUS.<br>(YES, NO) | **NO** | Yes | No |
| TRACECURSORADDRESS | TRACE CURSOR ADDRESS.<br>(YES, NO) | **NO** | Yes | No |
| TRACEDATA | TRACE MODULE DATA.<br>This parameter is for debugging purposes only and should be used only under the guidance of technical support. | **X'07FE'** | Yes | No |
| TRACEDETACHEVENTS | TRACE DETACH EVENTS.<br>(YES, NO) | **YES** | Yes | No |
| TRACEDISABLEEVENTS | TRACE DISABLE EVENTS.<br>(YES, NO) | **YES** | Yes | No |
| TRACEENABLEEVENTS | TRACE ENABLE EVENTS.<br>(YES, NO) | **YES** | Yes | No |
| TRACEEXCEPTIONEVENTS | TRACE EXCEPTION EVENTS.<br>(YES, NO) | **YES** | Yes | No |

| Parameter Name | Parameter Description | Default Value | Modifiable After Initialization | Output Only |
|---|---|---|---|---|
| TRACEEXCIDPLEVENTS | TRACE EXCI DPL EVENTS.<br>(YES, NO) | **NO** | Yes | No |
| TRACEEXCIEVENTS | TRACE EXCI EVENTS.<br>(YES, NO) | **YES** | Yes | No |
| TRACEFILEEVENTS | TRACE FILE EVENTS.<br>(YES, NO)<br>This option controls if file-related processing events are logged to the wrap-around trace. | **YES** | Yes | No |
| TRACEFULLDPLDATA | TRACE FULL DPL DATA.<br>(YES, NO)<br>This parameter controls whether the entire COMMAREA for DPL events is traced. If this parameter is set to YES, then the complete COMMAREA for DPL events will be traced using Trace Browse. If this parameter is set to NO, then the full COMMAREA will not be traced. | **NO** | Yes | No |
| TRACEFULLRRSDATA | TRACE FULL RRS DATA.<br>(YES, NO)<br>This parameter controls whether or not the entire RRS work area will be traced for RRS events using Trace Browse. If this parameter is set to YES, the complete RRSAREA for RRS events will be traced using Trace Browse. If this parameter is set to NO, then only the amount of data that will fit in a standard message block will be traced.. | **NO** | Yes | No |
| TRACEGLVEVENTS | TRACE GLOBAL VARIABLE EVENTS.<br>(YES, NO) | **YES** | Yes | No |
| TRACEHLLENQDEQ | TRACE PRODUCT HLL ENQ/DEQ ACTIVITY.<br>(YES, NO)<br>If this flag is set, any ENQ or DEQ operations generated by HLL PRODUCT components via the internal-use-only API module are traced. | **NO** | Yes | No |

| Parameter Name | Parameter Description | Default Value | Modifiable After Initialization | Output Only |
|---|---|---|---|---|
| TRACEHSMEVENTS | TRACE DFHSM EVENTS AS FILE EVENTS..<br>(YES, N0)<br>This parameter controls whether or not DFHSM request processing operations are traced as FILE events. The TRACEFILEEVENT option must also be set to YES for the parameter to have any effect. | **NO** | Yes | No |
| TRACEIBMMQEVENTS | TRACE IBM/MQ EVENTS.<br>(YES, NO) | **YES** | Yes | No |
| TRACEIBMMQGP | TRACE IBM/MQ MGET/MPUT EVENTS.<br>(YES, NO) | **NO** | Yes | No |
| TRACEIMSDLIEVENTS | TRACE IMS DLI EVENTS. | **NO** | Yes | No |
| TRACEIMSEVENTS | TRACE IMS EVENTS.<br>(YES, NO)<br>Causes all events related to retrieving IMS data to be traced. | **YES** | Yes | No |
| TRACEINTERVAL | TRACE INTERVAL PROCESSING.<br>(YES, NO)<br>This parameter controls the tracing of interval processing. If this flag is set to YES, then a text message is written into Trace Browse just before each type of interval processing is performed. If this parameter is set to NO, then a text message is not added to Trace Browse as part of interval processing. Note that interval processing is performed in either case. This flag should be set to YES only to debug problems with interval processing. | **NO** | Yes | No |
| TRACEITCIPAPI | API TRACING FOR ITC/IP EVENTS<br>(YES, NO). | **NO** | Yes | No |

| Parameter Name | Parameter Description | Default Value | Modifiable After Initialization | Output Only |
|---|---|---|---|---|
| TRACEITCIPDATA | TRACE FULL INTERLINK TCP/IP DATA.<br>(YES, NO)<br>This parameter controls whether the full Interlink TCP/IP data for Interlink read/write events is traced or not. If this parameter is set to YES, then the complete Interlink TCP/IP data for Interlink read/write events will be traced using Trace Browse. If this parameter is set to NO, then the full Interlink TCP/IP data will not be traced. Note, that this parameter only controls tracing for Interlink TCP/IP. | **NO** | Yes | No |
| TRACEITCIPEVENTS | TRACE ITC/IP EVENTS.<br>(YES, NO) | **YES** | Yes | No |
| TRACEITCIPGTF | GTF TRACING FOR ITC/IP EVENTS.<br>(YES, NO) | **NO** | Yes | No |
| TRACEITCIPRW | TRACE ITC/IP READ/WRITE EVENTS.<br>(YES, NO) | **NO** | Yes | No |
| TRACELU62DATA | TRACE FULL LU 6.2 DATA.<br>(YES, NO)<br>This parameter controls whether the full LU 6.2 data for LU 6.2 read/write events is traced or not. If this parameter is set to YES, then the complete LU 6.2 data for LU 6.2 read/write events will be traced using Trace Browse. If this parameter is set to NO, then the full LU 6.2 data will not be traced. | **NO** | Yes | No |
| TRACELU62DETAIL | TRACE DETAILED LU 6.2 EVENTS.<br>(YES, NO) | **NO** | Yes | No |
| TRACELU62EVENTS | TRACE LU 6.2 EVENTS.<br>(YES, NO) | **NO** | Yes | No |
| TRACELU62RDWR | TRACE LU 6.2 READ/WRITE EVENTS.<br>(YES, NO). | **NO** | Yes | No |

| Parameter Name | Parameter Description | Default Value | Modifiable After Initialization | Output Only |
|---|---|---|---|---|
| TRACEMERGE | MERGE SUCCESSFUL FETCH EVENTS.<br>(YES, NO)<br>Setting this parameter to YES will merge successful eternal fetches which belong to the same cursor and thread. | **YES** | Yes | No |
| TRACEMERGETHROW | MERGE SUCCESSFUL THROW EVENTS<br>(YES, NO) | **YES** | Yes | No |
| TRACEMESSAGEEVENTS | TRACE MESSAGE EVENTS.<br>(YES, NO)<br>Causes all message-related events types to be traced. | **YES** | Yes | No |
| TRACENOEVENTS | TRACE NO EVENT TYPE EVENTS.<br>(YES, NO).<br>This parameter enables the trace browse facility to trace events that are of an unknown event type. | **NO** | Yes | No |
| TRACENTRY | TRACE MODULE ENTRY.<br>This parameter is for debugging purposes only and should be used only under the guidance of technical support. | **X‘07FE’** | Yes | No |
| TRACEOEDATA | TRACE FULL OE SOCKETS DATA.<br>(YES, NO)<br>This parameter controls whether the full OE Sockets data for OE Sockets read/write events is traced or not. If this parameter is set to YES, then the complete OE Sockets data for OE Sockets read/write events will be traced using Trace Browse. If this parameter is set to NO, then the full OE Sockets data will not be traced. | **NO** | Yes | No |
| TRACEOEEVENTS | TRACE IBM OE SOCKETS EVENTS.<br>(YES, NO)<br>This parameter controls wheter or not IBM OE Sockets TCP/IP events should be traced. If this parameter is set to YES, IBM OE Sockets TCP/IP events will be traced. if this parameter is set to NO, then IBM OE Sockets TCP/IP events will not be traced. | **YES** | Yes | No |

| Parameter Name | Parameter Description | Default Value | Modifiable After Initialization | Output Only |
|---|---|---|---|---|
| TRACEOERW | TRACE OE SOCKETS READ/WRITE EVENTS.<br>(YES, NO)<br>This parameter controls whether or not IBM OE Sockets TCP/IP Read/Write events should be traced. If this parameter is set to YES, IBM OE Sockets TCP/IP Read/Write events will be traced. if this parameter is set to NO, then IBM OE Sockets TCP/IP Read/Write events will not be traced. | **NO** | Yes | No |
| TRACEOERWSTART | TRACE OE SOCKETS R/W EVENT START.<br>(YES, NO)<br>This parameter controls if the start of IBM OE Sockets TCP/IP Read/Write events should be traced or not. If this parameter is set to YES, the initialization of IBM OE TCP/IP Read/Write events will be traced. If set to NO, the initialization will not be traced. | **NO** | Yes | No |
| TRACEOTMAEVENTS | TRACE OTMA EVENTS.<br>(YES, NO)<br>This parameter is used to control the tracing of IMS/OTMA events. | **NO** | Yes | No |
| TRACERESPBUFFERS | TRACE HTTP RESP BUFFERING.<br>(YES, NO)<br>If this parameter is set the server generates trace entries for certain HTTP response buffering operations. The trace information is used mostly to diagnose problems when an out-bound HTTP response appears to be incomplete. | **NO** | Yes | No |
| TRACEREXXEXEC | TRACE REXX EXECUTION.<br>(YES, NO) | **NO** | Yes | No |
| TRACERPCEVENTS | TRACE RPC EVENTS.<br>(YES, NO)<br>Causes all events related to RPCs to be traced. | **YES** | Yes | No |
| TRACERPCEVENTS | TRACE ODBC CALL RPC EVENTS.<br>(YES, NO) | **YES** | Yes | No |

| Parameter Name | Parameter Description | Default Value | Modifiable After Initialization | Output Only |
|---|---|---|---|---|
| TRACERRSEVENTS | TRACE RRS EVENTS.<br>(YES, NO)<br>This parameter specifies whether or not RRS events will be traced via the Trace Browse Facility. | **YES** | Yes | No |
| TRACERRSXXXEVENTS | TRACE RRS XXX EVENTS.<br>(YES, NO) | **NO** | Yes | No |
| TRACESECURITYATTRIBS | TRACE WWW SECURITY ATTRIBUTES.<br>(YES, NO) | **YES** | Yes | No |
| TRACESECURITYDATA | TRACE SECURITY DATA.<br>(YES, NO)<br>The next flag controls tracing of security data. The only current security data is messages from Logon processing. These messages are copied into Trace Browse as text if the flag below is set. | **NO** | Yes | No |
| TRACESQLEVENTS | TRACE SQL EVENTS.<br>(YES, NO).<br>This parameter controls whether SQL events are traced or not. If this parameter is set to YES, then SQL events will be traced using Trace Browse. If this parameter is set to NO, then SQL events will not be traced. Note, that this parameter does not control the tracing of SQL events from the logging task. SQL events from the logging task are traced as SQM events. SQL events can be filtered using the Trace Browse profile facility. | **YES** | Yes | No |
| TRACESQLSOURCE | TRACE FULL SQL SOURCE.<br>(YES, NO)<br>This parameter controls whether the full SQL source for SQL events is traced or not. If this parameter is set to YES, then the complete SQL source for SQL events will be traced using Trace Browse. If this parameter is set to NO, then the full SQL source will not be traced. | **NO** | Yes | No |

| Parameter Name | Parameter Description | Default Value | Modifiable After Initialization | Output Only |
|---|---|---|---|---|
| TRACESQMEVENTS | TRACE SQL EVENTS FROM LOGGING.<br>(YES, NO)<br>This parameter controls whether SQL events from the logging task are traced or not. If this parameter is set to YES, then SQL events from the logging task will be traced using Trace Browse. Note, that the event type will be SQM, not SQL. If this parameter is set to NO, then SQL events from the logging task will not be traced. Note, that this parameter only controls the tracing of SQL events from the logging task. The tracing of all other SQL events is controlled using the TRACESQLEVENTS parameter. SQM events can be filtered using the Trace Browse profile facility. | **YES** | Yes | No |
| TRACESRPFUNCTION | TRACE SERVICE PROVIDER FUNCTIONS.<br>(YES, NO)<br>This parameter can be set on to cause the service requester/ provider interface to generate trace messages during internal operations. This parameter should only be set under the specific guidance of the Technical Support group. | **NO** | Yes | No |
| TRACESSLACCEPT | TRACE SSL ACCEPT CALLS.<br>(YES, NO)<br>If this flag is set, SSL accept calls are traced. | **YES** | Yes | No |
| TRACESSLACCEPTSTATES | TRACE SSL ACCEPT STATES.<br>(YES, NO)<br>If this flag is set, SSL acceptance processing stages are traced. | **YES** | Yes | No |
| TRACESSLCLOSE | TRACE SSL CLOSE CALLS.<br>(YES, NO)<br>If this flag is set, SSL close calls are traced. | **YES** | Yes | No |
| TRACESSLEVENTS | TRACE SSL EVENTS.<br>(YES, NO) | **YES** | Yes | No |
| TRACESSLFILEBIO | TRACE SSL FILE INTERCEPTS.<br>(YES, NO)<br>If this flag is set, SSL file operation intercepts are traced. | **NO** | Yes | No |

| Parameter Name | Parameter Description | Default Value | Modifiable After Initialization | Output Only |
|---|---|---|---|---|
| TRACESSLREAD | TRACE SSL READ CALLS.<br>(YES, NO)<br>If this flag is set, SSL read calls are traced. | **YES** | Yes | No |
| TRACESSLTCPIPBIO | TRACE SSL TCP/IP INTERCEPTS.<br>(YES, NO)<br>If this flag is set, SSL TCP/IP intercept operations are traced. | **NO** | Yes | No |
| TRACESSLVERSION | TRACE SSL CODE VERSION.<br>(YES, NO)<br>If this flag is set, SSL_ACCEPT and SSL_GET_CTX operations trace the SSLeay version string. | **NO** | Yes | No |
| TRACESSLWRITE | TRACE SSL WRITE CALLS.<br>(YES, NO)<br>If this flag is set, SSL write calls are traced. | **YES** | Yes | No |
| TRACESSSEVENTS | TRACE SSS EVENTS.<br>(YES, NO)<br>This parameter is used to control tracing of Shadow Security Server events. Specifying YES causes all events to be traced. | | Yes | No |
| TRACESTACK | TRACE STACK USAGE.<br>(YES, NO)<br>This field controls whether or not stack trace is on. | **NO** | Yes | No |
| TRACESTATICSQL | TRACE STATIC SQL SOURCE.<br>(YES, NO) | **NO** | Yes | No |
| TRACESTORAGEEVENTS | TRACE STORAGE EVENTS<br>(YES, NO)<br>Causes all trace storage getting and freeing events to be traced. | **NO** | Yes | No |
| TRACESTREVENTS | TRACE STR EVENTS FROM SYSTEM TRACE.<br>(YES, NO) | **YES** | Yes | No |

| Parameter Name | Parameter Description | Default Value | Modifiable After Initialization | Output Only |
|---|---|---|---|---|
| TRACETCPIPDATA | TRACE FULL TCP/IP DATA.<br>(YES, NO)<br>This parameter controls whether the full TCP/IP data for TCP/IP read/write events is traced or not. If this parameter is set to YES, then the complete TCP/IP data for TCP/IP read/write events will be traced using Trace Browse. If this parameter is set to NO, then the full TCP/IP data will not be traced. Note, that this parameter only controls tracing for no-OE IBM TCP/IP. | **NO** | Yes | No |
| TRACETCPIPEVENTS | TRACE TCP/IP EVENTS.<br>(YES, NO).<br>This parameter controls if IBM TCP/IP events should be traced or not. If this parameter is set to YES, IBM TCP/IP events will be traced. if this parameter is set to NO, then IBM TCP/IP events will not be traced. Note, that a separate parameter is used to control whether the simulated external events for IBM TCP/IP are traced or not. The parameter that controls the tracing of external interrupts is TRACETCPIPEXTINT. | **YES** | Yes | No |
| TRACETCPIPEXTINT | TRACE TCP/IP EXTERNAL INTERRUPT EVENTS.<br>(YES, NO)<br>This parameter controls if IBM TCP/IP external interrupt events should be traced or not. If this parameter is set to YES, then the simulated external interrupts used by IBM TCP/IP will be traced. If this parameter is set to NO, then the simulated external interrupts used by IBM TCP/IP will not be traced. | **NO** | Yes | No |
| TRACETCPIPRDWR | TRACE TCP/IP READ/WRITE EVENTS.<br>(YES, NO)<br>This parameter controls if IBM TCP/IP read/write events should be traced or not. If this parameter is set to YES, IBM TCP/IP read/write events will be traced. if this parameter is set to NO, then IBM TCP/IP read/write events will not be traced. | **NO** | Yes | No |
| TRACETEXTEVENTS | TRACE TEXT EVENTS.<br>(YES, NO) | **YES** | Yes | No |

| Parameter Name | Parameter Description | Default Value | Modifiable After Initialization | Output Only |
|---|---|---|---|---|
| TRACETODEVENTS | TRACE TOD EVENTS.<br>(YES, NO) | **YES** | Yes | No |
| TRACETSOEVENTS | TRACE TSO EVENTS.<br>(YES, NO)<br>This option controls if out-board TSO server events are logged to the wrap-around trace. | **YES** | Yes | No |
| TRACEURLREAD | TRACE HTTP RECEIVE STATES.<br>(YES, NO)<br>If this parameter is set, the server generates trace entries while receiving in-bound URL requests. The trace information is used mostly to diagnose problems when an in-bound HTTP request cannot be received. | **NO** | Yes | No |
| TRACEXCFEVENTS | TRACE XCF EVENTS.<br>This parameter is used to control the tracing of coupling facility (XCF) events. | **NO** | Yes | No |
| TRACEWEBAPIEVENTS | TRACE WEB API EVENTS.<br>(YES, NO) | **YES** | Yes | No |
| TRACEWLMCALLS | TRACE WLM API CALLS.<br>(YES, NO)<br>This parameter is used to control tracing of Shadow Server calls to the WLM APIs for transaction management. Specifying YES causes all calls to be traced. | **NO** | Yes | No |
| TRACEWTOMODULES | WTO MODULE ENTRY/EXIT MESSAGES.<br>(YES, NO)<br>This parameter is for debugging purposes only and should be used only under the guidance of technical support. | **NO** | Yes | No |
| TRACEWWWEVENTS | TRACE WWW EVENTS.<br>(YES, NO) | **YES** | Yes | No |

| Parameter Name | Parameter Description | Default Value | Modifiable After Initialization | Output Only |
|---|---|---|---|---|
| TRACEXIT | TRACE MODULE EXIT.<br>This parameter is for debugging purposes only and should be used only under the guidance of technical support. | **X‘07FE’** | Yes | No |
| TSOSRVTRACEOPER | TRACE TSOSRV OPERATIONS.<br>(YES, NO)<br>This parameter indicates whether TSO Server dispatching and control operations should be traced. | **NO** | Yes | No |

# PRODTSOSRV

| Parameter Name | Parameter Description | Default Value | Modifiable After Initialization | Output Only |
|---|---|---|---|---|
| TSOMAXSERVERS | TSOSRV MAXIMUM ACTIVE SERVER COUNT.<br>This parameter sets the maximum number of out-board TSO servers to be started and controlled by the TSO Server facility. The TSO facility will start no more than this many out-board servers.<br>**Minimum Value: 1 Maximum Value: 30** | **2 SERVERS** | Yes | No |
| TSOMINSERVERS | TSOSRV MINIMUM ACTIVE SERVER COUNT.<br>This parameter sets the minimum number of out-board TSO servers to be started and controlled by the TSO Server facility. The TSO facility will keep at least this many out-board servers active.<br>**Minimum Value: 1 Maximum Value: 30** | **0 SERVERS** | Yes | No |
| TSOSRVACTIVE | TSO SERVER FACILITY ACTIVE.<br>(YES, NO)<br>This parameter indicates whether or not the TSO Server facility is to be enabled. If set to YES during the initialization procedure, Shadow will enable and control out-board TSO Server address spaces. If set to NO, Shadow will not process TSO Servers.<br>This parameter cannot be changed after initialization. Also, if the operational environment is not correct for out-board TSO Server operation (e.g. Shadow is operating in 'test' mode under TSO), this parameter is ignored. | **NO** | No | No |
| TSOSRVALLOWRESTART | ALLOW RESTART OF SERVERS THAT FAILED INIT.<br>(YES, NO)<br>This parameter specifies whether the main product address space will attempt to restart an out-board TSO server which has failed during initialization. | **NO** | Yes | No |

| Parameter Name | Parameter Description | Default Value | Modifiable After Initialization | Output Only |
|---|---|---|---|---|
| TSOSRVCMDCPUTIME | TSOSRV MAXIMUM CPU TIME PER TSO COMMAND.<br>This parameter specifies, in seconds, the maximum amount of CPU time which any single command is allowed to consume. If the command exceeds, this CPU time limit is purged.<br>**Minimum Value: 1 Maximum Value: None** | **15 SECONDS** | Yes | No |
| TSOSRVCMDRUNTIME | TSOSRV MAXIMUM RUN TIME PER TSO COMMAND.<br>This parameter specifies, in seconds, the maximum amount of wall clock time any single command is allowed to execute within an out-board TSO server. If the command run time is exceeded, the current command is purged.<br>**Minimum Value: 1 Maximum Value: None** | **120 SECONDS** | Yes | No |
| TSOSRVCMDSECURITY | TSOSRV SECURITY.<br>This parameter specifies whether each command passed to an out-board TSO Server is executed under control of a transaction-level Userid, or under the Userid assigned to the out-board TSO Server address space. | **USERID** | No | No |
| TSOSRVCMDWAITTIME | TSOSRV MAXIMUM WAIT TIME PER TSO COMMAND.<br>This parameter specifies, in seconds, the maximum amount of wall clock time any single command is allowed to wait upon a resource. If the command remains in a wait state longer than this limit, the current command is purged.<br>**Minimum Value: 1 Maximum Value: None** | **120 SECONDS** | Yes | No |
| TSOSRVDORMANTTIME | TSOSRV MAXIMUM SERVER DORMANT TIME.<br>This parameter sets the time value, in seconds, after which a dormant out-board TSO server will be stopped.<br>**Minimum Value: 60 Maximum Value: 16777216** | **60 SECONDS** | Yes | No |
| TSOSRVMAXLINES | TSOSRV MAXIMUM OUTPUT LINE COUNT PER TSO COMMAND.<br>This parameter specifies the maximum number of data lines which a single command may output to SYSTSPRT. If the command exceeds this limitation, it is purged from the system. | **2000 LINES** | Yes | No |

| Parameter Name | Parameter Description | Default Value | Modifiable After Initialization | Output Only |
|---|---|---|---|---|
| TSOSRVMAXQUEUE | TSOSRV MAXIMUM EXECUTE QUEUE SIZE.<br>This parameter indicates the number of entry slots to be created in the TSO Server command queue.<br>**Minimum Value: 1 Maximum Value: None** | **1024 COMMANDS** | No | No |
| TSOSRVMSGID | TSOSRV SERVER DEFAULT PROFILE MSGID.<br>(YES, NO) | **NO** | Yes | No |
| TSOSRVPROCNAME | TSOSRV SERVER STARTED TASK NAME.<br>This parameter specifies the procedure name of the out-board TSO Server started task. This procedure name is used in all start commands which create a new out-board server address space. | **'SWSTSO'** | N/A | No |
| TSOSRVQUEUEADD | TSOSRV QUEUE DEPTH TO ADD A SERVER.<br>This parameter sets the number of execution command queue entries which must be waiting to be executed before an additional out-board TSO server is started. A new server is started only when the current number of servers is lower than the maximum limit. | **20** | Yes | No |
| TSOSRVQUEUEADDRESS | TSOSRV EXECUTE QUEUE ADDRESS.<br>This parameter displays the address of the TSO Server command execution header. The parameter is used for display purposes only. | **X'00000000'** | No | Yes |
| TSOSRVSERVERANCHOR | SERVER CONTROL BLOCK AREA ADDRESS.<br>This parameter can be displayed to show the anchor word where all out-board TSO Server control blocks are referenced. | **X'00000000'** | No | Yes |
| TSOSRVSMFRECORDING | TSOSRV TRANSACTION SMF RECORDING.<br>(YES, NO)<br>This parameter specifies whether SMF records are recorded for each out-board TSO Server address space at termination, and whether SMF records are generated to record execution statistics for each command executed within and out-board server. | **NO** | Yes | No |

| Parameter Name | Parameter Description | Default Value | Modifiable After Initialization | Output Only |
|---|---|---|---|---|
| TSOSRVSTARTUPPARM | TSOSRV SERVER OPTIONAL START PARAMETER.<br>This parameter specifies additional parameters which are appended to the start command used to create each new out-board server address space. | **NULL** | Yes | No |
| TSOSRVSTRIPPROMPTS | DEFAULT OUTPUT STRIPING FOR SYSTSPRT.<br>(YES, NO)<br>This parameter sets the default used to determine if certain messages sent to SYSTSPRT by out-board TSO servers are striped from the output stream. The messages striped are the echo of the command, the “READY” prompt, and certain ACF2 logon-time messages. The value of this option can be individually overridden using the STRIPPROMPTS command of ADDRESS TSO. | **YES** | Yes | No |
| TSOSRVSWAPPABLE | TSOSRV SERVERS ARE SWAPPABLE.<br>(YES, NO)<br>This parameter specifies whether each out-board TSO server address space is made non-swappable. | **YES** | Yes | No |
| TSOSRVTIMELIMIT | DEFAULT TSO CMD TIMEOUT FOR WWW RULES.<br>This parameter sets the default wait time limit value for TSO interface commands. If a TSO command issued to an out-board TSO server does not complete within the specified time limit, the requesting rule will resume execution. This default time limit value can be overridden within an individual REXX procedure using the SETTIMEOUT command.<br>**Minimum Value: 1 Maximum Value: 900** | **120 SECONDS** | Yes | No |

# PRODUSS

| Parameter Name | Parameter Description | Default Value | Modifiable After Initialization | Output Only |
|---|---|---|---|---|
| OEHFS | OPEN EDITION HFS SUPPORT.<br>This parameter controls whether or not Open Edition (Unix Systems Services) HFS file support is to be enabled. If ENABLE is specified, WWW rules may be created which use the PATH() keyword to map inbound URL requests to objects in the HFS file system. If DISABLE is specified, HFS access is not supported by the Server. If a PATH() keyword is used in a WWW rule definition, the rule will not be enabled.<br>**NOTE:** For this version of the Server, all access to the HFS files is performed under the control and authority of the Server's default runtime userid (DEFAULTRUNAUTH).<br>Both the default userid and the Server's address space level userid MUST have an OMVS segment and UID/GIF value defined by their perspective RACF, ACF/2, or TopSecret profiles. | **DISABLE** | No | No |

| Parameter Name | Parameter Description | Default Value | Modifiable After Initialization | Output Only |
|---|---|---|---|---|
| DOCMENTROOT | HFS DOCUMENT ROOT PATH PREFIX.<br>This parameter can be used to specify an HFS directory root path. The value, if specified, is prepended to all RELATIVE pathname constructs in order to formulate the ABSOLUTE pathname. It is NOT prepended to ABSOLUTE pathname constructs. (An ABSOLUTE pathname begins with "/"; A RELATIVE pathname does not begin with "/").<br>In order to make ruleset-level and application-level promotions easier to manage, use this HFS directory prefix to differentiate between production and test copies of the Server.<br>The value coded for this parameter is CASE SENSITIVE, and must begin and end with a “/”, and is limited to 64 bytes in length.<br>**NOTE:** ALL HFS pathname constructs are limited internally to a total of 256 bytes in length;, i.e. the length of this prefix, plus the ruleset-level HFSROOT() prefix (if non-NULL), plus the rule defintion PATH() length, plus additional URL request bytes substituted for wildcards into the PATH() operand at execution time. | **NULL** | No | No |

# PRODWLM

| Parameter Name | Parameter Description | Default Value | Modifiable After Initialization | Output Only |
|---|---|---|---|---|
| WLMCLASSPLAN | CLASSIFY USING DB2 PLAN NAME.<br>(YES, NO)<br>This parameter controls whether or not WLM will use the DB2 plan name when classifying Shadow Server transactions. If YES is specified, WLM will use the DB2 plan name as a criterion when looking for a classification rule match.. | **NO** | Yes | No |
| WLMCLASSSPM | CLASSIFY USING SUBSYSTEM PARAMETER.<br>(YES, NO)<br>This parameter controls whether or not WLM will use the subsystem parameter (WLMSUBSYSPARM) when classifying Shadow Server transactions. If YES is specified, WLM will use the subsystem parameter as a criterion when looking for a classification rule match. | **NO** | Yes | No |
| WLMCLASSTRAN | CLASSIFY USING TRANSACTION NAME.<br>(YES, NO)<br>This parameter controls whether or not WLM will use the transaction name when classifying Shadow Server transactions. If YES is specified, WLM will use the Shadow transaction name as a criterion when looking for a classification rule match. | **NO** | Yes | No |
| WLMCLASSUSER | CLASSIFY USING USERID<br>(YES, NO)<br>This parameter controls whether or not WLM will use the userid when classifying Shadow Server transactions. If YES is specified, WLM will use the userid as a criterion when looking for a classification rule match. | **NO** | Yes | No |

| Parameter Name | Parameter Description | Default Value | Modifiable After Initialization | Output Only |
|---|---|---|---|---|
| WLMCONNECT | INITIALIZE WLM SUPPORT.<br>(YES, NO)<br>This parameter specifies whether or not the Shadow Server address space is to attempt to connect to the OS/390 WorkLoad Manager (WLM) as a WLM Work Manager. If YES is specified, the Shadow Server will use WLM enclaves for transaction execution. | **NO** | No | No |
| WLMSUBSYSNAM | WORKMANAGER SUBSSYSTEM NAME.<br>This parameter is used to identify the Shadow Server address space. The combination of WLMSUBSYSTEM and WLMSUBSYSNAM uniquely identifies an address space to WLM. This parameter defaults to the Shadow Server subsystem ID. | **'SDBB' (SD)**<br>**'SWSS' (SWS)** | No | No |
| WLMSUBSYSPARM | WORKMANAGER SUBSSYSTEM PARAMETER<br>This parameter can be used to provide an arbitrary identifier for this Shadow server address space. This parameter in conjunction with the WLMCLASSSPM parameter, can be used to classify all work for this Shadow Server address space, into a particular WLM service class. | **NULL** | Yes | No |
| WLMSUBSYSTEM | WORKMANAGER SUBSSYSTEM TYPE<br>This parameter is used to specify the subsystem type to be used for the Shadow Server address space. The subsytem type is used to select the transaction classification rules, which determine the WLM service class to be used for a transaction. This parameter defaults to the first three characters of the Shadow subsystem ID. | **'SDB' (SD)**<br>**'SWS' (SWS)** | No | No |

# PRODWWW

| Parameter Name | Parameter Description | Default Value | Modifiable After Initialization | Output Only |
|---|---|---|---|---|
| ASCIIEBCDICMAPPING | ASCII/EBCDIC LANGUAGE MAPPING.<br>This parameter specifies the national language table set to be used by the Web Server when performing ASCII to EBCDIC and EBCDIC to ASCII conversions. | **ENU** | Yes | No |
| CLOSEDELAY | CLOSE DELAY TIME FOR WWW CLIENTS.<br>**Minimum Value: 0 Maximum Value: 10000** | **0 SECONDS** | Yes | No |
| GLVSTATETRACEDEFAULT | DEFAULT GLVSTATE.$TRACE VALUE.<br>YES, NO<br>This parameter sets the default GLVSTATE.$TRACE value for web transaction programs. If NO, tracing of individual Automated State Management Facilitly (ASMF) is not preformed unless requested; if YES, all ASMF operations are traced. | **NO** | Yes | No |
| HTTPRESPMODE | HTTP RESPONSE MODE FOR NEW URLS.<br>This parameter can be used to set the HTTP response processing mode under which each new web transaction begins operation. NEON Systems strongly recommends you use the default value for this parameter unless needed to maintain operational compatibility with older versions of the Server. | **SERVER-PARSED** | Yes | No |

| Parameter Name | Parameter Description | Default Value | Modifiable After Initialization | Output Only |
|---|---|---|---|---|
| MAXCHAINEDBUFFERS | MAX RESP BUFFERS TO TRIGGER AUTO-FLUSH<br>This parameter, when it is set to a non-zero value, specifies that the number of currently-held 32k output buffers is monitored at each output event. Whenever the inuse buffer count reaches the MAXCHAINEDBUFFERS value, the server automatically issues a SWSSEND (FLUSH) request.<br>Monitoring of this threshhold is only active when the current transaction is operating in “NON-PARSED-HEADER” (RESPONSEMODE(NONE)) mode. This type of monitoring and intermediate flush-to-client operation is never active when a web transaction is operation in “SERVER-PARSED-HEADER” (RESPMODE (SERVER) mode.<br>We recommend that automatic threshhold checking not be used, unless needed to maintain operational compatibility with an older release of the Server. Instead,use the RESPMODE (NONE) and AUTOFLUSH (nnnn) keywords on an individual /*WWW header.<br>**Minimum Value: 0 Maximum Value: 32767** | **0 BUFFERS** | Yes | No |
| MAXHTTPRESPBUFFERS | MAX HTTP RESP BUFFERS HELD BY A TASK.<br>This parameter exerts a limitation on the total number of 32k outbound HTTP response buffers which any single URL transaction may simultaneously hold in storage.<br>As an HTTP response is being generated, data which is being buffered occupies 32k out-bound buffers, and is not normally transmitted until the transaction procedure ends. When a non-zero value is set, it limits the total number of 32k out-bound response buffers which may be concurrently held by a single web transaction subtask. If this limit is exceeded, the Server generates a user ABEND X’722’ with reason code 500 in order to cancel the transaction procedure. If this parameter is set to zero, the number of concurrently held output buffers is not monitored.<br>**Minimum Value: 0 Maximum Value: 32767** | **0 BUFFERS** | Yes | No |

| Parameter Name | Parameter Description | Default Value | Modifiable After Initialization | Output Only |
|---|---|---|---|---|
| MAXHTTPRESPBYTES | MAXIMUM BYTES FOR ANY HTTP RESPONSE.<br>This parameter sets a global limit on the total number of data bytes which can be output in response to any individual URL request.<br>This limit operates strictly by monitoring the bytes, if any, which have already been transmitted as part of the HTTP response,plus the count of bytes currently buffered awaiting transmission. If this limit is exceeded, the Server generates a user ABEND X'722' with REASON CODE 501, in order to cancel the entire transaction subtask.<br>If set to zero, no byte count limitationi s imposed.<br>**Minimum Value: 0 Maximum Value: 2147483647** | **10485760 BYTES** | Yes | No |
| NOHTXAUTOEXPIRE | SUPPRESS AUTOMATAIC "EXPIRES:" HEADERS<br>YES, NO<br>This parameter controls whether the Server will automatically generate an "Expires:" HTTP response header when a dynamicallly generated response, tailored by the HTML Extension Facility, has been buffered for output.<br>If NO, the Server generates an "Expires:" response header (containing the current date and time) automatically, provided that no other "Expires:" response header was buffered for output while creating the response.<br>If set to YES, the Server does not automatically generate "Expires:" HTTP response headers. | **NO** | Yes | No |
| SHORTNESTEDDELIMS | ALLOW SHORT NESTED DELIMITERS.<br>YES, NO<br>This paramter controls whether or not HTX parsing will allow for short nested delimiters of the form '<..>'. | **YES** | Yes | No |

| Parameter Name | Parameter Description | Default Value | Modifiable After Initialization | Output Only |
|---|---|---|---|---|
| TRACEASMFRESPONSE | TRACE ASMF RESPONSE/EOT PROCESSING.<br>YES, NO<br>This parameter controls tracing of out-bound Automated State Management Facility (ASMF) operations. If off, these operations are not traced unless unsuccessful. If on, all response-time and end-of-transaction-time processing by ASMF is traced. | **NO** | Yes | No |
| TRACEASMFREQUESTS | TRACE ASMF REQUEST PROCESSING<br>YES, NO<br>This paramter controls tracing of in-bound request processing by the Automated State Management Facility (ASMF). Restoration operations are traced when this parameter is set. Only unsuccessful in-bound processing generates a trace message if this parameter is off. | **YES** | Yes | No |
| TRACEHTML | DEFAULT SENDTRACE SETTING.<br>(YES, NO)<br>This parameter is the global default value for each WWW rule definition's SENDTRACE keyword. If set to YES, SENDTRACE(YES) is the assumed default for all WWW rules. | **NO** | Yes | No |
| TRANSINTTIMINGS | TRACE TIMINGS FOR WWW SERVICES.<br>(YES, NO).<br>When set to YES various API routines of Shadow OS/390 Web Server generate trace records to record entry and exit events. Using the "D CPU" command in trace browse will display the CPU time column. The CPU timings for these event records are precise, regardless of the setting of the PRECISECPUTIME option. You can use this option to display the amount of CPU time required to process various WWW services, such as buffering a cached file for transmission, or processing HTML extensions. It is recommended that you use NOT set this option except to periodically collect performance data. | **NO** | Yes | No |
| TRACEURL | TRACE INBOUND HTTP REQUESTS.<br>(YES, NO) | **YES** | Yes | No |

| Parameter Name | Parameter Description | Default Value | Modifiable After Initialization | Output Only |
|---|---|---|---|---|
| TRACEURLPARSE | DEFAULT PARSETRACE SETTING.<br>(YES, NO)<br>This parameter is the global default value for each WWW rule definition's PARSETRACE keyword. If set to YES, PARSETRACE(YES) is the assumed default for all WWW rules. | **NO** | Yes | No |
| WWWSEFLAG01 | WWW SE FLAG 01.<br>(YES, NO) | **NO** | Yes | No |

# Obsolete

| Parameter Name | Parameter Description | Default Value | Modifiable After Initialization | Output Only |
|---|---|---|---|---|
| ASCII_EBCDIC_MAPPING | ASCII/EBCDIC LANGUAGE MAPPING. | **ENU** | Yes | No |
| AUTHPROSET | AUTHORIZATION RULESET NAME. | **'ATH'** | No | No |
| EPRIVLIMIT | EPRIVATE STORAGE UTILIZATION LIMIT.<br>This parameter was used to control how much extended private area storage the product should be allowed to allocate. This parameter is now obsolete. Extended private area storage is now managed by the system to provide maximum reliability and availability.<br>**Minimum Value: 1048576 Maximum Value: 2147483647** | **2097151K** | N/A | Yes |
| EPROALTFIX | SEF RULESET DATASET NAMES ALTERNATE PREFIXES. | **NULL** | No | No |
| EPROPREFIX | SEF RULESET DATASET NAMES PREFIX. | **'CSD.AI38.SV030100.S'** | No | No |
| EPROSUBSYS | SEF RULESET DATASETS SUBSYSTEM NAME. | **NULL** | No | No |
| EPROSUFFIX | SEF RULESET DATASET NAMES SUFFIX. | **'EXEC'** | No | No |
| EXECDATASETNAME | REXX EXEC DATA SET NAME. | **'CSD.AI38.SV040100.E'** | Yes | No |
| FREEACEEBLOCKS | FREE ACEE CONTROL BLOCKS.<br>(YES, NO)<br>This parameter controls if a RACROUTE DELETE command should be issued out of End-Of-Task processing to free the ACEE created for ODBC threads. The default should be YES. However, this causes abends in RACF processing in some cases (IBM bugs). | **YES** | Yes | No |
| MAXLONGVARCHAR | MAXIMUM LONG VARCHAR DATA LENGTH.<br>The maximum allowable length of a LONG VARCHAR field. Under some circumstances the maximum must be set low so that LONG VARCHAR data can be sent using a 32K buffer. Note that the actual data can not be longer than the value set below. | **1000000 BYTES** | Yes | No |

| Parameter Name | Parameter Description | Default Value | Modifiable After Initialization | Output Only |
|---|---|---|---|---|
| PRELOADEXECS | PRELOAD REXX EXECS INTO STORAGE.<br>(YES, NO) | **NO** | Yes | No |
| PRIVLIMIT | PRIVATE STORAGE UTILIZATION LIMIT.<br>This parameter was used to control how much below the 16 MB line private area storage the product should be allowed to allocate. This parameter is now obsolete. Below the 16 MB line private area storage is now managed by the system to provide maximum reliability and availability.<br>**Minimum Value: 131072 Maximum Value: 8388608** | **12288K** | N/A | Yes |
| RUNSDF | CLIENTS CAN USE THE SDF PROGRAM.<br>(YES, NO) | **NO** | Yes | No |
| TCPAPITYPE | TCP/IP IUCV API TYPE. | **0** | No | No |
| TRACEREMOTERPC | TRACE REMOTE PROCESSING RPC.<br>(YES, NO) | **NO** | Yes | No |
| TRANSACTIONTIMEOUT | TRANSACTION TIMEOUT VALUE.<br>This parameter can be used to limit the wait time for the completion of a transaction. If the transaction times out, a message is placed in the communication buffer to notify the client that a timeout has occurred. | **0 SECONDS** | Yes | No |
| TYPEPROSET | TYP RULESET NAME. | **'TYP'** | No | No |
| USECMCO | USE CMCO CONTROL BLOCKS.<br>(YES, NO)<br>This parameter forces a different set of control blocks to be used to send SQL requests between two hosts. Mainframe to mainframe SQL processing is no longer supported by Shadow Server on the host. This parameter is obsolete and should never be set or used in any way. | **NO** | Yes | No |

| Parameter Name | Parameter Description | Default Value | Modifiable After Initialization | Output Only |
|---|---|---|---|---|
| USECMOF | USE CMOF CONTROL BLOCKS.<br>(YES, NO)<br>This parameter forces a different set of control blocks to be used to send SQL requests between two hosts. Mainframe to mainframe SQL processing is no longer supported by Shadow Server on the host. This parameter is obsolete and should never be set or used in any way. | **NO** | Yes | No |
| WWWEPROSET | AUTHORIZATION RULESET NAME. | **'WWW'** | No | No |

# Index


## A

ACF2SAFCALL 77
ADABAS 3
ADABASAUTOMAP 3
ADABASNODB2 3
ADABASSECURITY 3
ADABASUID 3
ADDITIONALSQLDATA 93
ADJUSTREGIONSIZE 56
ALLOCSECURITYHIGH 77
APPC/IMS 45
APPLID 19
ARCHIVEDATACLASS 11
ARCHIVEDEFCLPARMS 11
ARCHIVEDSNPREFIX 11
ARCHIVEMGMTCLASS 11
ARCHIVESTORCLASS 11
ASCII_EBCDIC_MAPPING 136
ASCIIEBCDICMAPPING 131
ATHINDEX 89
AUTHPROSET 136
AUTOCANCELTM 56
AUTOCOMMITCALL 93
AUTOCOMMITCC 93
AUTOSTATICCOMMIT 93
AUTOSTATICDEFER 93
AUTOSTATICSQL 93
AUTOUSERID 94

## B

BASEINTERVAL 56
BLOCKFETCH 94
BMP 45
BMPALTTCB 45
BMPNAME 45
BMPPARM 45
BROWSEARCHIVE 12
BROWSEARCHIVECOUNT 12
BROWSEARCHIVECUSHION 12
BROWSEBLOCKS 12
BROWSEDDNAME 13
BROWSEINTERVAL 13
BROWSEMAX 13
BYPASSCOMPRESSION 19
BYPASSRRS 75
BYPASSRRSEXITS 75

## C

CALLMAXROWS 69
CALLROWSSIZE 69
CANCELWAITTIME 56
CENSORAPIDATAVALUES 77
CENSORHTTPRESP 77
CENSORURLAUTHDATA 77
CENSORURLQUERYDATA 78
CHECKCONVIDINTERVAL 4
CHECKDATAINTERVAL 57
CHECKLIMITSINTERVAL 57
CHECKRPCAUTHORITY 69
CHECKSESSIONS 57
CHECKTOKENSINTERVAL 105
CICSCONNECTRETRY 15
CICSDATACONV 15
CICSIRCSVCNO 15
CICSIRCSVCVR 15
CICSLOADBALANCE 15
CICSMAXCONNECTIONS 15
CICSPROCOWNER 15
CICSSUBSYSTEM 15
CICSTXNSERVERNAME 16
CICSTXNTIMEOUT 16
CLEARARCHIVERECOVERY 13
CLIENTLOGON 78
CLIENTQUIESCEDELAY 57
CLOSEDELAY 131
CLOSEWITHDATA 94
CONNECTIONTEXT 48
CONNECTRETRYINT 19
CONNECTTIMEOUT 19
CONVERTNULLS 45
CREATEGLOBAL 94
CSA 101
CSALIMIT 101
CURRENTCPU 48
CURRENTTOKENADDRESS 105
CURRENTTOKENBLOCK 105

## D

DATASIZE 101
DB2ATTACHFACILITY 94
DB2CONCURRENTCN 48
DB2CONCURRENTHW 48
DB2CONCURRENTMX 48
DB2LICCONCURMX 48
DBCSTABLENAME 58
DBCTL 45
DEBUGOPTIONS 58
DEBUGSEFWAIT 107
DEBUGSGMG 107
DEFAULTCPUTIME 58
DEFAULTDB2PLAN 94
DEFAULTDB2PROCNAME 95
DEFAULTDB2SUBSYS 95
DEFAULTEXCIFORMURL 42

DEFAULTGENFORMURL 42
DEFAULTIMSFORMURL 42
DEFAULTRPCPARM 69
DELETEEXCIMODULES 16
DFHSM 28
DFHSMCLEANUPINTERVAL 28
DFHSMDRAIN 29
DFHSMDRAINAUTO 29
DFHSMSHUTDOWNWAIT 29
DFHSMSTATUS 30
DISPATCH 58
DLLIBDDNAME 58
DOCMENTROOT 128
DSPC 58
DURETRY 16
DYNAMICSQL 95

## E

ECSA 101
ECSALIMIT 101
EMINPRIV 101
ENABLEMDIAPI 95
EPRIV 101
EPRIVLIMIT 136
EPROALTFIX 136
eproaltfix 136
EPROINDEX 89
EPROPREFIXeproprefix 136
EPROSOURCETEXT 89
EPROSUBSYS 136
EPROSUFFIX 136
EPROTRACE 89, 107
ERRORCPUTIME 58
ERRORSTACKSIZE 102
ERRORWAITTIME 58
EXCI 16
EXCIAPITYPE 16
EXCICALLBYREF 16
EXCICONNECTIONNAME 17
EXCIDATACONV 17
EXCINDEX 89
EXCIPIPEPREALLOC 17
EXCIPIPEPREOPEN 17
EXECDATASETNAME 136
EXECDSNAME 58
EXPANDEDSQLBLOCKS 95
EXPIRATIONDATE 48
EXPIRATIONDAYS 48
EXPOSEWWWPASSWORD 78
EXTRACTDATACLASS 13
EXTRACTDEFCLPARMS 13
EXTRACTDSNPREFIX 13
EXTRACTMGMTCLASS 14
EXTRACTSTORCLASS 14
EXTRAINTERVAL 58
EXTTRACE 17

## F

FAILCPUTIME 59
FAILENQHOLDTIME 69
FAILEXCLUSIVETIME 59
FAILSHARETIME 59
FAILSQLCPUTIME 59
FAILUPDATETIME 59
FAILWAITTIME 59
FILEAPIRECALL 30
FILECACHE 30
FILECLOSEAFTER 31
FILEHRECALL 32
FILEIOMODE 33
FILEMESSAGES 33
FILEMOUNT 33
FILERECALL 33
FILEREXXTOOLRECALL 34
FILERULERECALL 34
FILESECURITY 35
FILESHAREDDN 35
FILESHAREDSN 35
FILESTAGINGSIZELIMIT 36
FIRSTCPU 48
FORCESECURITYLOW 78
FREEACEEBLOCKS 136

## G

GENERATEFONTORDERS 42
GENERATESTYLECOLOR 43
GETLOGONMESSAGES 79
GETSECONDARYLIST 96
GLOBALADDR 38
GLOBALALLOC 38
GLOBALBACKUPCOUNT 38
GLOBALBACKUPDSN 38
GLOBALBACKUPEND 38
GLOBALBACKUPINTVAL 38
GLOBALBACKUPMDSCB 38
GLOBALBACKUPNEXT 38
GLOBALBACKUPPROC 38
GLOBALBACKUPSTART 38
GLOBALBACKUPUNIT 38
GLOBALBLOCKS 38
GLOBALBLOCKSUSED 38
GLOBALCHECKCOUNT 38
GLOBALCONTROL 39
GLOBALDATE 39
GLOBALDIV 39
GLOBALFREE 39
GLOBALFREEAREAS 39
GLOBALINTERVAL 39
GLOBALLENGTH 39
GLOBALMAX 39
GLOBALMSGS 39
GLOBALNEXT 39
GLOBALPAGES 39
GLOBALPOOL 39
GLOBALREBUILD 39

GLOBALRESTORETIME 39
GLOBALRETRY 39
GLOBALSIZE 39
GLOBALSUBPOOL 39
GLOBALTCB 39
GLOBALTEMPADDR 40
GLOBALTEMPALLOC 40
GLOBALTEMPBLKSUSED 40
GLOBALTEMPFREE 40
GLOBALTEMPFREEAREAS 40
GLOBALTEMPMAX 40
GLOBALTEMPUPDATE 40
GLOBALTEMPUSED 40
GLOBALTEMPWARNIV 40
GLOBALTEMPWARNTH 40
GLOBALTIME 40
GLOBALTOKEN 40
GLOBALUPDATE 40
GLOBALUPDATECHECK 40
GLOBALUSED 41
GLOBALWARNINTVAL 41
GLOBALWARNTH 41
GLOBALWARNTHRESH 41
GLVCHAINMAX 41
GLVINDEX 89
GLVPENDINGHIGH 41
GLVPENDINGMAX 41
GLVSHAREDFILE 41
GLVSTATETRACEDEFAULT 131
GRANTGLOBAL 96
GROUPDIRECTOR 59
GROUPNAME 59
GTF 17

## H

HEXIPSOURCE 79
HFSAUTHMODE 79
HIGHMODULEDATE 60
HIGHMODULENAME 60
HIGHMODULETIME 60
HIGHMODULEVERSION 60
HOSTFUNCTIONALLEVEL 96
HRECALLWAIT 37
HRECALLWAITMAX 37
HTMLBACKGROUNDCOLOR 43
HTMLDEFAULTFONTCOLOR 43
HTMLFONTBLUE 43
HTMLFONTGREEN 43
HTMLFONTHILIGHT 43
HTMLFONTPINK 43
HTMLFONTRED 44
HTMLFONTTURQUOISE 44
HTMLFONTWHITE 44
HTMLFONTYELLOW 44
HTMLINPUTHILIGHT 44
HTMLINPUTWHITE 44
HTTPRESPMODE 131

## I

IBMHOSTDOMAIN 19
IBMPORTNUMBER 19
IBMSSLPORTNUMBER 20
IDENTIFYDSNHLI 96
IGNOREDCODE01 96
IGNOREDCODE02 96
IGNOREDCODE03 96
IGNOREDCODE04 96
IGNOREDCODE05 96
IGNOREDCODE06 96
IGNOREDCODE07 96
IGNOREDCODE08 96
IGNOREDCODE09 97
IGNOREDCODE10 97
IMPLEMENTTLS 80
IMSBMPTIMEOUT 45
IMSBMPWAITTIME 45
IMSCLASS 46
IMSCNVIDTBLSZ 4
IMSCONVTYPE 4
IMSDDNAME 46
IMSDEALLOCONVTIME 4
IMSDEFAULTMAPNAME 4
IMSDLIPRMLOC 46
IMSDSNAME 46
IMSFPBUFFERS 46
IMSFPOVERFLOW 46
IMSFUNCLEVEL 46
IMSGROUPNAME 46
IMSID 46
IMSLOCALLU 5
IMSLTERMTABLESEQ 45
IMSLUEE0 5
IMSLUEE0ESCSEQ 5
IMSMAXTHREADS 46
IMSMINTHREADS 46
IMSMODENAME 6
IMSNBABUFFERS 46
IMSOTMA 46
IMSPARTNERLU 6
IMSPROCOWNER 46
IMSQUEUEKEEPTIME 6
IMSRCVALLOCTIMEOUT 6
IMSRECVALLOCTYPE 7
IMSRETURNCONTROL 7
IMSSECURITYTYPE 8
IMSSUFFIX 46
IMSSYMDEST 9
IMSSYNCLEVEL 9
IMSTIMEOUT 47
IMSTXNTIMEOUT 10
IMSUSERID 47
IMSWAITTIME 47
ISPLLIBDSNAME 60
ISPMLIBDSNAME 60
ISPPLIBDSNAME 60
ISPSLIBDSNAME 60

ISPTLIBDSNAME 60
ITCHOSTDOMAIN 20
ITCKEEPALIVEOPTION 20
ITCKEEPALIVETIME 20
ITCLISTENQDEPTH 20
ITCMAXBUFFERSIZE 20
ITCPORTNUMBER 20
ITCSUBSYSTEM 21
ITCTCLOSETIMEOUT 21

## K

KEEPALIVE 21
KEEPALIVELIMIT 22
KEEPALIVETIMEOUT 22
KILLWAITPOST 61

## L

LE370LIBKEEP 69
LERUNTIMEOPTS 61
LICENSECODE 48
LOGAPMVSSUM 50
LOGAPMVSSUMTABLE 50
LOGDB2PLNAME 50
LOGDB2SUBSYS 50
LOGDELAY 50
LOGEXCEPTIONSTABLE 51
LOGFAILURELIMIT 51
LOGINTERVALS 51
LOGINTERVALSTABLE 51
LOGMEMORYAPPC/MVS 51
LOGMEMORYINTERVALS 52
LOGRETAINAPMVSSUM 52
LOGRETAININTERVALS 52
LOGRETAINSESSIONS 52
LOGRETAINSQL 53
LOGRETAINURLS 53
LOGSESSIONS 53
LOGSESSIONSTABLE 53
LOGSOURCETABLE 53
LOGSQLSOURCE 54
LOGSTORAGE 54
LOGSTORAGETABLE 54
LOGURLS 54
LOGURLSTABLE 54
LOGUSERID 54
LOGWAIT 55
LOGWARNINGLIMIT 55
LOOKASIDESIZE 97

## M

MAXABENDRATE 61
MAXCHAINEDBUFFERS 132
MAXCMDRATE 61
MAXCPUTIME 61
MAXDB2ACTIVETHREADS 97
MAXHTTPRESPBUFFERS 132
MAXHTTPRESPBYTES 133
MAXLICENSEDCLIENTS 49
MAXLOGRATE 61
MAXLONGVARCHAR 136
maxlongvarchar 136
MAXMSGRATE 62
MAXROWS 97
MAXTIMERONS 97
MAXUDPSIZE 22
MDIERRORCODE 97
MDISQLSTATE 98
MDISTORAGEVALUE 98
MINCPUTIME 62
MINPRIV 102
MONITORAPPC/MVS 10
monospace fonts vi
MSGCASE 17
MSGDRAINRATE 89
MSGROUTEFROM1 22
MSGROUTEFROM2 22
MSGROUTETO1_1 22
MSGROUTETO1_2 22
MSGROUTETO1_3 22
MSGROUTETO1_4 22
MSGROUTETO1_5 22
MSGROUTETO2_1 23
MSGROUTETO2_2 23
MSGROUTETO2_3 23
MSGROUTETO2_4 23
MSGROUTETO2_5 23
MSGTHRESHOLD 89

## N

NEONMRO 17
NETMODE 23
NETWORK ADJUST 23
NETWORKBUFFERSIZE 23
NOCATCHUP 89
NOHTXAUTOEXPIRE 133
NTCFORCEACCEPT 23
NTCINTERFACE 23
NTCLISTENQDEPTH 23
NTCPORTNUMBER 24
NTCSUBSYSTEM 24

## O

ODBCCALLRPCS 69
ODBCCATALOGLEVEL 98
ODBCOVERHTTP 98
OEASYNCIO 24
OEHFS 127
OEHOSTDOMAIN 24
OEMVENDOR 49
OEPORTNUMBER 25
OESSLPORTNUMBER 25
OESTACK 25
OPTROWS 99
organization
    publication v

## P

Parameters 1
PARAMPLIST 69
PASSEMPTYGROUPNAME 80
PASSIMSGROUPNAME 80
PBFU 70
PRECISECPUTIME 107
PREFETCH 99
PREFETCHROWS 99
PRELOAD 70
PRELOADEXECS 137
PREPARECALLRPCS 70
PRESENDBLOCKS 100
PRIMARYSTACKHW 102
PRIMARYSTACKMAX 102
PRIMARYSTACKSIZE 102
PRIV 102
PRIVLIMIT 137
PROCESS 62
PROCESSEP 62
PROCESSPC 62
PROCESSSVC 62
PROCESSTCB 63
PRODADABAS 3
PRODFAMILY 49
PRODFEATURES 49
PRODRRS 75
PRODRRSGROUP 75
PRODUSS 127
PROVIDEPASSWORDS 81

## Q

QUICKREFOPTIONS 63
QUIESCESYSTEMTYPE 63

## R

RACFGROUPLIST 81
Reader’s Comment form ix
READONLY 3
REALTIMESUMMARY 10
RECORDINGINTERVAL 55
RECTABLEENTRIES 75
RESERVEEHIGH 103
RESERVEELOW 103
RESERVEELSQA 103
RESERVEHIGH 103
RESERVELOW 104
RESERVELSQA 104
RESETCONTROLINTERVAL 63
RESOURCEMGRNAME 76
RESOURCETYPE 81
REUSECLIENTACEES 81
REUSETHREADS 63
REVERSEMATCH 89
REXXDEFAULTADDRESS 68
REXXMAXCLAUSES 68
REXXMAXCOMMANDS 68
REXXMAXPGMSIZE 68
REXXMAXQUEUE 68
REXXMAXSAYS 68
REXXMAXSECONDS 68
REXXMAXSTRINGLENGTH 68
ROLLBACKRPCABEND 70
RPC01SPECIALREQ 70
RPC02SPECIALREQ 70
RPC03SPECIALREQ 70
RPC04SPECIALREQ 71
RPC05SPECIALREQ 71
RPC06SPECIALREQ 71
RPC07SPECIALREQ 71
RPC08SPECIALREQ 71
RPC09SPECIALREQ 71
RPC10SPECIALREQ 71
RPC11SPECIALREQ 71
RPC12SPECIALREQ 71
RPC13SPECIALREQ 71
RPC14SPECIALREQ 71
RPC15SPECIALREQ 71
RPC16SPECIALREQ 71
RPC17SPECIALREQ 72
RPC18SPECIALREQ 72
RPC19SPECIALREQ 72
RPC20SPECIALREQ 72
RPC21SPECIALREQ 72
RPC22SPECIALREQ 72
RPC23SPECIALREQ 72
RPC24SPECIALREQ 72
RPC25SPECIALREQ 72
RPC26SPECIALREQ 72
RPC27SPECIALREQ 72
RPC28SPECIALREQ 72
RPC29SPECIALREQ 72
RPC30SPECIALREQ 73
RPC31SPECIALREQ 73
RPC32SPECIALREQ 73
RPC33SPECIALREQ 73
RPC34SPECIALREQ 73
RPC35SPECIALREQ 73
RPC36SPECIALREQ 73
RPC37SPECIALREQ 73
RPC38SPECIALREQ 73
RPC39SPECIALREQ 73
RPC40SPECIALREQ 73
RPC41SPECIALREQ 73
RPC42SPECIALREQ 73
RPC43SPECIALREQ 74
RPC44SPECIALREQ 74
RPC45SPECIALREQ 74
RPC46SPECIALREQ 74
RPC47SPECIALREQ 74
RPC48SPECIALREQ 74
RPC49SPECIALREQ 74
RPC50SPECIALREQ 74
RPCINDEX 89
RPCSUBPOOL 74
RRS 76

RRSCONTEXT 76
RULESETSEFPROXY 82
RUNSDF 137

## S

SAFVERSION 83
SDB Started Task Parameters 1
SECURITYMSGSUPP 83
SECURITYPACKAGE 83
SECURITYVERSION 83
SEFACTIVE 89
SEFCMDQUEUE 89
SEFDEFAULTADDRESS 89
SEFDESC 90
SEFDEST 90
SEFEXECQUEUE 90
SEFFIRELIMIT 90
SEFGLVEVENTS 90
SEFINITREXX 90
SEFLIMITDISABLE 90
SEFMAXCLAUSES 90
SEFMAXCOMMANDS 90
SEFMAXPGMSIZE 90
SEFMAXQUEUE 90
SEFMAXSAYS 90
SEFMAXSECONDS 90
SEFROUTE 91
SEFSIZE 91
SEFSUBPOOL 91
SEFV3COMPATIBLE 91
SESSIONFAILTIME 64
SESSIONQUEUEADDRESS 64
SESSIONWAITINTERVAL 18
SESSIONWAITTIME 18
SEVERRPCABEND 74
SHARERUNAUTHACEES 83
SHARESUBPOOLZERO 104
SHUTDOWNWAIT 64
SMFEPRODISABLE 91
SMFFULLSQL 107
SMFNUMBER 108
SMFTRACEASTEXT 108
SMFTRANSACT 108
SOCKETLINGER 26
SPECIALTABLEPREFIX 100
SQLINDEX 91
SSL 83
SSLACCEPTTIMEOUT 26
SSLAUTODETECT 84
SSLENCLAVETERMINATE 84
SSLINITIALIZED 84
SSLINSTALLTYPE 84
SSLUSERID 85
SSSCLIENTEXPD 85
SSSCLIENTSTEAL 85
SSSDELETEAFTER 86
SSSEVENTSTEAL 86
SSSEXITACTIVE 86
SSSEXITREUSE 86
SSSMAXCLIENTS 87
SSSMAXEVENTS 87
SSSMAXMSGS 87
SSSMSGEXPD 87
SSSMSGSTEAL 87
STACKINCREMENTAMOUNT 104
STATICSQL 100
support, technical viii
SWIURLNAME 64

## T

TARGETTHREADCOUNT 64
TCPAPITYPE 137
TCPMAXSESSIONS 26
TCPMSGLIM 26
TCPNAME 26
TCSSLPORTNUMBER 21
technical support viii
TERMINATEINTERVAL 65
TERMINATELOGGING 55
THREADLEVELTRACE 108
THREADREUSELIMIT 65
THREADTIMEOUT 65
TIMEOUT 18
TODINDEX 91
TOKENBLOCKCOUNT 105
TOKENBLOCKPTR 105
TOKENENTRYCOUNT 105
TOKENSALLOCATED 105
TOKENSDELETED 105
TOKENSINUSE 105
TOKENSTIMEDOUT 105
TOKENSTORAGE 106
TOKENTIMEOUT 106
TRACE 109
TRACE24GETS 109
TRACEABENDEVENTS 109
TRACEABENDRETRYINFO 109
TRACEABENDSDWARC1 109
TRACEACEECHANGES 110
TRACEAPPCDATA 110
TRACEAPPCMVSEVENTS 110
TRACEAPPCMVSMN 110
TRACEAPPCMVSSR 110
TRACEASMFREQUESTS 134
TRACEASMFRESPONSE 134
TRACEATTACHEVENTS 110
TRACEAUTHEVENTS 110
TRACEBROWSECOUNT 65
TRACEBROWSEGROUP1 110
TRACEBROWSEGROUP2 111
TRACEBROWSEGROUP3 111
TRACEBROWSEGROUP4 111
TRACEBROWSEMAXLIMIT 65
TRACECEVENTS 111
TRACECICSEVENTS 111
TRACECURSOR 111

TRACECURSORADDRESS 111
TRACEDATA 111
TRACEDETACHEVENTS 111
TRACEDISABLEEVENTS 111
TRACEENABLEEVENTS 111
TRACEEXCEPTIONEVENTS 111
TRACEEXCIDPLEVENTS 112
TRACEEXCIEVENTS 112
TRACEFILEEVENTS 112
TRACEFULLDPLDATA 112
TRACEFULLRRSDATA 112
TRACEGLVEVENTS 112
TRACEHHLENQDEQ 112
TRACEHSMEVENTS 113
TRACEHTML 134
TRACEIBMMQEVENTS 113
TRACEIBMMQGP 113
TRACEIMSDLIEVENTS 113
TRACEIMSEVENTS 113
TRACEINTERVAL 113
TRACEITCIPAPI 113
TRACEITCIPDATA 114
TRACEITCIPEVENTS 114
TRACEITCIPGTF 114
TRACEITCIPRW 114
TRACELU62DATA 114
TRACELU62DETAIL 114
TRACELU62EVENTS 114
TRACELU62RDWR 114
TRACEMERGE 115
TRACEMERGETHROW 115
TRACEMESSAGEEVENTS 115
TRACENOEVENTS 115
TRACENTRY 115
TRACEOEDATA 115
TRACEOEEVENTS 115
TRACEOERW 116
TRACEOERWSTART 116
TRACEOTMAEVENTS 116
TRACEREMOTERPC 137
TRACERESPBUFFERS 116
TRACEREXXEXEC 116
TRACERPCEVENTS 116
TRACERRSEVENTS 117
TRACERRSXXXEVENTS 117
TRACESECURITYATTRIBS 117
TRACESECURITYDATA 117
TRACESQLEVENTS 117
TRACESQLSOURCE 117
TRACESQMEVENTS 118
TRACESRPFUNCTION 118
TRACESSLACCEPT 118
TRACESSLACCEPTSTATES 118
TRACESSLCLOSE 118
TRACESSLEVENTS 118
TRACESSLFILEBIO 118
TRACESSLREAD 119
TRACESSLTCPIPBIO 119
TRACESSLVERSION 119
TRACESSLWRITE 119
TRACESSSEVENTS 119
TRACESTACK 119
TRACESTATICSQL 119
TRACESTORAGEEVENTS 119
TRACESTREVENTS 119
TRACESZE 18
TRACETCPIPDATA 120
TRACETCPIPEVENTS 120
TRACETCPIPEXTINT 120
TRACETCPIPRDWR 120
TRACETEXTEVENTS 120
TRACETODEVENTS 121
TRACETSOEVENTS 121
TRACEURL 134
TRACEURLPARSE 135
TRACEURLREAD 121
TRACEWEBAPIEVENTS 121
TRACEWLMCALLS 121
TRACEWTOMODULES 121
TRACEWWWEVENTS 121
TRACEXCFEVENTS 121
TRACEXIT 122
TRANSACTIONTIMEOUT 137
TRANSINTTIMINGS 134
TRAP 18
TSODESC 91
TSODEST 91
TSOMAXSERVERS 123
TSOMINSERVERS 123
TSOROUTE 92
TSOSRVACTIVE 123
TSOSRVALLOWRESTART 123
TSOSRVCMDCPUTIME 124
TSOSRVCMDRUNTIME 124
TSOSRVCMDSECURITY 124
TSOSRVCMDWAITTIME 124
TSOSRVDORMANTTIME 124
TSOSRVMAXLINES 124
TSOSRVMAXQUEUE 125
TSOSRVMSGID 125
TSOSRVPROCNAME 125
TSOSRVQUEUEADD 125
TSOSRVQUEUEADDRESS 125
TSOSRVSERVERANCHOR 125
TSOSRVSMFRECORDING 125
TSOSRVSTARTUPPARM 126
TSOSRVSTRIPPROMPTS 126
TSOSRVSWAPPABLE 126
TSOSRVTIMELIMIT 126
TSOSRVTRACEOPER 122
TYPEPROSET 137
TYPINDEX 92

## U

UDPTIMEOUT 26
UNCENSORZOOMONLY 88
UPCASEMESSAGES 100

URLRESOURCETYPE 88
USECANCELTHREAD 66
USECMCO 137
USECMOF 138
USERABENDKILL 66

## V

VTAMEXITS 27

## W

WAITINTERVAL 66
WARNINGCPUTIME 66
WARNINGWAITTIME 67
WLMCLASSPLAN 129
WLMCLASSSPM 129
WLMCLASSTRAN 129
WLMCLASSUSER 129
WLMCONNECT 130
WLMSUBSYSNAM 130
WLMSUBSYSPARM 130
WLMSUBSYSTEM 130
WWWDEFAULTAUTHREQ 88
WWWDEFAULTRUNAUTH 88
WWWEPROSET 138
WWWINDEX 92
WWWRUNAUTHFORMATS 88
WWWRUNAUTHLOCATION 88
WWWSEFLAG01 135

# *Reader's Comment Form*

At NEON Systems, Inc. we are always looking for good ideas. If you have a suggestion or comment regarding any of our publications, please complete this form, and mail or fax it to us at the following address. Thank you.

Please complete the following information, or attach your business card here.

**Your Name:** _____

**Phone Number:** _____

**Your Company:** _____

**Address:** _____

_____

**Publication Name:** _____

**Version Number *and* Date** (see page ii)**:** _____

**Suggestion/Request:** _____

_____

_____

_____

_____

_____

_____

Please mail or fax this page to:

**NEON Systems, Inc.**
**14100 SW Freeway, Suite 500**
**Sugar Land, Texas 77478, U. S. A.**

Fax Number: (**281) 242-3880**