z/OS V2R3 Communications Server

zERT Aggregation recording interval Documentation changes for APAR PH25049

Contents

Chapter 1. New Function Summary

zERT Aggregation recording interval

z/OS V2R3 Communications Server with APAR PH25049 provides a zERT Aggregation recording interval that is not bound to the system SMF interval. This interval allows zERT summary records to be generated at an interval that can range from 1 to 24 hours.

Note : With APAR PH24543, you can configure this function in the Network Configuration Assistant (NCA).

zERT summary records can be collected as SMF type 119, subtype 12 records in the System Management Facility data sets or log streams. zERT summary records can also be collected by a real-time NMI application using the SYSTCPES service.

Warning :

Decreasing the frequency at which zERT summary records are written can increase the amount of 64-bit pageable, private memory needed. This is because zERT aggregation information is held longer in memory before being captured in SMF records.

Using the zERT Aggregation recording interval

To use the zERT Aggregation recording interval, perform the tasks in Table 1 on page 1.

Table 1. zERT Aggregation recording interval

Task/Procedure	Reference
Determine if you want to use the zERT aggregation recording interval.	Enabling a longer zERT aggregation recording interval in z/OS Communications Server: IP Configuration Guide
Enable the zERT aggregation recording interval using the new INTVAL and SYNCVAL sub-parameters on the GLOBALCONFIG ZERT AGGREGATION statement.	GLOBALCONFIG statement in z/OS Communications Server: IP Configuration Reference
Display zERT aggregation recording interval settings.	Netstat CONFIG/-f report in z/OS Communications Server: IP System Administrator's Commands

To find all new and updated topics about zERT Aggregation recording interval, see Table 2 on page 1.

Table 2. All related topics about zERT Aggregation recording interval

Book name	Topics
z/OS Communications Server: IP Configuration Guide	• Enabling a longer zERT aggregation recording interval • z/OS® Encryption Readiness Technology (zERT) Concepts • How does zERT aggregation provide the information? • Enabling zERT aggregation • Setting up TCP/IP operating characteristics in PROFILE.TCPIP
z/OS Communications Server: IP Configuration Reference	• GLOBALCONFIG statement

Table 2. All related topics about zERT Aggregation recording interval (continued)	
Book name	**Topics**
z/OS Communications Server: IP System Administrator's Commands	• D TCPIP,,STOR command • Netstat CONFIG/-f report
z/OS Communications Server: IP Programmer's Guide and Reference	• Format and details for poll-type requests • TCP/IP statistics record (subtype 5) • zERT Summary record (subtype 12) • TCP/IP profile record Global configuration section
z/OS Communications Server: IP Messages Volume 4 (EZZ, SNM)	• EZZ8455I

z/OS Encryption Readiness Technology (zERT) aggregation

z/OS V2R3 Communications Server, introduced a new function called z/OS Encryption Readiness Technology (zERT). With zERT, the TCP/IP stack acts as a focal point in collecting and reporting the cryptographic security attributes of IPv4 and IPv6 application traffic that is protected using the TLS/SSL, SSH, and IPSec cryptographic network security protocols. The collected connection level data is written to SMF in SMF 119 subtype 11 records.

In certain environments, the volume of SMF 119 subtype 11 records can be large. z/OS V2R3 Communications Server, with APAR PI83362, provides the zERT aggregation function. The zERT aggregation function provides an alternative SMF view of the collected security session data. This alternate view is written in the form of new SMF 119 subtype 12 records that summarize the use of security sessions by many application connections over time and which are written at the end of each SMF/INTVAL interval. This alternate view condenses the volume of SMF record data while still providing all the critical security information.

Decreasing the frequency at which zERT summary records are written may increase the amount of 64-bit pageable, private memory needed, because the zERT aggregation information is held longer in memory before being written out to SMF.

Restrictions :

No restrictions beyond those described for the zERT Discovery function that was initially provided with z/OS V2R3 Communications Server. The interval at which the SMF 119 subtype 12 records are created will be determined by the ZERT AGGregation sub-parameter INTVAL. (INTVAL/SYNCVAL sub-parameters are available in z/OS V2R3 Communications Server with APAR PH25049.)

Note : With Network Configuration Assistant (NCA) APAR PI94208, this function is available in the TCP/IP security resources, and in the SMF and real time Network Management services.

Table 3. zERT aggregation	
Task/Procedure	**Reference**
Plan for collection and storage of zERT summary SMF records and decide whether or not you want to discontinue collection of zERT connection detail records.	• Monitoring cryptographic network protection: z/OS encryption readiness technology (zERT) in z/OS Communications Server: IP Configuration Guide • z/OS MVS System Management Facilities (SMF)
Enable the zERT aggregation function.	GLOBALCONFIG statement in z/OS Communications Server: IP Configuration Reference

Table 3. zERT aggregation (continued)	
Task/Procedure	**Reference**
If you want zERT summary records to be available in the System Management Facility data sets or log streams, specify SMFCONFIG TYPE119 ZERTSUMMARY.	• SMFCONFIG statement in z/OS Communications Server: IP Configuration Reference
If you want zERT summary records to be available to a real-time NMI application: • Perform the necessary RACF® processing to authorize the NMI application to use the zERT Summary SMF NMI service (SYSTCPES). • Specify NETMONITOR ZERTSUMMARY in the TCP/IP profile.	• Requests sent by the client to the server: SYSTCPES service in z/OS Communications Server: IP Programmer's Guide and Reference • NETMONITOR statement in z/OS Communications Server: IP Configuration Reference
Display zERT aggregation configuration settings.	Netstat CONFIG/-f report in z/OS Communications Server: IP System Administrator's Commands
Enable the zERT aggregation INTVAL and SYNCVAL.	GLOBALCONFIG statement in z/OS Communications Server: IP Configuration Reference
Display zERT aggregation INTVAL and SYNCVAL configuration settings.	Netstat CONFIG/-f report in z/OS Communications Server: IP System Administrator's Commands

To find all related topics about zERT aggregation, see Table 4 on page 3.

Table 4. All related topics about zERT aggregation	
Book name	**Topics**
z/OS Communications Server: IP Configuration Guide	• Monitoring cryptographic network protection: z/OS encryption readiness technology (zERT) • What are the limitations for zERT discovery? • What does zERT aggregation collect? • How does zERT aggregation summarize the information? • How does zERT aggregation provide the information? • How does zERT aggregation determine the server port? • Using z/OS Encryption Readiness Technology (zERT) • Enabling zERT discovery • Enabling zERT aggregation • Enabling a longer zERT aggregation recording interval • Selecting a destination for zERT discovery SMF records • Selecting a destination for zERT aggregation SMF records • Disabling zERT discovery • Disabling zERT aggregation

Table 4. All related topics about zERT aggregation (continued)	
Book name	**Topics**
z/OS Communications Server: IP Configuration Reference	• GLOBALCONFIG statement • SMFCONFIG statement • NETMONITOR statement
z/OS Communications Server: IP System Administrator's Commands	• Netstat CONFIG/-f report
z/OS Communications Server: IP Programmer's Guide and Reference	• Real-time TCP/IP network monitoring NMI • Connecting to the AF_UNIX stream socket • Authorizing the applications • Real-time NMI: Connecting to the server • Real-time NMI: Interacting with the servers • Real-time NMI: Common record header • Real-time NMI: Requests sent by the client to the server • Requests sent by the client to the server: SYSTCPES service • Records sent by the server to the client: Initialization record • Records sent by the server to the client: Token record • EZBTMIC1 or EZBTMIC4 parameters • Processing the cte records for SYSTCPER • Processing the cte records for SYSTCPES • SMF type 119 records • TCP/IP profile record Global configuration section • TCP/IP profile record management section • zERT Summary record (subtype 12)

IBM zERT Network Analyzer

z/OS Management Facility (z/OSMF) V2R3 with APAR PH03137, provides a new plug-in named IBM® zERT Network Analyzer. IBM zERT Network Analyzer is a web-based graphical user interface that z/OS network security administrators can use to analyze and report on data reported in zERT Summary records.

z/OS V2R3 Communications Server introduced a new feature called z/OS Encryption Readiness Technology (zERT). zERT positions the TCP/IP stack to act as a focal point for collecting and reporting the cryptographic security attributes of IPv4 and IPv6 TCP and Enterprise Extender (EE) connection traffic that is protected using the TLS/SSL, SSH and IPSec cryptographic network security protocols. Connection data is written to z/OS System Management Facility (SMF) in two new SMF type 119 records:

- zERT Connection Detail (subtype 11) records are written on a per-connection basis to record the cryptographic protection history of a given TCP or EE connection.
- zERT Summary (subtype 12) records are written on a per-security-session basis at the end of each SMF interval to summarize the repeated use of security sessions during the interval.

z/OS Management Facility (z/OSMF) V2R3 is enhanced by APAR PH03137 to provide a new plug-in named IBM zERT Network Analyzer. IBM zERT Network Analyzer is a web-based graphical user interface that

z/OS network security administrators can use to analyze and report on data reported in zERT Summary records.

To get a quick start with IBM zERT Network Analyzer, see IBM zERT Network Analyzer tutorial.

Dependency :

- You must have installed z/OSMF V2R3 APARs PH04391 and PH00712 to use IBM zERT Network Analyzer.
- The IBM zERT Network Analyzer task requires Db2® 11 for z/OS and above.

Table 5. IBM zERT Network Analyzer

Task/Procedure	Reference
Enable collection of zERT Summary (SMF Type 119 subtype 12) SMF records • Enable zERT Aggregation function by specifying the GLOBALCONFIG ZERT AGGREGATION statement. • Enable a longer interval at which the SMF 119 subtype 12 (zERT summary) records are created by using the INTVAL sub-parameter of the ZERT AGGregation statement. • Display zERT aggregation INTVAL or SYNCVAL configuration settings. • Direct zERT aggregation to write the zERT Summary SMF records to the System Management Facility (SMF) by specifying the SMFCONFIG TYPE119 ZERTSUMMARY statement. • Enable the recording of type 119 records, and optionally define the SMF interval duration, in your SMF parmlib member.	• z/OS Communications Server: IP Configuration Guide • GLOBALCONFIG statement in z/OS Communications Server: IP Configuration Reference • SMFCONFIG statement in z/OS Communications Server: IP Configuration Reference • z/OS MVS System Management Facilities (SMF)
Dump the collected zERT Summary records to a sequential data set using the IFASMFDP or IFASMFDL program • Use IFASMFDP for SMF data sets • Use IFASMFDL for SMF log streams	z/OS MVS System Management Facilities (SMF)
Enable the IBM zERT Network Analyzer plug-in in z/OSMF by adding ZERT_ANALYZER to the PLUGINS statement.	IZUPRMxx reference information in IBM z/OS Management Facility Configuration Guide
Authorize the user IDs that will be using IBM zERT Network Analyzer	Updating z/OS for the IBM zERT Network Analyzer plug-in in IBM z/OS Management Facility Configuration Guide
Create the proper Db2 for z/OS database definitions to use with IBM zERT Network Analyzer	Updating z/OS for the IBM zERT Network Analyzer plug-in in IBM z/OS Management Facility Configuration Guide
Start the z/OSMF IBM zERT Network Analyzer plug-in	• When using the z/OSMF traditional view, expand the **Analysis** category in the navigation area, and select **IBM zERT Network Analyzer**. • When using the z/OSMF desktop view, click the **IBM zERT Network Analyzer** icon.

Table 5. IBM zERT Network Analyzer (continued)	
Task/Procedure	**Reference**
Import the dumped zERT SMF Summary records into IBM zERT Network Analyzer	IBM zERT Network Analyzer online help, *Analysis* category under the IBM z/OS Management Facility online help
Analyze the imported zERT Summary data using IBM zERT Network Analyzer query and reporting functions	IBM zERT Network Analyzer online help, *Analysis* category under the IBM z/OS Management Facility online help

To find all related topics about IBM zERT Network Analyzer, see Table 6 on page 6.

Table 6. All related topics about IBM zERT Network Analyzer	
Book name	**Topics**
z/OS Communications Server: IP Configuration Guide	• z/OS Encryption Readiness Technology (zERT) Concepts • Selecting a destination for zERT aggregation SMF records • Using IBM zERT Network Analyzer
z/OS Communications Server: IP Programmer's Guide and Reference	• zERT connection detail record (subtype 11) • zERT Summary record (subtype 12)
IBM z/OS Management Facility Configuration Guide	• IZUPRMxx reference information • Selecting which z/OSMF plug-ins to add • IBM zERT Network Analyzer task overview • Updating z/OS for the IBM zERT Network Analyzer Plug-in • Problems when using IBM zERT Network Analyzer • Steps for sending information to IBM Support • Resource authorizations for the IBM zERT Network Analyzer plug-in
IBM zERT Network Analyzer online help	Messages: IZUETXXXXX

Chapter 2. IP Configuration Guide

z/OS Encryption Readiness Technology (zERT) Concepts

With the increasing number of corporate, industry, and government regulations regarding cryptographic protection of data in flight, as well as discoveries of weaknesses in existing cryptographic protocols and algorithms, it is important for z/OS administrators and auditors to be able to assess the quality of the cryptographic network protection being applied to their key z/OS workloads.

The landscape of cryptographic network protection on z/OS is varied and can be complex. Because of this, performing such an assessment can be very difficult. Consider the following:

- There are two different TLS/SSL protocol implementations on z/OS:
  - z/OS Cryptographic Services System SSL, which is available to software written in C or C++.
  - Java™ Secure Sockets Extension (JSSE), which available to Java software (and is written in Java itself).

  Programs and middleware that use System SSL directly usually have their own unique configuration methods to control the details of TLS/SSL protection. Programs and middleware that use JSSE often require a set of JSSE-specific environment variable or parameters to configure their TLS/SSL protection.
- z/OS Communications Server provides AT-TLS, which invokes System SSL on behalf of applications and middleware based on policies that you create.
- z/OS Communications Server provides a full IPSec implementation, also configured through policies that you create.
- z/OS also supports the Secure Shell (SSH) protocol , although Communications Server does not directly use it. For more information of Secure Shell (SSH) protocol, see *z/OS OpenSSH* in *z/OS Introduction and Release Guide*.
- Each of these protocols allow the local and remote endpoints to negotiate the exact protection methods to be used for any given security session. This means understanding the z/OS configuration for a given application's cryptographic protection only tells you which attributes (protocol versions, algorithms, key lengths, and so on.) are possible - it usually does not tell you the exact set of protection attributes that were agreed upon for any specific security session that is established between the z/OS application and a remote endpoint.

With such variety of protocols, configuration methods, and audit and log records, it can be difficult to clearly understand the overall state of cryptographic network protection for your z/OS system.

zERT positions the z/OS TCP/IP stack as a central collection and reporting point for the cryptographic protection attributes for TLS, SSL, SSH, and IPSec security sessions that are protecting TCP and Enterprise Extender connections that terminate on the local stack. This collection and reporting function is called zERT discovery.

A second reporting function called zERT aggregation summarizes the repeated use of security sessions over a period of time (the system's SMF interval or the specified zERT aggregation INTVAL setting). By focusing more on the security session than on individual TCP or EE connections, zERT aggregation provides the same level of cryptographic detail as zERT discovery, but with significantly fewer SMF records. The zERT aggregation function requires the use of zERT discovery to collect the individual TCP and EE connection information that is then summarized by zERT aggregation.

IBM zERT Network Analyzer is a z/OS Management Facility (z/OSMF) task that provides a GUI-based tool for analyzing cryptographic protection characteristics of TCP and Enterprise Extender (EE) connections on your system, using SMF records generated by the zERT aggregation function.

How does zERT aggregation provide the information?

You can direct zERT aggregation to write the summarized records to the z/OS System Management Facility, to a network management application using the real-time NMI for zERT summary information, or to both. Regardless of the destination, the information is recorded using SMF 119 subtype 12 (zERT summary) records.

zERT summary records have the following general format:

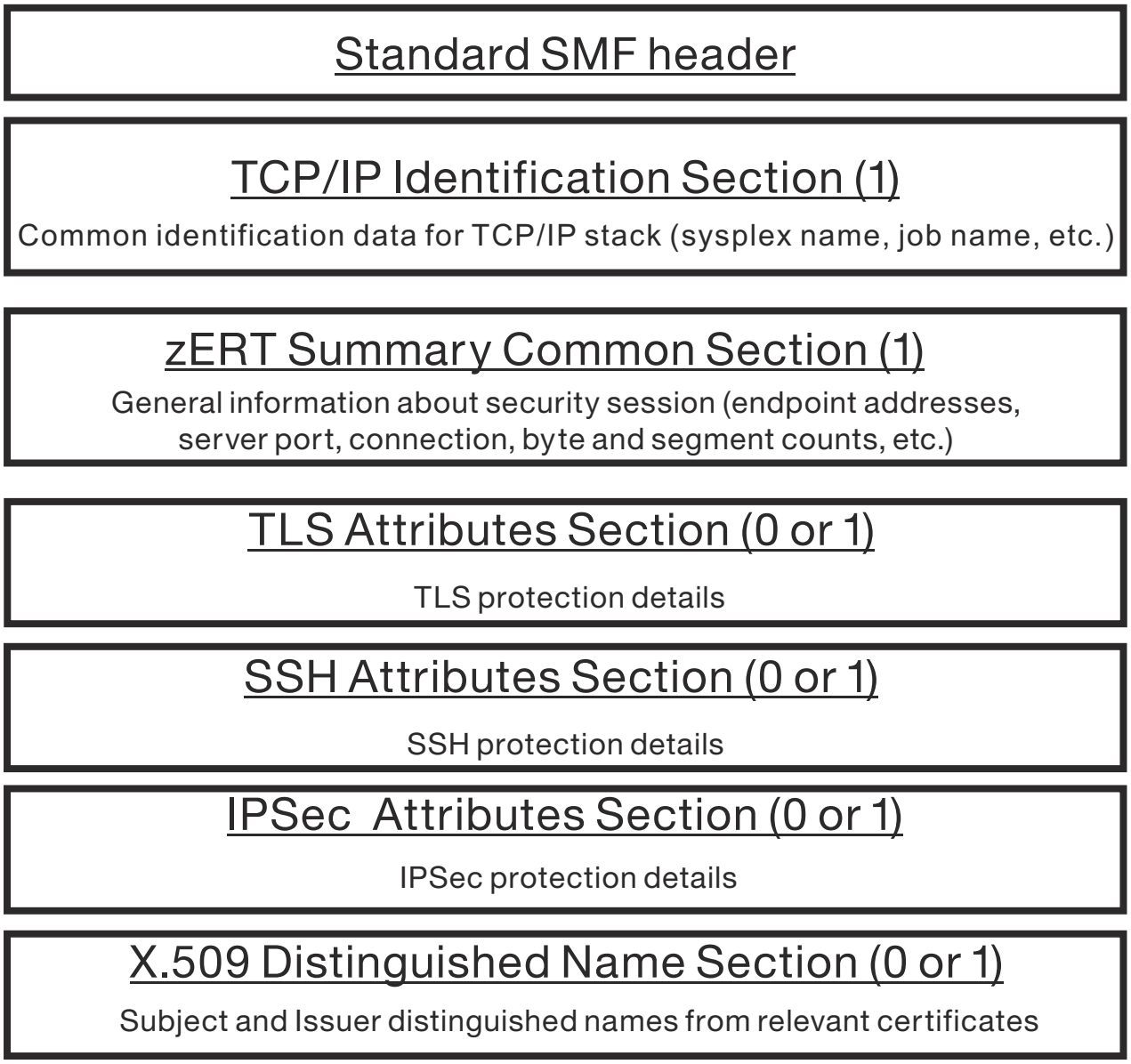


Figure 1. zERT summary (SMF 119 subtype 12) record layout

One zERT summary record is generated at the end of each SMF/INTVAL interval for each unique security session that was used during that interval. Each zERT summary record contains information about the security session in the zERT Identification Section. This includes the server and client security session endpoints, the server port or port range, attributes of the owner of the local socket such as userid and jobname, and statistical information about the usage of the security session. This statistical information includes the following information for the security session:

- The total number of TCP or EE connections that were ever covered by this session.
- The number of active TCP or EE connections that are covered by this session at the time of reporting.

- The number of partial TCP or EE connections that were ever covered by this session. A *partial TCP connection* is one that changed significant security attributes at some point, for instance changed from having no recognized cryptographic protection to having a recognized cryptographic protection.
- The number of short lived TCP connections that were covered by this session. EE connections are never considered to be short lived connections.
- The number of inbound and outbound data bytes and segments processed by the TCP or EE connections when they were covered by this security session.

The record contains both the values for these statistics at the end of the current reporting period as well as the values at the start of the current reporting period. You can subtract the ending values from the starting values to determine the activity for the SMF interval.

Security sessions that provide no recognized cryptographic protection, meaning they represent connections flowing as clear text or are protected in a way that zERT discovery does not recognize, will only contain the TCP/IP Identification and zERT Identification sections. For security sessions that represent TLS, SSL, SSH or IPSec attributes, the record contains a protocol-specific section that records the significant cryptographic attributes of the security session.

Guideline : A zERT summary SMF record will have at most one protocol-specific section since it represents a single security session rather than individual connections.

If X.509 Distinguished Names are used by the security session, the record also includes that section.

Subtype 12 records are also written in two special cases: when zERT aggregation is enabled or disabled through the GLOBALCONFIG parameter.

For more information on the SMF 119 zERT summary (subtype 12) records, see zERT Summary record (subtype 12) in z/OS Communications Server: IP Programmer's Guide and Reference.

Enabling zERT aggregation

You turn the zERT aggregation function on in the TCP/IP stack by specifying the GLOBALCONFIG ZERT AGGREGATION parameter in the TCP/IP profile data set. Enabling zERT aggregation causes the TCP/IP stack to collect cryptographic protection summary information for security sessions involving TCP and Enterprise Extender connections that terminate at the stack.

Guideline : Because zERT aggregation summarizes information collected by zERT discovery processing, zERT discovery must be enabled for zERT aggregation to operate; however, you do not have to enable SMF recording of zERT connection detail records for zERT aggregation to function correctly.

Results : If you dynamically enable zERT aggregation using VARY OBEYFILE, zERT does not collect summary information about TCP and Enterprise Extender connections that were established before enabling zERT aggregation.

Enabling a longer zERT aggregation recording interval

By default, zERT aggregation writes the data it has collected in the form of SMF 119 subtype 12 (zERT summary) records on each SMF interval. If you want to reduce the frequency at which these records are written, you can specify a zERT aggregation-specific recording interval between 1 and 24 hours (in one hour increments) by specifying the INTVAL sub-parameter of the GLOBALCONFIG ZERT AGGregation parameter in the TCP/IP profile data set.

If you specify an INTVAL value, it is recommended that the value divides evenly into 24 hours to ensure the SMF 119-12 records are written at the same times each day.

To specify exactly when the INTVAL intervals are to begin, you can specify the SYNCVAL sub-parameter of INTVAL. SYNCVAL defines the reference time from which the INTVAL intervals begin to be applied. SYNCVAL is specified as a time of day in 24 hour clock format (hh:mm). For example, if SYNCVAL is specified as 06:00 (6 AM) and INTVAL is set to 12 hours, then zERT aggregation will write its SMF records at 6 AM and 6 PM daily. If not specified, the default SYNCVAL is midnight (00:00).

For more information on enabling zERT aggregation, see GLOBALCONFIG statement in z/OS Communications Server: IP Configuration Reference.

Decreasing the frequency at which zERT summary records are written may increase the amount of 64-bit pageable, private memory needed, because the zERT aggregation information is held longer in memory before being written out to SMF. Any increase in memory usage will be dependent on the nature of network connections as well as the length of the recording interval. Environments where the majority of the connection patterns are consistent throughout a 24 hour period will require less memory than those where connection patterns vary significantly throughout the day. A longer recording interval increases the likelihood of greater memory consumption. In environments where 64-bit private storage is limited, consider configuring a smaller zERT aggregation recording interval in order to allow for more frequent flushing of zERT Aggregation information.

For more information on displaying zERT aggregation storage information, see DISPLAY TCPIP,,STOR in z/OS Communications Server: IP System Administrator's Commands.

Setting up TCP/IP operating characteristics in PROFILE.TCPIP

Figure 2 on page 12 shows a portion of the sample configuration file for the TCP/IP address space, PROFILE.TCPIP. This sample can be copied from SEZAINST(SAMPPROF). Figure 2 on page 12 includes the portion of the sample that shows how to set up TCP/IP operating characteristics. Descriptions for the statements follow Figure 2 on page 12.

```
; ====================================================================
; ====================================================================
; General TCP/IP address space configuration
; ====================================================================
;
; ARPAGE: Specifies the number of minutes between creation or
;   revalidation of an LCS ARP table entry and the deletion of the
;   entry.
;
ARPAGE 20
;
; --------------------------------------------------------------------
;
; GLOBALCONFIG: Provides settings for the entire TCP/IP stack
;
; Example GLOBALCONFIG to offload TCP segmentation to OSA-Express
; features
;
; GLOBALCONFIG SEGMENTATIONOFFLOAD
;
; Example GLOBALCONFIG to exploit HiperSockets multiple write
; support
;
; GLOBALCONFIG IQDMULTIWRITE
;
; Example GLOBALCONFIG to displace TCP/IP CPU cycles onto a zIIP
; for certain workloads
;
; GLOBALCONFIG ZIIP IPSECURITY IQDIOMULTIWRITE
;
; Example GLOBALCONFIG to assign OSA-Express QDIO write priority
; values to packets associated with WorkLoad Manager service classes,
; and to forwarded packets
;
; GLOBALCONFIG WLMPRIORITYQ
;              IOPRI1 0
;              IOPRI2 1
;              IOPRI3 2 3
;              IOPRI4 4 5 6 FWD
;
; Example GLOBALCONFIG to enable SMC-R and SMC-D processing
;
; GLOBALCONFIG SMCD FIXEDMEMORY 1000
;                            SMCR PFID 203 PFID 205 FIXEDMEMORY 1000
;
; Example GLOBALCONFIG to enable zERT processing
;
; GLOBALCONFIG ZERT AGGREGATION INTVAL 2 SYNCVAL 00:00
; --------------------------------------------------------------------
;
```

```
; IPCONFIG: Provides settings for the IPv4 IP layer of TCP/IP.
;
; Example IPCONFIG for single stack/single system:
;
IPCONFIG DATAGRAMFWD SYSPLEXROUTING
;
; Example IPCONFIG for automatic activation of inter-stack dynamic XCF
;   and Same Host (IUTSAMEH) interfaces
;
; IPCONFIG DYNAMICXCF 201.1.10.10 255.255.255.0 2
;
; Example IPCONFIG for IPSECURITY support:
;
; IPCONFIG IPSECURITY
;
; Example IPCONFIG to provide accelerated forwarding at the DLC layer
;   for OSA-Express QDIO and HiperSockets packets
;
; IPCONFIG QDIOACCELERATOR
;
; ------------------------------------------------------------------
;
; IPCONFIG6: Provides settings for the IPv6 IP layer of TCP/IP.
;
; Example IPCONFIG6 to enable IPv6 packet forwarding and the use of
;   virtual IP addresses as source addresses in outbound datagrams:
;
; IPCONFIG6 DATAGRAMFWD SOURCEVIPA
;
; Example IPCONFIG6 for automatic activation of inter-stack dynamic XCF
;   and Same Host (IUTSAMEH) interfaces
;
; IPCONFIG6 DYNAMICXCF 2001::151:0000
;
; ------------------------------------------------------------------
;
; SOMAXCONN: Specifies maximum length for the connection request queue
;   created by the socket call listen().
;
SOMAXCONN 10
;
; ------------------------------------------------------------------
;
; TCPCONFIG: Provides settings for the TCP layer of TCP/IP.
;            RESTRICTLOWPORTS limits access to ports below 1024
;            to authorized applications.  Applications can be
;            authorized to low ports in three ways:
;             - via PORT or PORTRANGE with the appropriate jobname
;                or wildcard jobname
;             - APF authorized
;             - superuser
;
TCPCONFIG TCPSENDBFRSIZE 32K TCPRCVBUFRSIZE 32K SENDGARBAGE FALSE
          RESTRICTLOWPORTS
;
; Example TCPCONFIG to change the KEEPALIVE interval for applications
; that enable the SO_KEEPALIVE socket option but do not override
; the interval using the TCP_KEEPALIVE socket option.
;
; TCPCONFIG INTERVAL 30
;
; Example TCPCONFIG for AT-TLS support:
;
; TCPCONFIG TTLS
;
; ------------------------------------------------------------------
;
; UDPCONFIG: Provides settings for the UDP layer of TCP/IP
;            RESTRICTLOWPORTS limits access to ports below 1024
;            to authorized applications.  Applications can be
;            authorized to low ports in three ways:
;             - via PORT or PORTRANGE with the appropriate jobname
;                or wildcard jobname
;             - APF authorized
;             - superuser
;
UDPCONFIG RESTRICTLOWPORTS
;
; ------------------------------------------------------------------
;
; SRCIP: Provides the following functionality:
;   - Provides for the substitution of a source IP address on a
```

```
;     jobname-specific or destination-specific basis, for applications
;     which specify either the IPv4 INADDR_ANY address, or the IPv6
;     unspecified address (in6addr_any) for the source IP address.
;     This may be done when an application issues an explicit bind()
;     call with either of these addresses, or when it bypasses issuing
;     an explicit bind() call and issues a connect().
;   - Provides the ability to designate if default source address
;     selection should prefer a public or a temporary IPv6 address
;     for the specified jobs.
;
;
; Example SRCIP to substitute a source IP address
;
;SRCIP
; JOBNAME      USER15            9.43.242.5
; JOBNAME      USER*             9.43.242.4
; JOBNAME      USER15            2001::092B:F203
; JOBNAME      JOB*              ETHER1
; DESTINATION  9.67.114.02       9.43.240.7
; DESTINATION  2003::090C:F246   INTF1
; JOBNAME      *                 9.43.242.3
; JOBNAME      *                 9.43.242.3
; JOBNAME      PAYROLL*          9.42.242.5              BOTH
; JOBNAME      SERVER1           9.42.242.4              SERVER
; JOBNAME      CLIENT*           2001:0DB8::9:43:242:6   CLIENT
;ENDSRCIP
;
; Example SRCIP to cause default source address selection to prefer
; public or temporary IPv6 addresses
;
;SRCIP
; JOBNAME      IPV6PUB            PUBLICADDRS
; JOBNAME      IPV6TEMP           TEMPADDRS
;ENDSRCIP
;
; ----------------------------------------------------------------------
;
; DEFADDRTABLE: Can be used to configure the policy table for IPv6
; default address selection.
;
;DEFADDRTABLE
; Prefix          Precedence Label
; ::1/128           50         0
; ::/0              40         1
; 2002::/16         30         2
; ::/96             20         3
; ::ffff:0.0.0.0/96 10         4
;ENDDEFADDRTABLE
```

Figure 2. Example of TCP/IP operating characteristics in PROFILE.TCPIP

The following information describes the statements that are shown in Figure 2 on page 12. For more information about any of these statements, see z/OS Communications Server: IP Configuration Reference. For information specific to IPv6 support, see z/OS Communications Server: IPv6 Network and Application Design Guide.

ARPAGE

Use ARPAGE to set the number of minutes between a revalidation and deletion of ARP table entries for LCS devices. If you want to describe this value in seconds versus minutes, use the IPCONFIG ARPTO statement.

GLOBALCONFIG

Use GLOBALCONFIG to print several counters in text format. These counters include number of TCP retransmissions and total number of TCP segments sent from the TCP/IP system. Most installations use the SMF facility of MVS™ to collect these counters in a more standard way.

Use GLOBALCONFIG to enable use of Shared Memory Communications over RMDA (SMC-R) and Shared Memory Communication - Direct Memory Access (SMC-D). For more details about SMC-R and SMC-D, see Shared Memory Communications.

Use the ECSALIMIT parameter on the GLOBALCONFIG statement to limit TCP/IP use of common storage. The POOLLIMIT parameter can be used to limit TCP/IP use of private storage pools.

Use ZERT to enable z/OS Encryption Readiness Technology (zERT). For more information, see Monitoring cryptographic network protection: z/OS encryption readiness technology (zERT).

IPCONFIG

Use IPCONFIG to configure various settings of the IP layer of TCP/IP. Use ARPTO to specify the ARP timeout value in seconds for LCS devices. For more information, see the ARPAGE description.

Use CLAWUSEDOUBLENOP on vendor devices that document the need for double NOPs on each CCW.

Use DATAGRAMFWD if this TCP/IP is to be a router and must forward datagrams to other routers. Use IGNOREREDIRECT when a dynamic routing program is used and ICMP redirect packets are to be ignored by the TCP/IP address space. MULTIPATH is used to inform TCP/IP how to distribute traffic across equal cost routes.

Use IPSECURITY to restrict this host to be a network firewall.

SOURCEVIPA enables interface fault tolerance for z/OS clients that establish outbound connections. When SOURCEVIPA is set, outbound datagrams use the corresponding virtual IP address (VIPA) in the HOME list instead of the physical interfaces IP address. SOURCEVIPA has no effect on RIP servers such as OMPROUTE.

TCPSTACKSOURCEVIPA allows z/OS clients to specify a sysplex-wide source IP address for TCP connections. When TCPSTACKSOURCEVIPA is set, outbound TCP datagrams use the IP address that is specified in the TCPSTACKSOURCEVIPA statement instead of static VIPA addresses or physical interface addresses.

Use SYSPLEXROUTING to communicate interface changes within a sysplex domain to the workload manager (WLM). DYNAMICXCF allows the cross communication facility within a sysplex to dynamically generate connections within a sysplex domain. If DYNAMICXCF is used with a dynamic routing program like OMPROUTE, the BSDROUTINGPARMS and the OMPROUTE configuration files must be updated with subnet mask and cost information. For more information about other configuration parameters that are required, see the usage notes related to the DYNAMICXCF parameter under the IPCONFIG statement in z/OS Communications Server: IP Configuration Reference.

Use REASSEMBLYTIMEOUT to specify the TCP/IP reassemble timeout value in seconds, and the TTL specifies the TCP/IP time to live or hop count value.

Use PATHMTUDISCOVERY to indicate to TCP/IP that it is to dynamically discover the path MTU, which is the minimum of MTUs of each hop in the path.

Use STOPONCLAWERROR to indicate to the TCP/IP stack to stop channel programs (HALTIO and HALTSIO) when a device error is detected.

Use QDIOACCELERATOR to request accelerated packet forwarding for OSA-Express® QDIO Ethernet and HiperSockets interfaces.

IPCONFIG6

Use IPCONFIG6 to update the IP layer of TCP/IP with information that pertains to IPv6.

Use DATAGRAMFWD to enable the transfer of data between networks.

Use DYNAMICXCF to enable Dynamic XCF support for IPv6.

SOMAXCONN

Use SOMAXCON to specify the maximum number of sockets queued on a listener.

SRCIP

Use the SRCIP - ENDSRCIP profile statement block to configure one of the following functions:

- Enable an application to use a designated IP address as its source address for outbound TCP connections, or to enable a TCP server application to bind to a specific IP address when it is establishing its listening socket.
- Indicate that the default source address selection algorithm prefers public or temporary IPv6 addresses for specific jobs.

For outbound TCP connections, when a source IP address was designated for a specified job name or destination address and the source IP address exists at the time the outbound TCP connection is initiated, this source IP address is used, overriding other source IP address selection methods as described in Source IP address selection. This source address selection occurs for applications that issue a connect() call and that did not previously bind the socket to an IP address, or for those applications that bind to the IPv4 INADDR_ANY address or to the IPv6 unspecified address (in6addr_any) before they issue the connect() call.

For TCP server applications, when the application issues a bind to INADDR_ANY or in6addr_any and a matching JOBNAME rule for SERVER or BOTH is specified, the designated IP address is used on the listening socket. This situation makes the server application bind specific, where client applications can connect to the server by using only the designated IP address. This capability can be useful when the applications do not provide a method for the user to specify a specific IP address for their listening sockets, or in situations when the server application creates listening sockets by using an ephemeral port that is assigned dynamically by TCP/IP. For scenarios when the application binds to specific, well-known ports, the BIND keyword on the PORT reservation statement in the TCP/IP profile can be used instead and has precedence over the SRCIP block specifications.

If you use distributed DVIPAs as a designated source within the SRCIP block, you might also be required to specify the EXPLICITBINDPORTRANGE parameter on the GLOBALCONFIG statement. For more information about the GLOBALCONFIG statement and its parameters, see z/OS Communications Server: IP Configuration Reference.

Guidelines :

- Applications that bind to INADDR_ANY or in6addr_any that match on an SRCIP JOBNAME or DESTINATION statement do not have the designated IP address as their source address upon completion of the bind() call. The source address is not set to the designated address until completion of the subsequent connect() (client applications) or listen() (server applications) call. This situation is important to note for applications that issue a getsockname() call after a bind() call to retrieve the source IP address. This processing is different from the processing that occurs when a TCP server application is converted to being bind specific using the BIND keyword on the PORT statements in the TCP/IP profile. When you are using the BIND keyword on the PORT statement, the designated IP address is set upon completion of the bind() call, and some applications such as Db2 depend on this behavior.
- When you are using an SRCIP JOBNAME statement for an IPv6 server application, code an IPv6 address and not an IPv6 interface. Otherwise, the source address that is chosen for that IP interface might not be the best choice for the server application to be bound to. For information about the default source address selection algorithm, see z/OS Communications Server: IPv6 Network and Application Design Guide.

TCPCONFIG

Use the TCPCONFIG statement to configure various settings of the TCP protocol layer:

- Use the INTERVAL parameter if necessary to change the default keepalive value to a value other than 120 minutes. Use the KEEPALIVEPROBES parameter to specify the number of probes to send before a connection times out. Use the KEEPALIVEPROBEINTERVAL parameter to specify the amount of time between the sending of each probe.
- Use the FINWAIT2TIME parameter to specify a different timeout value for a TCP connection that is in the FINWAIT2 state.
- Use the TIMEWAITINTERVAL parameter to specify a different timeout value for a TCP connection that is in the TIMEWAIT state.
- Use the SENDGARBAGE parameter to cause the keepalive packet to contain 1 byte of random data and an incorrect sequence number. The random data and incorrect sequence number assure that the remote TCP does not accept the data.
- Use the TCPTIMESTAMP parameter to choose whether to participate in time stamp negotiation.
- Use the MAXIMUMRETRANSMITTIME parameter to limit the length of time before a connection times out.

- Use the RETRANSMITATTEMPTS parameter to indicate the number of packets to retransmit before a connection times out.
- Use the CONNECTTIMEOUT parameter to limit the amount of time before the initial connection times out.
- Use the CONNECTINITINTERVAL parameter to specify the initial retransmit interval for a connect call.
- Use the QUEUEDRTT parameter to specify the round trip time that triggers outbound serialization logic.
- Use the FRRTHRESHOLD parameter to specify the number of duplicates that are needed to trigger Fast Retransmit, Fast Recovery logic.
- Use the DELAYACKS parameter to alter the behavior of acknowledgments and delay their transmission.
- Use the NONAGLE parameter to override use of the Nagle algorithm. The Nagle algorithm is used to delay small packets from being sent.
- If you specify the RESTRICTLOWPORTS parameter, only applications that meet at least one of the following criteria are allowed to bind to low ports (1–1023):
  - The port is reserved for the application by the PORT or PORTRANGE statement.
  - The application runs with APF authorization.
  - The application runs with effective POSIX UID zero.
- If you want to control TCP buffering to limit storage usage or to manage large bandwidth devices, use the TCPSENDBFRSIZE, TCPRCVBUFRSIZE, TCPMAXSENDBUFRSIZE, and TCPMAXRCVBUFRSIZE parameters.
- Use the TTLS parameter to configure the TCP/IP stack for AT-TLS support.
- Use the EPHEMERALPORTS parameter to limit the ephemeral port range that the TCP/IP stack uses to assign a port to a socket. The EPHEMERALPORTS port range is used in the following situations when EXPLICITBINDPORTRANGE, SYSPLEXPORTS, or FTP PASSIVEDATAPORTS processing cannot determine the port number to assign:
  - An application issues an explicit bind() call for port 0
  - An application bypasses issuing an explicit bind() call and issues a connect() call
- The SELECTIVEACK parameter causes the TCP/IP stack to generate selective acknowledgments as defined in RFC 2018 and to use incoming selective acknowledgments to improve TCP retransmission processing as defined in RFC 3517. A TCP connection can experience poor performance when multiple packets are lost from one window of data. With the limited information available from cumulative acknowledgments, a TCP sender can learn about only a single lost packet per round-trip time. A Selective Acknowledgment (SACK) mechanism with a selective repeat retransmission policy can help to overcome these limitations. The receiving TCP sends back SACK packets to the sender to inform the sender of data that was received. The sending TCP can then retransmit only the missing data segments.

UDPCONFIG

Use UDPCONFIG to configure various settings of the UDP protocol layer. NOUDPCHKSUM can be used to eliminate check summing overhead for IPv4 UDP packets. This option is ignored for UDP datagrams that are flowing over an IPv6 network, as UDP Checksum is a required function on an IPv6 network.

If RESTRICTLOWPORTS is specified, only applications that meet at least one of the following criteria are allowed to bind to low ports (1–1023):

- The port is reserved for the application by the PORT or PORTRANGE statement.
- The application runs with APF authorization.
- The application runs with effective POSIX UID zero.

If an installation wants to control UDP buffering (to limit storage usage or to manage large bandwidth devices), use the UDPSENDBFRSIZE and UDPRCVBUFRSIZE parameters. UDPQUEUELIMIT can be used to set a queue limit for UDP. UDPQUEUELIMIT is useful for installations that want to limit the size of the queue of UDP datagrams that an application can have waiting before the TCP/IP address space starts discarding them.

Use EPHEMERALPORTS to limit the ephemeral port range that the TCP/IP stack uses to assign a port to a socket in the following situations:

- An application issues an explicit bind() call for port 0
- An application bypasses issuing an explicit bind() call

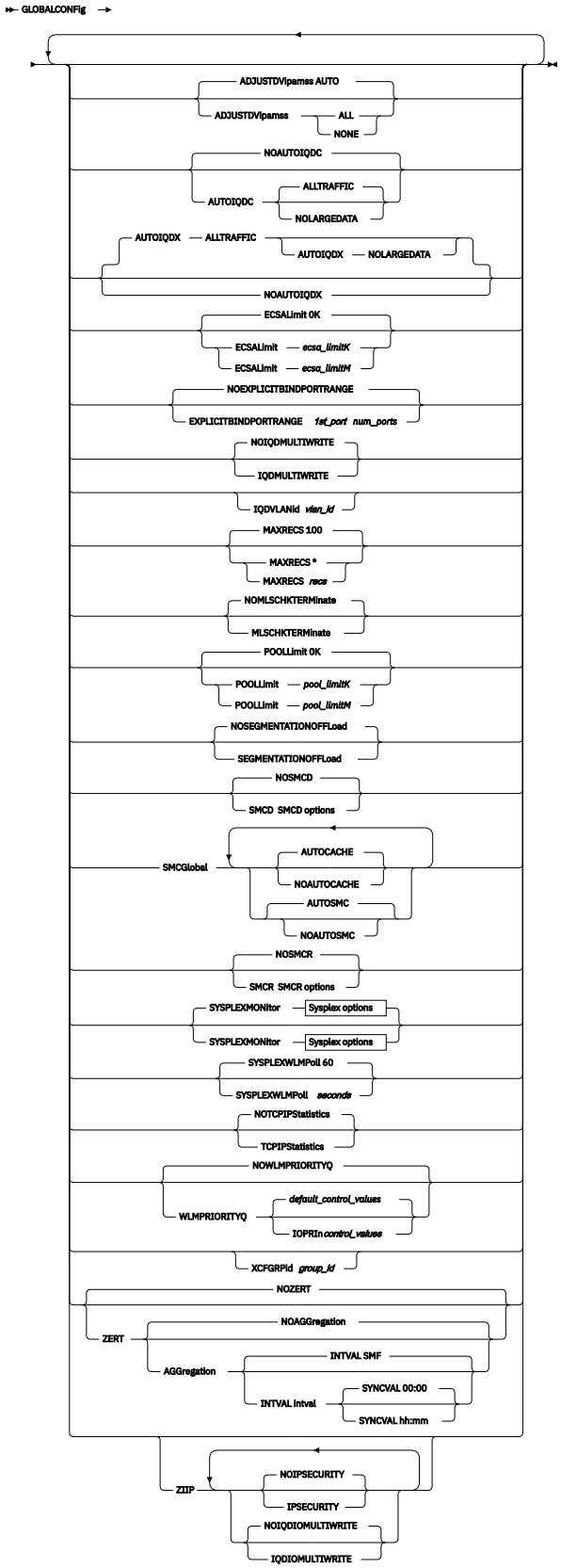
Chapter 3. IP Configuration Reference

GLOBALCONFIG statement

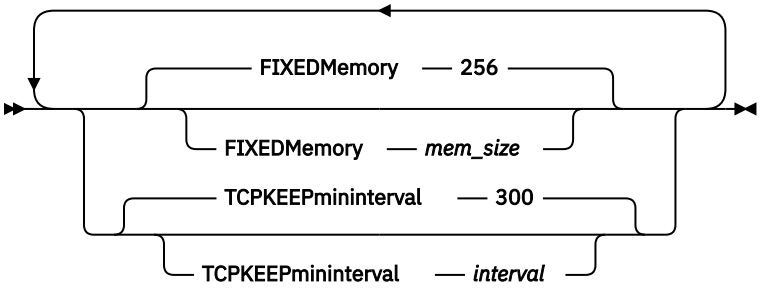
Use the GLOBALCONFIG statement to pass global configuration parameters to TCP/IP.

Syntax

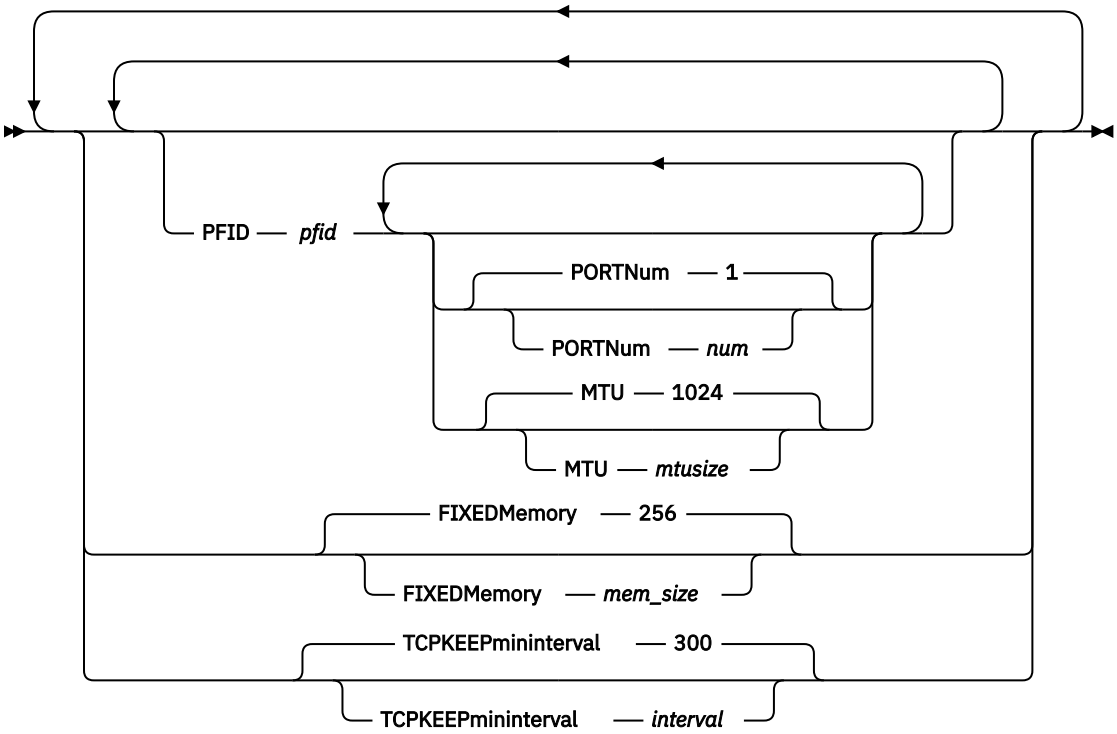
Tip: Specify the parameters for this statement in any order.


GLOBALCONFig
ADJUSTDVipamss AUTO
ADJUSTDVipamss
ALL
NONE
NOAUTOIQDC
AUTOIQDC
ALLTRAFFIC
NOLARGEDATA
AUTOIQDX
ALLTRAFFIC
AUTOIQDX
NOLARGEDATA
NOAUTOIQDX
ECSALimit 0K
ECSALimit
ecsa_limitK
ECSALimit
ecsa_limitM
NOEXPLICITBINDPORTRANGE
EXPLICITBINDPORTRANGE
1st_port num_ports
NOIQDMULTIWRITE
IQDMULTIWRITE
IQDVLANId
vlan_id
MAXRECS 100
MAXRECS *
MAXRECS
recs
NOMLSCHKTERMinate
MLSCHKTERMinate
POOLLimit 0K
POOLLimit
pool_limitK
POOLLimit
pool_limitM
NOSEGMENTATIONOFFLoad
SEGMENTATIONOFFLoad
NOSMCD
SMCD SMCD options
SMCGlobal
AUTOCACHE
NOAUTOCACHE
AUTOSMC
NOAUTOSMC
NOSMCR
SMCR SMCR options
SYSPLEXMONitor
Sysplex options
SYSPLEXMONitor
Sysplex options
SYSPLEXWLMPoll 60
SYSPLEXWLMPoll
seconds
NOTCPIPStatistics
TCPIPStatistics
NOWLMPRIORITYQ
WLMPRIORITYQ
default_control_values
IOPRIn control_values
XCFGRPId
group_id
NOZERT
ZERT
NOAGGregation
AGGregation
INTVAL SMF
INTVAL intval
SYNCVAL 00:00
SYNCVAL hh:mm
ZIIP
NOIPSECURITY
IPSECURITY
NOIQDIOMULTIWRITE
IQDIOMULTIWRITE

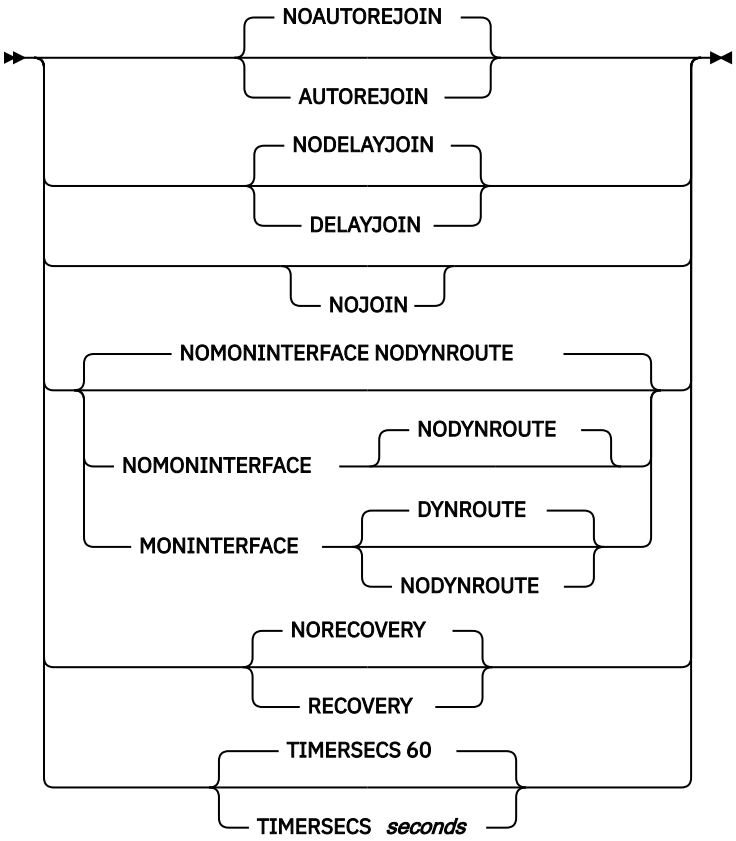
SMCD options



SMCR options



Sysplex options



Parameters

ADJUSTDVIPAMSS AUTO | ALL | NONE
Specifies subparameters to control whether TCP/IP changes the Maximum Segment Size (MSS) that is advertised for a TCP connection. Connections that use VIPAROUTE to forward Sysplex Distributor packets add a Generic Routing Encapsulation (GRE) header to the packet. The addition of a GRE header increases the packet size and can cause IP fragmentation between the distributor and the target stack. To avoid this fragmentation, the length of the GRE header can be subtracted from the MSS that TCP connections advertise at connection establishment. Changes to ADJUSTDVIPAMSS affect only the new connections.

AUTO
Indicates that TCP/IP automatically adjusts the MSS to accommodate the length of a GRE header. For inbound connections on a target stack, the MSS is adjusted if the destination address is a distributed DVIPA and VIPAROUTE is being used. For outbound connections on a target stack, the MSS is adjusted if the source IP address is a distributed DVIPA. This is the default value.

ALL
Indicates that TCP/IP adjusts the MSS for connections that use a DVIPA as the local IP address, whether the DVIPA is distributed or not.

NONE
Indicates that TCP/IP does not adjust the MSS for any connections.

AUTOIQDC | NOAUTOIQDC
Specifies whether to dynamically create Internal Queued Direct I/O (IQD) interfaces and transparently converge the dynamic IQD interfaces with the associated OSA interfaces for OSD CHPIDs. The IQD interface that is dynamically created and managed is logically converged with the OSA interface and is referred to as a HiperSockets Converged Interface (IQDC).

See “Steps for modifying” on page 35 for details about changing this parameter while the TCP/IP stack is active. See z/OS Communications Server: IP Configuration Guide for information about the HiperSockets Converged Interface and the IQD External Bridge function.

NOAUTOIQDC
Do not use dynamic IQD (HiperSockets) converged interfaces. This value is the default value.

AUTOIQDC
Dynamically manage access to IQD (HiperSockets). AUTOIQDC will dynamically create / delete, start / stop, and transparently converge IQD interfaces (IQDC) with your OSA interfaces for external networks. AUTOIQDC applies to IPv4 (IPAQENET) and IPv6 (IPAQNET6) OSD interface statements. The OSD interface statement must also have been defined with the virtual MAC (VMAC) parameter to request an OSA-generated VMAC address.

The OSA (OSD) CHPID associated with your OSA interface must have a PNetID configured in HCD for the associated OSA port. The dynamic IQD interface is created if an IQD CHPID is found to be configured (in HCD) with the External Bridge function and a PNetID that matches your OSA PNetID.

ALLTRAFFIC
Use IQD interfaces for all eligible outbound traffic flowing on the external IP data network. This value is the default value.

NOLARGEDATA
Do not use IQD dynamic interfaces for outbound TCP socket data transmissions of length 32 KB or larger. Use dynamic IQD interfaces for all other eligible outbound traffic. See z/OS Communications Server: IP Configuration Guide for more information about this setting.

Tips :

- When coding AUTOIQDC, also code IQDIOMULTIWRITE on the GLOBALCONFIG statement to optimize outbound write processing over all HiperSockets interfaces.
- Even when coding AUTOIQDC, some traffic might use the OSD interface to avoid fragmentation. If you use jumbo frames for your OSD interfaces that are associated with a converged HiperSockets CHPID, specify (in HCD) an IQD frame size larger than 16 KB when you configure your converged HiperSockets CHPID. This avoids fragmentation, which allows more traffic to flow over the converged HiperSockets interface.

AUTOIQDX | NOAUTOIQDX
Specifies whether to use dynamic Internal Queued Direct I/O extensions (IQDX) interfaces for connectivity to the intraensemble data network.

See “Steps for modifying” on page 35 for details about changing this parameter while the TCP/IP stack is active. See z/OS Communications Server: IP Configuration Guide for information about the intraensemble data network and the dynamic IQDX function.

NOAUTOIQDX
Do not use dynamic IQDX interfaces.

AUTOIQDX
Use dynamic IQDX interfaces when an IQD CHPID has been configured with the Internal Queued Direct I/O extensions (IQDX) function. This value is the default value.

ALLTRAFFIC
Use IQDX interfaces for all eligible outbound traffic on the intraensemble data network. This value is the default value.

NOLARGEDATA
Do not use IQDX interfaces for outbound TCP socket data transmissions of length 32KB or larger. Use IQDX interfaces for all other eligible outbound traffic. See z/OS Communications Server: IP Configuration Guide for more information about performance considerations for the IEDN-enabled HiperSockets function.

ECSALIMIT *ecsalimit* K | M

Specifies the maximum amount of extended common service area (ECSA) that TCP/IP can use. This limit can be expressed as a number followed by a K (which represents 1024 bytes), or a number followed by an M (which represents 1048576 bytes). If the K suffix is used, *ecsalimit* must be in the range 10240K and 2096128K inclusive or 0. If the M suffix is used, *ecsalimit* must be in the range 10M and 2047M inclusive or 0. The default is no limit, and it can be specified as 0 K or 0 M. The minimum value for ECSALIMIT and POOLLIMIT is not allowed to be set to a value if the current storage in use would be greater than or equal to 80% of that value (for example, not allowed to set it such that there is an immediate storage shortage).

ECSALIMIT ensures that TCP/IP does not overuse common storage. It is intended to improve system reliability by limiting TCP/IP's storage usage. The limit must account for peak storage usage during periods of high system activity or TCP/IP storage abends might occur. The limit does not include storage used by communications storage manager (CSM). CSM ECSA storage is managed independently of the TCP/IP ECSALIMIT. See z/OS Communications Server: SNA Network Implementation Guide for more information about CSM.

Specifying a nonzero ECSALIMIT enables warning messages EZZ4360I, EZZ4361I, and EZZ4362I to appear if a storage shortage occurs.

EXPLICITBINDPORTRANGE | NOEXPLICITBINDPORTRANGE

NOEXPLICITBINDPORTRANGE

Indicates that this stack does not participate in the allocation of ports from a pool of ports. The ports in the pool are guaranteed to be unique across the sysplex in that they are allocated to only one requestor in the sysplex at any one time, when processing an explicit bind() of a TCP socket to the IPv4 INADDR_ANY address, or to the IPv6 unspecified address (in6addr_any), and port 0.

EXPLICITBINDPORTRANGE

Indicates that this stack participates in the allocation of ports from a pool of ports guaranteed to be unique across the sysplex, when processing an explicit bind() of a TCP socket to the IPv4 INADDR_ANY address, or to the IPv6 unspecified address (in6addr_any), and port 0. This parameter also designates the range of ports that defines that pool. This parameter defines the range used by all stacks participating in EXPLICITBINDPORTRANGE port allocation processing throughout the sysplex. The most recently processed profile or OBEYFILE command that specifies EXPLICITBINDPORTRANGE defines the range for the sysplex.

Use this parameter so that you can specify distributed DVIPAs as the source IP address on DESTINATION or JOBNAME rules in a SRCIP block. See SRCIP statement.

1st_port

The starting port for the range of ports. The *1st_port* value is in the range 1024 - 65535. The sum of the *1st_port* value plus the *num_ports* value minus 1 cannot exceed 65535.

num_ports

The number of ports in the range. The *num_ports* value is in the range 1 - 64512. The sum of the *1st_port* value plus the *num_ports* value minus 1 cannot exceed 65535.

Guidelines:

- All TCP/IP stacks in the sysplex that participate in EXPLICITBINDPORTRANGE processing should have the same port range specified. To ensure this, specify the GLOBALCONFIG EXPLICITBINDPORTRANGE statement in a file that is specified in an INCLUDE statement in the TCP profiles data set of all the participating stacks.
- The port range defined on the EXPLICITBINDPORTRANGE parameter should not overlap any existing port reservations of any TCP/IP stacks in the sysplex. Any reserved ports that are within the EXPLICITBINDPORTRANGE range are excluded from the EXPLICITBINDPORTRANGE port pool, effectively making the pool smaller.
- The EXPLICITBINDPORTRANGE port range must be large enough to accommodate all applications in the sysplex that might issue explicit bind() calls for the IPv4 INADDR_ANY address, or for the IPv6 unspecified address (in6addr_any), and port 0.

- If additional TCP/IP stacks or systems are introduced into the sysplex, the extent of the port range defined by EXPLICITBINDPORTRANGE should be re-evaluated.
- If the size of the port range defined by the EXPLICITBINDPORTRANGE parameter is too large, there are fewer ports available for local ephemeral port allocation.

Restriction: In a common INET (CINET) environment, this parameter is accepted, but the EXPLICITBINDPORTRANGE function is supported in a limited set of conditions only. It is supported when CINET is managing one stack only on the system or when the affected application has established stack affinity. Otherwise, results can be unpredictable.

IQDMULTIWRITE | NOIQDMULTIWRITE
Specifies whether HiperSockets interfaces should use multiple write support. HiperSockets multiple write might reduce CPU usage and might provide a performance improvement for large outbound messages that are typically generated by traditional streaming workloads such as file transfer, and interactive web-based services workloads such as XML or SOAP. This parameter applies to all HiperSockets interfaces, including IUTIQDIO and IQDIOINTF6 interfaces created for Dynamic XCF.

Restriction: HiperSockets multiple write is effective only on an IBM System z10 or later and when z/OS is not running as a guest in a z/VM® environment.

See the modifying information in this topic for details about changing this parameter while the TCP/IP stack is active. See the HiperSockets multiple write information in z/OS Communications Server: IP Configuration Guide for more information about HiperSockets multiple write support.

NOIQDMULTIWRITE
HiperSockets interfaces do not use the multiple write support. This is the default.

IQDMULTIWRITE
HiperSockets interfaces do use the multiple write support.

IQDVLANID *vlan_id*
Specifies a VLAN ID to be used when HiperSockets (iQDIO) connectivity is used for dynamic XCF support. VLAN IDs are used to partition communication across HiperSockets. Stacks on the same CPC using the same HiperSockets CHPID that use the same VLAN ID can establish communications; stacks on the same CPC using the same HiperSockets CHPID that use different VLAN IDs cannot.

The specified value, *vlan_id*, is used for both IPv4 and IPv6 DYNAMICXCF HiperSockets connectivity. This parameter is intended to be used in conjunction with the GLOBALCONFIG XCFGRPID parameter to support subplexing.

Subplexing enables TCP/IP participation in a Sysplex to be partitioned into subsets based on the XCFGRPID value. When using subplexing, TCP/IP stacks with the same XCFGRPID value should specify the same IQDVLANID value. Stacks with different XCFGRPID values should have different IQDVLANID values. If you have stacks in the default subplex (that is, stacks that do not specify an XCFGRPID value) that use the same HiperSockets CHPID as stacks within a non-default subplex (an XCFGRPID value was specified), then the stacks in the default subplex should specify an IQDVLANID value that is different from the other IQDVLANID values specified by the other non-default subplex stacks that use the same HiperSockets CHPID.

Restriction: The IQDVLANID parameter can be specified only in the initial profile.

Valid VLAN IDs are in the range 1 - 4094. For more information about VLANs and Hipersockets see z/OS Communications Server: IP Configuration Guide.

MAXRECS
Specifies the maximum number of records to be displayed by the DISPLAY TCPIP,,NETSTAT operator command. The term *records* refers to the number of entries displayed on each report. For example, for the connection-related reports, a record is a TCP connection or listener, or a UDP endpoint. This configured value is used when the MAX parameter is not explicitly specified on the command. The default value is 100. If the number of output lines exceeds the maximum number of lines for a multi-line Write to Operator (WTO), the report output is truncated. See the information about the Display TCPIP,,NETSTAT command in z/OS Communications Server: IP System Administrator's Commands for more details about the command.

***

A value of asterisk (*) specifies that all records are to be displayed.

recs

This value specifies the number of records to be displayed. The valid range is 1 - 65535.

MLSCHKTERMINATE | NOMLSCHKTERMINATE

NOMLSCHKTERMINATE

Specifies that the stack should remain active after writing an informational message when inconsistent configuration information is discovered in a multilevel-secure environment.

Informational message EZD1217I is written to the system console summarizing the number of problems found. Additional informational messages between EZD1219I and EZD1234I are written to the job log for each configuration inconsistency found.

This is the default value.

MLSCHKTERMINATE

Specifies that the stack should be terminated after writing an informational message when inconsistent configuration information is discovered in a multilevel-secure environment.

Informational message EZD1217I is written to the system console summarizing the number of problems found. Additional informational messages between EZD1219I and EZD1234I are written to the job log for each configuration inconsistency found.

POOLLIMIT *pool_limit* K | M

Specifies the maximum amount of authorized private storage that TCP/IP can use within the TCP/IP address space. This limit can be expressed as a number followed by a K (which represents 1024 bytes), or a number followed by an M (which represents 1048576 bytes). If the K suffix is used, *pool_limit* must be in the range 10240K and 2096128K inclusive or 0. If the M suffix is used, *pool_limit* must be in the range 10M and 2047M inclusive or 0. The default is no limit, and it can be specified as 0K or 0M. The minimum value for ECSALIMIT and POOLLIMIT is not allowed to be set to a value if the current storage in use would be greater than or equal to 80% of that value (for example, not allowed to set it such that there is an immediate storage shortage).

POOLLIMIT ensures that TCP/IP does not overuse its authorized private storage. Most systems can use the default POOLLIMIT (no limit). Systems with limited paging capacity can use POOLLIMIT to help limit TCP/IP storage usage. If the limit is used, it must account for peak storage usage during periods of high system activity or TCP/IP storage abends might occur.

POOLLIMIT can be higher than the REGION size on the TCP/IP start procedure because POOLLIMIT applies to authorized storage, whereas REGION applies to unauthorized storage. Specifying a nonzero POOLLIMIT enables warning messages EZZ4364I, EZZ4365I, and EZZ4366I to appear if a storage shortage occurs.

SEGMENTATIONOFFLOAD | NOSEGMENTATIONOFFLOAD

Specifies whether the stack should offload TCP segmentation for IPv4 packets to OSA-Express features. TCP segmentation offload support transfers the overhead of segmenting outbound data into individual TCP packets to QDIO-attached OSA-Express devices whose features that support this function. Offloading segmentation of streaming-type workloads reduces CPU use and increases throughput. This parameter is ignored for OSA-Express features that do not support segmentation offload.

Guideline : The support for specifying IPv4 segmentation offload on the GLOBALCONFIG profile statement has been deprecated. The parameters are still supported on the GLOBALCONFIG statement, but the support for specifying these parameters on the GLOBALCONFIG statement will be dropped in a future release. It is recommended to specify these parameters on the IPCONFIG profile statement instead.

Rule : The SEGMENTATIONOFFLOAD and NOSEGMENTATIONOFFLOAD parameters specified on the IPCONFIG statement override the equivalent parameters specified on the GLOBALCONFIG statement.

See the Modifying topic for information about changing this parameter while the TCP/IP stack is active. See TCP segmentation offload information in z/OS Communications Server: IP Configuration Guide for more information about TCP segmentation offload support.

NOSEGMENTATIONOFFLOAD
TCP segmentation is performed by the TCP/IP stack. This is the default.

SEGMENTATIONOFFLOAD
TCP segmentation is offloaded to the OSA-Express feature.

SMCD | NOSMCD
Specifies whether this stack uses Shared Memory Communications - Direct Memory Access (SMC-D). For more information about SMC-D, see in z/OS Communications Server: IP Configuration Guide.

NOSMCD
Specifies that this stack should not use SMC-D communications. This is the default setting.

SMCD
Specifies that this stack should use SMC-D communications. You can use this parameter to define operational characteristics for SMC-D communications.

Result : The AUTOCACHE and AUTOSMC monitoring functions are started if SMCGLOBAL AUTOCACHE and AUTOSMC are configured, either by default or by being explicitly specified.

If you specify the SMCD parameter without any subparameters, you get one of the following results:

- If you specify the SMCD parameter for the first time, the FIXEDMEMORY and TCPKEEPMININTERVAL subparameters are set to default values.
- If you previously specified the SMCD parameter with subparameters, TCP/IP retains the knowledge of the subparameter settings, even if SMC-D processing is stopped by issuing the VARY TCPIP,,OBEYFILE command with a data set that contains a GLOBALCONFIG NOSMCD parameter. Therefore, a subsequent specification of a GLOBALCONFIG SMCD profile statement resumes SMC-D processing with the previous subparameter settings.

FIXEDMEMORY *mem_size*
Specifies the maximum amount of 64-bit storage that the stack can use for the receive buffers that are required for SMC-D communications. The *mem_size* value is an integer in the range 30 - 9999, and represents the maximum storage in megabytes of data. The default value is 256 megabytes.

TCPKEEPMININTERVAL *interval*
This interval specifies the minimum interval that TCP keepalive packets are sent on the TCP path of an SMC-D link.

Rules :

- If a keepalive interval is also specified on the INTERVAL parameter of the TCPCONFIG statement or is set for a specific SMC-D link socket by the TCP_KEEPALIVE setsockopt() option, the largest of the three interval values is used.
- The valid range for this interval is 0-2147460 seconds, and the default value is 300 seconds.
- A value of 0 disables TCP keepalive probe packets on the TCP path of an SMC-D link.
- The SO_KEEPALIVE setsockopt() option must be set to use keepalive processing.

Result : The TCPKEEPMININTERVAL setting has no effect on keepalive processing for the SMC-D path of an SMC-D link.

For more information about TCP keepalive processing for the TCP path and the SMC-D path of SMC-D links, see TCP keepalive in z/OS Communications Server: IP Configuration Guide.

SMCGLOBAL

Specifies global settings for Shared Memory Communications (SMC). SMC includes Shared Memory Communications over Remote Direct Memory Access (RDMA), or SMC-R, for external data network communications and Shared Memory Communications - Direct Memory Access (SMC-D). For more

information about SMC-R and SMC-D, see Shared Memory Communications in z/OS Communications Server: IP Configuration Guide.

AUTOCACHE | NOAUTOCACHE

Specifies whether this stack caches unsuccessful attempts to use SMC communication. Use SMCGLOBAL AUTOCACHE to prevent the overhead of persistent attempts to establish a TCP connection to a specific destination IP address over SMC if previous attempts to the same destination failed.

NOAUTOCACHE

Specifies that this stack does not cache unsuccessful attempts to create an SMC-R or SMC-D link and that existing SMC cache entries are cleared.

AUTOCACHE

Specifies that this stack caches unsuccessful attempts to create an SMC-R or SMC-D link per destination IP address. Subsequent TCP connections to the same destination bypass the use of SMC while the IP address is cached. To clear this cache, specify the NOAUTOCACHE subparameter. Cached entries remain in effect for approximately 20 minutes. AUTOCACHE is the default setting. The AUTOCACHE function is started only when you enable SMC. For more information about enabling SMC, see the description of the GLOBALCONFIG SMCR and SMCD parameters.

AUTOSMC | NOAUTOSMC

Specifies whether this stack monitors inbound TCP connections to dynamically determine whether SMC is beneficial for a local TCP server application. Results of this monitoring influence whether TCP connections to a particular server or port use SMC. AUTOSMC monitoring ensures that TCP connections use the most appropriate communications protocol, either TCP or SMC. You can use the Netstat ALL/-A command to monitor the results of this dynamic monitoring and SMC enablement or disablement. For more information about the Netstat ALL/-A command, see Netstat ALL/-A report in z/OS Communications Server: IP System Administrator's Commands.

Guideline : Configuration of either SMC or NOSMC on the PORT or PORTRANGE statement overrides configuration of the AUTOSMC monitoring function for particular servers. For more information, see PORT statement and PORTRANGE statement.

NOAUTOSMC

Specifies that this stack does not monitor inbound TCP connections to determine whether the connections can benefit from using SMC.

AUTOSMC

Specifies that this stack monitors inbound TCP connections to determine whether the connections can benefit from using SMC. AUTOSMC is the default setting. The AUTOSMC monitoring function is started only when you enable SMC. For more information about enabling SMC, see the description of the GLOBALCONFIG SMCR and SMCD parameters.

SMCR | NOSMCR

Specifies whether this stack uses Shared Memory Communications over Remote Direct Memory Access (RDMA), or SMC-R, for intraensemble data network (IEDN) or external data network communications. For more information about SMC-R, see Shared Memory Communications over Remote Direct Memory Access in z/OS Communications Server: IP Configuration Guide.

NOSMCR

Specifies that this stack should not use SMC-R for IEDN or external data network communications. This is the default setting.

SMCR

Specifies that this stack should use SMC-R for IEDN or external data network communications. Use this parameter to define the "RoCE Express" features that this stack should use for SMC-R communications. You can use this parameter to define additional operational characteristics for SMC-R communications.

Result : If at least one PFID is defined, the AUTOCACHE and AUTOSMC monitoring functions are started if SMCGLOBAL AUTOCACHE and AUTOSMC are configured, either by default or by being explicitly specified.

If you specify the SMCR parameter without any subparameters, you get one of the following results:

- If this is the first time that you specify the SMCR parameter, the following results occur:
  - No Peripheral Component Interconnect® Express (PCIe) function IDs are defined.
  - FIXEDMEMORY and TCPKEEPMININTERVAL subparameters are set to default values.
- If you previously specified the SMCR parameter with subparameters, TCP/IP retains the knowledge of the subparameter settings, even if SMC-R processing is stopped by issuing the VARY TCPIP,,OBEYFILE command with a data set that contains a GLOBALCONFIG NOSMCR parameter. Therefore, a subsequent specification of a GLOBALCONFIG SMCR profile statement resumes SMC-R processing with the previous subparameter settings.

PFID *pfid*

Specifies the Peripheral Component Interconnect Express (PCIe) function ID (PFID) value for a "RoCE Express" feature that this stack uses. A *pfid* is a 2-byte hexadecimal value that identifies this TCP/IP stack's representation of a "RoCE Express" feature. z/OS supports values for *pfid* in the range 0 to FFFF. The maximum supported PFID value depends on the System z® machine level.

Rules :

- You must code at least one PFID subparameter for this stack to use SMC-R communications.
- You can specify a maximum of 16 PFID subparameter values on the SMCR parameter.
- The value for each PFID and PORTNUM pair must be unique.
- When the "RoCE Express" feature operates in a shared RoCE environment, you cannot simultaneously activate a "RoCE Express" feature that uses the same PFID value from different TCP/IP stacks within the same logical partition (LPAR).

PORTNUM *num*

Specifies the "RoCE Express" port number to use for a particular PFID. Configure each PFID to use only a single port. The port number can be 1 or 2; 1 is the default value.

Rules :

- You do not need to configure PORTNUM for IBM RoCE Express2 features in the TCP/IP profile. The correct port number for these features is configured in the Hardware Configuration Definition (HCD) and is learned by VTAM® and the TCP/IP stack during PFID activation. VTAM ignores the GLOBALCONFIG SMCR PORTNUM value if it differs from the port number configured in the HCD for the IBM RoCE Express2 feature.
- If the 10 GbE RoCE Express feature operates in a dedicated RoCE environment, you can activate either port 1 or port 2 but not both simultaneously for an individual PFID value. If PORTNUM 1 and PORTNUM 2 definitions for the same PFID value are created, the port that is first activated is used.
- If the 10 GbE RoCE Express feature operates in a shared RoCE environment, you can use both port 1 and port 2 on an individual RNIC adapter, but the PFID value that is associated with each port must be different. You cannot simultaneously activate PORTNUM 1 and PORTNUM 2 definitions for the same PFID value.

  For example, if PFID 0013 and PFID 0014 are both defined in HCD to represent the RNIC adapter with PCHID value 0140, you can configure `PFID 0013 PORT 1 PFID 0014 PORT 2` to use both ports on the RNIC adapter. However, if you specify `PFID 0013 PORT 1 PFID 0013 PORT 2`, only the first port that is activated is used.

MTU *mtusize*

Specifies the maximum transmission unit (MTU) value to be used for a particular PFID. The MTU value can be 1024, 2048, or 4096. The default value is 1024 and can be used for most workloads. If you set the MTU size to 2048 or 4096, you must also enable jumbo frames on all switches in the network path for all peer hosts. For more information about the RoCE maximum transmission unit, see z/OS Communications Server: IP Configuration Guide.

FIXEDMEMORY ***mem_size***
Specifies the maximum amount of 64-bit storage that the stack can use for the send and receive buffers that are required for SMC-R communications. The *mem_size* value is an integer in the range 30 - 9999, and represents the maximum storage in megabytes of data. The default value is 256 megabytes.

TCPKEEPMININTERVAL ***interval***
This interval specifies the minimum interval that TCP keepalive packets are sent on the TCP path of an SMC-R link.

Rules :

- If a keepalive interval is also specified on the INTERVAL parameter of the TCPCONFIG statement or is set for a specific SMC-R link socket by the TCP_KEEPALIVE setsockopt() option, the largest of the three interval values is used.
- The valid range for this interval is 0-2147460 seconds, and the default is 300 seconds.
- A value of 0 disables TCP keepalive probe packets on the TCP path of an SMC-R link.
- The SO_KEEPALIVE setsockopt() option must be set for keepalive processing to be used.

Result : The TCPKEEPMININTERVAL setting has no effect on keepalive processing for the SMC-R path of an SMC-R link.

For more information about TCP keepalive processing for the TCP path and the SMC-R path of SMC-R links, see TCP keepalive in z/OS Communications Server: IP Configuration Guide.

SYSPLEXMONITOR
Specifies SYSPLEXMONITOR subparameters to configure the operation of the sysplex autonomics function. For more information about connectivity problems in a sysplex, see z/OS Communications Server: IP Configuration Guide.

If the SYSPLEXMONITOR parameter is not specified in the initial TCP/IP profile, then the sysplex autonomics function uses the default values for all SYSPLEXMONITOR subparameters. If the SYSPLEXMONITOR parameter is specified but not all subparameters are specified in the initial TCP/IP profile, then the sysplex autonomics function uses the default values for those SYSPLEXMONITOR subparameters that are not specified. For example, if SYSPLEXMONITOR is specified without RECOVERY or NORECOVERY specified in the initial profile, then the NORECOVERY action is in effect.

Rule: If you specify the GLOBALCONFIG statement in a data set associated with a VARY TCPIP,,OBEYFILE command and the SYSPLEXMONITOR parameter is specified without any subparameters, an informational message is issued and the parameter is ignored.

AUTOREJOIN | NOAUTOREJOIN
Specifies whether TCP/IP should automatically rejoin the TCP/IP sysplex group when a detected problem is relieved after the stack has left the sysplex group.

NOAUTOREJOIN
Do not rejoin the TCP/IP sysplex group when a detected problem is relieved. This is the default value.

AUTOREJOIN
When all detected problems (that caused the stack to leave the sysplex group) are relieved, the stack automatically rejoins the sysplex group and reprocesses the saved VIPADYNAMIC block configuration.

Restriction: AUTOREJOIN cannot be configured when NORECOVERY is configured (or set to the default value).

Guideline: AUTOREJOIN should be used when RECOVERY is configured to allow the stack to rejoin the sysplex group without operator intervention.

DELAYJOIN | NODELAYJOIN
Specify whether TCP/IP should delay joining or rejoining the TCP/IP sysplex group (EZBTCPCS) during stack initialization, or rejoining the sysplex group following a VARY TCPIP,,OBEYFILE command.

NODELAYJOIN
Attempt to join the TCP/IP sysplex group. When specified during stack initialization, the stack attempts to join the sysplex group. This is the default value.

DELAYJOIN
Delay joining the TCP/IP sysplex group and processing any VIPADYNAMIC block or DYNAMICXCF statements during stack initialization until OMPROUTE is started and active.

DYNROUTE | NODYNROUTE
Specifies whether TCP/IP should monitor the presence of dynamic routes over monitored network links or interfaces.

NODYNROUTE
The TCP/IP stack should not monitor the presence of dynamic routes over monitored network links or interfaces. When MONINTERFACE is not configured, this is the default value.

DYNROUTE
The TCP/IP stack should monitor the presence of dynamic routes over monitored network links or interfaces.

Tip: This level of monitoring is useful in detecting problems that OMPROUTE is having in communicating with other routing daemons on the selected network interfaces.

If no dynamic routes are present in the TCP/IP stack from that network, a specific interface attached to that network might not be active or routers attached to that network might not be active or healthy. In either case, when these conditions are detected, they provide a reasonable indication that client requests for DVIPAs or distributed DVIPAs owned by this TCP/IP stack might not reach this stack over that interface. These checks can help further qualify the state of a network interface on this TCP/IP stack. When the MONINTERFACE parameter is specified, This is the default value.

Restriction: DYNROUTE cannot be specified when NOMONINTERFACE is configured (or is the default value).

Rules:

- Specify DYNROUTE only when OMPROUTE is configured and started; otherwise, the TCP/IP stack might be forced to leave the TCP/IP sysplex group if RECOVERY is coded.
- If DYNROUTE is specified, also specify DELAYJOIN to avoid a scenario where the TCP/IP stack leaves the TCP/IP sysplex group before OMPROUTE is started.

NOJOIN
Specifies that the TCP/IP stack should not join the TCP/IP sysplex group (EZBTCPCS) during stack initialization. If this value is specified, the TCP/IP stack does not process any VIPADYNAMIC block or DYNAMICXCF statements. Any other GLOBALCONFIG SYSPLEXMONITOR parameter settings (configured or default) are ignored, and the settings are saved in case you want the TCP/IP stack to join the sysplex group at a later time.

If you subsequently issue a VARY TCPIP,,SYSPLEX,JOINGROUP command, the NOJOIN setting is overridden and the saved GLOBALCONFIG SYSPLEXMONITOR parameter settings become active. For example, if you configure NOJOIN and DELAYJOIN, DELAYJOIN is initially ignored. If you subsequently issue a V TCPIP,,SYSPLEX,JOINGROUP command, NOJOIN is overridden, DELAYJOIN becomes active, and the stack joins the sysplex group if OMPROUTE is initialized.

Any sysplex-related definitions within the TCP/IP profile, such as VIPADYNAMIC or IPCONFIG DYNAMICXCF statements, are not processed until the TCP/IP stack joins the sysplex group.

Restriction: You can specify this parameter only in the initial profile; you cannot specify it when you issue a VARY TCPIP,,OBEYFILE command.

MONINTERFACE | NOMONINTERFACE

NOMONINTERFACE
The TCP/IP stack should not monitor the status of any network links or interfaces. This is the default.

MONINTERFACE
The TCP/IP stack should monitor the status of specified network link or interfaces. The interfaces or links being monitored are those that are configured with the MONSYSPLEX keyword on the LINK or INTERFACE statement. See Summary of DEVICE and LINK statements or Summary of INTERFACE statements for more information.

Guideline: This level of monitoring can further qualify the health of the TCP/IP stack by ensuring that at least one key interface is active and available. This option can be useful in environments where the dynamic XCF interface is not configured as an alternate network path for this stack (for example, where no dynamic routes are advertised over dynamic XCF interfaces and no static or replaceable static routes are defined over those interfaces).

RECOVERY | NORECOVERY
Specify the action to be taken when a sysplex problem is detected.

NORECOVERY
When a problem is detected, issue messages regarding the problem but take no further action. This is the default value.

RECOVERY
When a problem is detected, issue messages regarding the problem, leave the TCP/IP sysplex group, and delete all DVIPA resources owned by this stack. As allowed by a configuration with backup capabilities, other members of the TCP/IP sysplex automatically take over the functions of this member that was removed from the TCP/IP sysplex group.

Recovery is the preferred method of operation because other members of the TCP/IP sysplex can automatically take over the functions of a member with no actions needed by an operator. IBM Health Checker for z/OS enhancements can be used to check whether the RECOVERY parameter has been specified when the IPCONFIG DYNAMICXCF or IPCONFIG6 DYNAMICXCF parameters have been specified. For more details about IBM Health Checker for z/OS enhancements, see the IBM Health Checker for z/OS enhancements information in the z/OS Communications Server: IP Diagnosis Guide.

TIMERSECS *seconds*
Time value specified in seconds. Determines how quickly the sysplex monitor reacts to problems with needed sysplex resources. Valid values are in the range 10 - 3600 seconds. The default value is 60 seconds.

SYSPLEXWLMPOLL *seconds*
Time value specified in seconds. Determines how quickly the sysplex distributor and its target servers poll WLM for new weight values. A short time results in quicker reactions to changes in target status. Valid values are in the range is 1 - 180 seconds. The default value is 60 seconds.

TCPIPSTATISTICS | NOTCPIPSTATISTICS

NOTCPIPSTATISTICS
Indicates that the TCP/IP counter values are not to be written to the output data set designated by the CFGPRINT JCL statement.

The NOTCPIPSTATISTICS parameter is confirmed by the message:

```
EZZ0613I TCPIPSTATISTICS IS DISABLED
```

This is the default value.

TCPIPSTATISTICS
Prints the values of several TCP/IP counters to the output data set designated by the CFGPRINT JCL statement. These counters include number of TCP retransmissions and the total number of TCP segments sent from the MVS TCP/IP system. These TCP/IP statistics are written to the designated output data set only during termination of the TCP/IP address space.

The TCPIPSTATISTICS parameter is confirmed by the message:

```
EZZ0613I TCPIPSTATISTICS IS ENABLED
```

The SMFCONFIG TCPIPSTATISTICS parameter (see SMFCONFIG statement) serves a different purpose. It requests that SMF records of subtype 5 containing TCP/IP statistics be created. These statistics are recorded in SMF type 118 or 119, subtype 5 records.

WLMPRIORITYQ | NOWLMPRIORITYQ

Specifies whether OSA-Express QDIO write priority values should be assigned to packets associated with WorkLoad Manager service classes, and to forwarded packets. See the information about prioritizing outbound OSA-Express data using the WorkLoad Manager service class in z/OS Communications Server: IP Configuration Guide .

NOWLMPRIORITYQ

Specifies that OSA-Express QDIO write priority values should not be assigned to packets associated with WorkLoad Manager service class values or to forwarded packets. This value is the default.

WLMPRIORITYQ

Specifies that OSA-Express QDIO write priority values should be assigned to packets associated with WorkLoad Manager service class valuesand to forwarded packets.

You can assign specific OSA-Express QDIO write priority values by using the IOPRI*n* subparameters, where *n* is one or more of the priority values in the range 1 - 4. For each subparameter, you can specify a control value in the range 0 - 6, which correlates to the WLM services classes, or you can specify the keyword FWD for forwarded packets. WLM supports a service class for the SYSTEM value, but this value is always assigned the OSA-Express QDIO write priority 1 and its assignment cannot be configured; therefore, a control value is not assigned for the SYSTEM WLM service class.

You can use the default assignment by specifying the WLMPRIORITYQ parameter without any IOPRI*n* subparameters. See the description of the *default_control_values* variable in this topic to understand the default assignment.

control_values

Control values are used to represent the WLM service classes and forwarded packets. Valid control values are the digits 0 - 6, which represent WLM service classes, or the keyword FWD, which represents forwarded packets. Table 7 on page 31 identifies the control value, the type of packet that it represents, and the default QDIO priority assigned to the packet:

Table 7. WLM Service Class Importance Levels

Control value	Type of packet	Default QDIO priority
0	System-defined service class (SYSSTC) used for high-priority started tasks	1
1	User-defined service classes with importance level 1	2
2	User-defined service classes with importance level 2	3
3	User-defined service classes with importance level 3	3
4	User-defined service classes with importance level 4	4
5	User-defined service classes with importance level 5	4
6	User-defined service classes associated with a discretionary goal	4
FWD	Forwarded packets	4

default_control_values
When the WLMPRIORITYQ parameter is specified without any IOPRIn subparameters, then the OSA-Express QDIO write priority values are assigned as shown Table 7 on page 31.

IOPRI*n* *control_values*
Use the IOPRIn subparameters to correlate control values with specific OSA-Express QDIO write priority values. You can use one or more of the following subparameter keywords:

- IOPRI1
- IOPRI2
- IOPRI3
- IOPRI4

Each subparameter keyword corresponds to one of the four QDIO write priority values, 1 through 4. Each subparameter can be specified once on a GLOBALCONFIG statement.

control_values
Indicates the type of packet to which the QDIO write priority value should be assigned. Valid values are:

Digits 0 - 6
Causes the QDIO write priority value that is specified by the IOPRI*n* subparameter to be assigned to packets associated with the WLM service classes represented by the control value.

FWD
This keyword causes the QDIO write priority value indicated by the IOPRIn subparameter to be assigned to forwarded packets.

Rules:

- IOPRIn must be followed by one or more priority level releases.
- You can specify more than one control value for an IOPRIn subparameter. Each control value must be separated by at least one blank.
- A specific control value can be specified only once in the set of IOPRIn subparameters on a GLOBALCONFIG statement.
- If any control value is not explicitly specified on an IOPRI*n* subparameter, then the associated packets are assigned a default QDIO write priority 4.

In the following example, QDIO priority 1 is assigned to packets associated with control values 0 and 1, QDIO priority 2 is assigned to packets associated with control value 2 and to forwarded packets, QDIO priority 3 is assigned to packets associated with control values 3 and 4, and QDIO priority 4 is assigned to packets associated with control values 5 and 6.

```
WLMPRIORITYQ  IOPRI1 0 1
              IOPRI2 2 FWD
              IOPRI3 3 4
              IOPRI4 5 6
```

XCFGRPID *group_id*
This parameter is needed only if you want subplexing. If specified, the value provides a 2-digit suffix that is used in generating the XCF group name that the TCP/IP stack joins. Valid values are in the range 2 - 31. The group name is EZBT*vvtt*, where the *vv* value is the VTAM XCF group ID suffix (specified with the XCFGRPID VTAM start option) and the *tt* value is the *group_id* value supplied on this parameter, used as a 2-digit value converted to character format. If no VTAM XCF group ID suffix was specified, the group name is EZBTCP*tt.* If no VTAM XCF group ID suffix and no TCP XCF group ID suffix is specified, the group name is EZBTCPCS.

These characters are also used as a suffix for the EZBDVIPA and EZBEPORT structure names, in the form EZBDVIPA*vvtt* and EZBEPORT*vvtt*. If no VTAM XCF group ID suffix was specified, the structure names are EZBDVIPA01*tt* and EZBEPORT01*tt*.

If XCFGRPID is not specified, the XCF group name is EZBTvvCS and the structure names are EZBDVIPA*vv* and EZBEPORT*vv*. If no VTAM XCF group id suffix was specified, the group name is EZBTCPCS and the structure names are EZBDVIPA and EZBEPORT.

Restriction: XCFGRPID can be specified only in the initial profile.

This allows multiple TCP/IP stacks to join separate Sysplex groups and access separate Coupling Facility structures, isolating sets of TCP/IP stacks into subplexes with XCF communication only with other TCP/IP stacks within the same subplex.

If HiperSockets is supported on this system, the IQDVLANID parameter, on the GLOBALCONFIG statement, must be specified if XCFGRPID is specified. Stacks on the same CPC using the same HiperSockets CHPID that specify the same XCFGRPID value must specify the same IQDVLANID value.

Stacks on the same CPC using the same HiperSockets CHPID specifying different XCFGRPID values must specify different IQDVLANID values. This allows partitioning of connectivity across the Sysplex to include partitioning of connectivity across HiperSockets.

Creating TCP/IP and VTAM subplexes can add some complexity to your VTAM and TCP/IP configurations and requires careful planning. Before setting this parameter you should review the information about setting up a subplex in the z/OS Communications Server: IP Configuration Guide.

ZERT|NOZERT

Specifies whether the z/OS Encryption Readiness Technology (zERT) will monitor TCP and Enterprise Extender traffic on this TCP/IP stack.

NOZERT

Indicates that the TCP and Enterprise Extender traffic will not be monitored by zERT. This is the default value.

ZERT

Indicates that TCP and Enterprise Extender traffic will be monitored by zERT. The zERT discovery function, which monitors and collects information about security sessions on a per-TCP- and per-EE connection basis, is always enabled when ZERT is specified. In addition, you can specify subparameters to enable other zERT functions.

AGGregation

Indicates that the zERT aggregation function is enabled. zERT aggregation uses the information collected by zERT discovery to create summarized security session information. The zERT aggregation information is reported at fixed intervals of time, as defined by the configured SMF interval value. For more details on the zERT aggregation function, see What does zERT aggregation collect? in z/OS Communications Server: IP Configuration Guide.

INTVal

Specifies the interval at which zERT aggregation will write the data it has collected as SMF type 119 subtype 12 (zERT summary) records.

SMF

Indicates the system's SMF interval is to be used. This is the default.

nn

Specifies the recording interval in one hour increments. Valid values are 1 through 24.

SYNCval *hh:mm*

Indicates a reference time for which zERT aggregation records will be recorded. SYNCVAL is a sub-parameter of the INTVAL sub-parameter. This field is in the form of 24 hour clock format hh:mm (hour and minute value separated by a colon). This reference time will be used to indicate the point from which the first zERT summary records should be recorded (SMF 119 subtype 12). If not specified, the default value is midnight or 00:00.

Specifies a reference time upon which INTVAL is based. This is the point from which aggregation should record its first set of SMF type 119 subtype 12 (zERT summary) records. The value is a time of day specified in hours and minutes in 24 hour clock format.

hh

Specifies the hour of the day, between midnight (00) and 11 PM (23).

:
Is a required separator with no intervening blanks.

mm
Specifies the minutes after the hour, between 00 and 59.

If SYNCVAL is not specified, the default is midnight (00:00).

NOAGGregation
Indicates that the zERT aggregation function is disabled. This is the default.

Note that additional configuration is required to specify where zERT data is to be written. For more information of those configuration parameters, see SMFCONFIG statement and NETMONITOR statement. Separate SMFCONFIG controls exist for zERT discovery records and for zERT aggregation records.

See the Steps for modifying in this topic for details about changing this parameter while the TCP/IP stack is active.

ZIIP
Specifies subparameters that control whether TCP/IP displaces CPU cycles onto a System z9® Integrated Information Processor (zIIP). You must specify at least one subparameter. If the ZIIP parameter is specified with no subparameters, an informational message is issued and the parameter is ignored.

IPSECURITY | NOIPSECURITY
Specifies whether TCP/IP should displace CPU cycles for IPSec workload to a zIIP. For more information about this function, see the Additional IPSec assist using z9® Integrated Information Processor (zIIP IP security) topic in z/OS Communications Server: IP Configuration Guide.

NOIPSECURITY
Do not displace CPU cycles for IPSec workload to a zIIP. This is the default value.

IPSECURITY
When possible, displace CPU cycles for IPSec workload to a zIIP. Workload Manager (WLM) definitions should be examined and possible changes made before this option is used. See the more detailed description in the additional IPSec Assist by way of z9 Integrated Information Processor (zIIP IPSECURITY) topic in z/OS Communications Server: IP Configuration Guide.

IQDIOMULTIWRITE | NOIQDIOMULTIWRITE
Specifies whether TCP/IP should displace CPU cycles for large outbound TCP messages that are typically created by traditional streaming work loads such as file transfer, and interactive web-based service workloads such as XML or SOAP. The TCP/IP outbound message must be at least 32KB in length before the write processing is off-loaded to an available zIIP specialty engine. For more information about this function, see the information about HiperSockets multiple write assist with IBM zIIP in z/OS Communications Server: IP Configuration Guide.

NOIQDIOMULTIWRITE
Do not displace CPU cycles for the writing of large TCP outbound messages to a zIIP. This is the default value.

IQDIOMULTIWRITE
When possible, displace CPU cycles for the writing of large TCP outbound messages to a zIIP.

Rules:

- You cannot specify IQDIOMULTIWRITE as a ZIIP parameter when GLOBALCONFIG IQDMULTIWRITE is not configured. When GLOBALCONFIG IQDMULTIWRITE is not configured, HiperSockets interfaces do not use the multiple write support.
- Only large TCP outbound messages (32KB and larger) are processed on the zIIP specialty engine.
- The TCP message must be originating from this node. Routed TCP messages are not eligible for zIIP assistance.

Tip: These ZIIP parameters apply to pre-defined HiperSockets interfaces, as well as HiperSockets interfaces that are created and used by dynamic XCF definitions.

Steps for modifying

To modify parameters for the GLOBALCONFIG statement, you must respecify the statement with the new parameters.

The following list describes how to modify individual parameters:

AUTOIQDC and NOAUTOIQDC

If you use the VARY TCPIP,,OBEYFILE command to change this parameter from AUTOIQDC to NOAUTOIQDC, no new dynamic IQDC interfaces will be activated. All active dynamic IQDC interfaces will remain active and available for use. To stop existing interfaces, you must issue a V TCPIP,,STOP command for each active IQDC interface.

If you use the VARY TCPIP,,OBEYFILE command to change this parameter from NOAUTOIQDC to AUTOIQDC, active OSD interfaces are not affected, but the stack will attempt to activate a dynamic IQDC interface on any subsequent OSD activations.

AUTOIQDX and NOAUTOIQDX

If you use the VARY TCPIP,,OBEYFILE command to change this parameter from AUTOIQDX to NOAUTOIQDX, no new dynamic IQDX interfaces will be activated. All active dynamic IQDX interfaces will remain active and available for use. To stop existing interfaces, you must issue a V TCPIP,,STOP command for each active IQDX interface.

If you use the VARY TCPIP,,OBEYFILE command to change this parameter from NOAUTOIQDX to AUTOIQDX, active OSX interfaces are not affected, but the stack will attempt to activate a dynamic IQDX interface on any subsequent OSX activations.

EXPLICITBINDPORTRANGE and NOEXPLICITBINDPORTRANGE

If you specified the EXPLICITBINDPORTRANGE parameter and then you change to the NOEXPLICITBINDRANGE parameter, then the stack stops allocating more ports from the EXPLICITBINDPORTRANGE pool. However, the existing active range for the EXPLICITBINDPORTRANGE pool in the coupling facility is unaffected unless you are changing the parameter on the last stack in the sysplex using this function.

If you specified the NOEXPLICITBINDPORTRANGE parameter and then you change to the EXPLICITBINDPORTRANGE parameter, then a range of ports used for the EXPLICITBINDPORTRANGE pool is set. The stack uses ports from that pool for explicit bind() requests to the IPv4 INADDR_ANY address, or to the IPv6 unspecified address (in6addr_any), and port 0. If the range specified on the EXPLICITBINDPORTRANGE parameter is different from the currently active range for the EXPLICITBINDPORTRANGE pool in the coupling facility, the new range replaces that value.

Changing the starting port (*1st_port*), the number of ports (*num_ports*), or both for the EXPLICITBINDPORTRANGE parameter changes the port numbers in the pool of ports that is guaranteed to be unique across the sysplex for future port allocation

Guidelines:

- Changing the range specified on the EXPLICITBINDPORTRANGE parameter of the GLOBALCONFIG statement affects every stack in the sysplex that has configured a GLOBALCONFIG EXPLICITBINDPORTRANGE value. Future port allocations for all such stacks use the new port range.
- Ports in the EXPLICITBINDPORTRANGE range are usually assigned to a stack in blocks of 64 ports. When expanding the range, use multiples of 64 multiplied by the number of stacks that use a GLOBALCONFIG EXPLICITBINDPORTRANGE configuration.

IQDMULTIWRITE and NOIQDMULTIWRITE

If this parameter is changed with the VARY TCPIP,,OBEYFILE command, the new value does not take effect for any active HiperSockets (iQDIO) interfaces. For a change in this parameter to take effect for an active iQDIO interface, you must stop and restart both the IPv4 and IPv6 interface for the change to be effective.

IQDVLANID

If the IQDVLANID parameter was previously specified and you modify that value, then you must stop and restart the TCP/IP stack for the change to take effect.

MLSCHKTERMINATE

You cannot change the MLSCHKTERMINATE parameter to the NOMLSCHKTERMINATE parameter when the RACF option MLSTABLE is on and the RACF option MLQUIET is off. You can always change the NOMLSCHKTERMINATE parameter to the MLSCHKTERMINATE parameter, but this change is ignored if the value is specified in the data set of a VARY TCPIP,,OBEYFILE command and consistency errors are detected at the same time.

SEGMENTATIONOFFLOAD and NOSEGMENTATIONOFFLOAD

If this parameter is changed with the VARY TCPIP,,OBEYFILE command, the new value does not take effect for any active OSA-Express QDIO interfaces. For a change in these parameters to take effect, all the OSA-Express QDIO interfaces that support TCP segmentation offload must be stopped and restarted.

SMCD and NOSMCD

- If SMC-D support is not enabled, you can specify the SMCD parameter in a VARY TCPIP,,OBEYFILE command data set to activate the support.

  Result : TCP/IP retains knowledge of the last set of SMCD subparameter values that are specified on the GLOBALCONFIG statement, even if GLOBALCONFIG NOSMCD was specified subsequently. If you issue a VARY TCPIP,,OBEYFILE command with GLOBALCONFIG SMCD specified, TCP/IP uses the last saved set of SMCD subparameters, unless new values for the subparameters are coded on the GLOBALCONFIG SMCD statement. Therefore, you can temporarily stop SMC-D processing by issuing a VARY TCPIP,,OBEYFILE command with GLOBALCONFIG NOSMCD specified. Then you can resume SMC-D processing with the previous subparameter settings by issuing a second VARY TCPIP,,OBEYFILE command with just GLOBALCONFIG SMCD specified. Specifying the SMCD parameter also causes the AUTOCACHE function and AUTOSMC monitoring function to be restarted if the SMCGLOBAL AUTOCACHE and AUTOSMC parameters are enabled.
- If SMC-D support is enabled, you can specify the NOSMCD parameter in a VARY TCPIP,,OBEYFILE command data set to deactivate the support.
  - No new TCP connections that use SMC-D processing will be established.
  - Existing TCP connections that use SMC-D will continue to use SMC-D processing.
  - The AUTOCACHE function will be stopped if SMC-R processing is not active.
  - The AUTOSMC monitoring function will be stopped if SMC-R processing is not active.

SMCGLOBAL

AUTOCACHE and NOAUTOCACHE

If you use the VARY TCPIP,,OBEYFILE command to change this parameter from AUTOCACHE to NOAUTOCACHE, the following actions occur:

- Destination IP addresses cached not to use SMC will be deleted from the cache.
- The stack will not cache unsuccessful attempts to create an SMC-R or SMC-D link per destination IP address for new TCP connections.

If you use the VARY TCPIP,,OBEYFILE command to change this parameter from NOAUTOCACHE to AUTOCACHE, the SMCGLOBAL AUTOCACHE function is enabled.

SMCR and NOSMCR

- If SMCR support is not enabled, you can specify the SMCR parameter in a VARY TCPIP,,OBEYFILE command data set to activate the support.

  Result : TCP/IP retains knowledge of the last set of SMCR subparameter values that are specified on the GLOBALCONFIG statement, even if GLOBALCONFIG NOSMCR was specified subsequently. If you issue a VARY TCPIP,,OBEYFILE command with GLOBALCONFIG SMCR specified, TCP/IP uses the saved last set of SMCR subparameters, unless new values for the subparameters are coded on the

GLOBALCONFIG SMCR statement. This allows you to temporarily stop SMC-R processing by issuing a VARY TCPIP,,OBEYFILE command with GLOBALCONFIG NOSMCR specified. Then you can resume SMC-R processing with the previous subparameter settings by issuing a second VARY TCPIP,,OBEYFILE command with just GLOBALCONFIG SMCR specified.Specifying the SMCR parameter also causes the AUTOCACHE function and AUTOSMC monitoring function to be restarted if the SMCGLOBAL AUTOCACHE and AUTOSMC parameters are enabled.

- If SMCR support is enabled, you can specify the NOSMCR parameter in a VARY TCPIP,,OBEYFILE command data set to deactivate the support.
  - No new TCP connections that use SMC-R processing will be established.
  - Existing TCP connections that use SMC-R will continue to use SMC-R processing.
  - The AUTOCACHE function will be stopped if SMC-D processing is not active.
  - The AUTOSMC monitoring function will be stopped if SMC-D processing is not active.
- You cannot change the SMCR PFID parameter values that are currently configured when the associated "RoCE Express" interfaces are active. To change the SMCR PFID parameter values that are currently configured, you must perform the following steps in order:
  1. Stop the associated "RoCE Express" interface.
  2. Issue the VARY TCPIP,,OBEYFILE command with the new PFID values that are coded in the command data set. The new PFID values replace the existing PFID values.
- To add PFID values when you have one or more PFID values coded, you must specify the existing PFID values and the additional PFID values on the SMCR parameter in the VARY TCPIP,,OBEYFILE command data set. Existing PFID values and any existing "RoCE Express" interfaces are not affected.

SYSPLEXMONITOR

AUTOREJOIN and NOAUTOREJOIN

If you change NOAUTOREJOIN to AUTOREJOIN after the stack has left the sysplex and before the problem that caused it to leave has been relieved, the stack automatically rejoins the sysplex group when the problem is relieved. However, if you change NOAUTOREJOIN to AUTOREJOIN after the problem that caused the stack to leave the group has been relieved, you must issue a VARY TCPIP,,SYSPLEX,JOINGROUP command to cause the stack to rejoin the sysplex.

DELAYJOIN and NODELAYJOIN

Changing from DELAYJOIN to NODELAYJOIN while the TCP/IP stack is in the process of delaying joining the sysplex group because OMPROUTE is not active causes the TCP/IP stack to immediately join the sysplex group.

Changing from NODELAYJOIN to DELAYJOIN has no immediate effect until the TCP/IP stack leaves the sysplex group and then attempts to rejoin while OMPROUTE is not active.

SYSPLEXWLMPOLL

You can change the polling rate for WLM values while the TCP/IP stack is active. In order for the change to be effective, you should change the polling rate on all stacks that participate in sysplex distribution (all active distributing stacks, any backup stacks that might take over distribution, and all target stacks).

WLMPRIORITYQ

If you specify WLMPRIORITYQ with the VARY TCPIP,,OBEYFILE command, the IOPRI*n* values are changed to the values specified for the *default_control_values* variable. The new values take effect immediately for all workloads influenced by this function.

WLMPRIORITYQ IOPRIn *control_values*

If you specify this parameter with the VARY TCPIP,,OBEYFILE command, and you do not specify all the control values, the QDIO priority 4 is assigned to packets associated with all control values omitted. The new values immediately take effect for all workloads influenced by this function.

Rule: You cannot modify individual IOPRI*n* control values. If you attempt to modify IOPRI*n* control values, but you specify only those control values that you want to modify, then the QDIO priority 4 is assigned to packets that are associated with any control values that you omitted.

XCFGRPID

For a change in this parameter to take effect, you must stop and restart the TCP/IP stack.

ZERT|NOZERT

If you use the VARY TCPIP,,OBEYFILE command to change INTVAL, zERT aggregation will flush the existing records and begin for new TCP and Enterprise Extender connections, with a new INTVAL. If not specified, the default value for INTVAL is SMF (which is the SMF interval time).

If you use the VARY TCPIP,,OBEYFILE command to change SYNCVAL, zERT aggregation will flush the existing records and will begin for new TCP and Enterprise Extender connections, with a new SYNCVAL (INTVAL always has to be specified since SYNCVAL is a sub-parameter of INTVAL). If not specified, the default time for SYNCVAL is 00:00 which is midnight.

If you use the VARY TCPIP,,OBEYFILE command to change this parameter from ZERT to NOZERT then all ZERT discovery and aggregation functions stop. In addition, all zERT SMF recording stops and all collected data is discarded.

If you use the VARY TCPIP,,OBEYFILE command to change this parameter from NOZERT to ZERT then zERT discovery will begin for new TCP and Enterprise Extender connections only. TCP and Enterprise Extender connections that existed before zERT discovery was enabled will not be monitored. The zERT aggregation function remains disabled.

If you use the VARY TCPIP,,OBEYFILE command to change this parameter from NOZERT to ZERT AGGREGATION, then both the zERT discovery and aggregation functions will begin for new TCP and Enterprise Extender connections only. TCP and Enterprise Extender connections that existed before zERT was enabled will not be monitored, nor will they be included in the aggregation function statistics.

If you use the VARY TCPIP,,OBEFYFILE command to change the subparameter from ZERT NOAGGREGATION to ZERT AGGREGATION, then zERT aggregation will begin for new TCP and Enterprise Extender connections only. TCP and Enterprise Extender connections that existed before zERT aggregation was enabled will not be included in the summarized data. The zERT discovery function is not affected.

If you use the VARY TCPIP,,OBEYFILE command to change the subparameter from ZERT AGGREGATION to ZERT NOAGGREGATION, then the zERT aggregation function stops, although zERT discovery continues. All zERT summary information is discarded and one final set of zERT summary records is generated.

Examples

This example shows the use of the SYSPLEXMONITOR parameter on the GLOBALCONFIG statement that enables many of the sysplex autonomics functions:

```
GLOBALCONFIG SYSPLEXMONITOR AUTOREJOIN DELAYJOIN MONINTERFACE DYNROUTE RECOVERY
```

The following example shows the use of the EXPLICITBINDPORTRANGE parameter to define 1024 ports in the range 5000 - 6023. The ports are used for explicit binds to the IPv4 INADDR_ANY address, or to the IPv6 unspecified address (in6addr_any), and port 0:

```
GLOBALCONFIG EXPLICITBINDPORTRANGE 5000 1024
```

The following examples show the use of the SMCR parameter to define two "RoCE Express" features that use PFID values 0018 and 0019 and port numbers 1 and 2, and to limit the stack to 500 megabytes of 64-bit storage for SMC-R communications. The first example represents 10 GbE RoCE Express features and the second example represents RoCE Express2 features.

```
GLOBALCONFIG SMCR PFID 0018 PORTNUM 1 PFID 0019 PORTNUM 2 FIXEDMEMORY 500
```

```
GLOBALCONFIG SMCR PFID 0018 PFID 0019 FIXEDMEMORY 500
```

The following example shows the use of the SMCD parameter to enable SMC-D support and limit the stack to 500 megabytes of 64-bit storage for SMC-D communications.

```
GLOBALCONFIG SMCD FIXEDMEMORY 500
```

Related topics

- SMFCONFIG statement
- For more information about TCP/IP networking in a multilevel-secure environment, see the security information in z/OS Communications Server: IP Configuration Guide.

Chapter 4. IP System Administrator's Commands

DISPLAY TCPIP,,STOR

Use the DISPLAY TCPIP,*procname*,STOR command to display TCP/IP storage usage information. You can use this command to verify the load module service level.

To verify load module service level, ensure that the eyecatcher for the module matches the latest PTF service for the module. When you contact IBM Service, you can use this command to verify that you are running on the correct TCP/IP service level.

Example

To display TCP/IP storage usage, issue the following command:

```
d tcpip,tcpip2,stor
EZZ8453I TCPIP STORAGE
EZZ8454I TCPIP2   STORAGE            CURRENT   MAXIMUM     LIMIT
EZD2018I 31-BIT
EZZ8455I          ECSA                45654K    56823K   204800K
EZZ8455I          PRIVATE            124634K   143743K   524288K
EZZ8455I          ECSA MODULES         8702K     8702K   NOLIMIT
EZD2018I 64-BIT
EZZ8455I          HVCOMMON               3M        3M   NOLIMIT
EZZ8455I          HVPRIVATE             50M       50M   NOLIMIT
EZZ8455I          TRACE HVCOMMON      2578M     2578M     2578M
EZZ8455I          ZERTAGG HVPRIVATE     56K     1024K   NOLIMIT
EZZ8455I          SMC-R FIXEDMEMORY     12M       16M       40M
EZD2024I            SMC-R SEND MEMORY    4M        4M
EZD2024I            SMC-R RECV MEMORY    8M       12M
EZZ8455I          SMC-D FIXEDMEMORY     12M       16M       40M
EZZ8459I DISPLAY TCPIP STOR COMPLETED SUCCESSFULLY
```

Usage

- If a module is built into multiple load modules, each occurrence is displayed.
- The storage display command is used to verify the load module service level of the TCP/IP stack. The command supports several, but not all, modules within the product.
- ZERTAGG memory information (message EZZ8455I) is included only when the zERT aggregation function is enabled. The zERT aggregation function is enabled by using the ZERT AGGREGATION parameter of the GLOBALCONFIG statement.
- SMC-R memory information (messages EZZ8455I and EZD2024I) is included only when the Shared Memory Communications over Remote Direct Memory Access (SMC-R) function is or was enabled on this TCP/IP stack. The SMC-R function is enabled by using the SMCR parameter of the GLOBALCONFIG statement.
- SMC-D memory information (messages EZZ8455I) is included only when the Shared Memory Communications - Direct Memory Access (SMC-D) function is or was enabled on this TCP/IP stack. The SMC-D function is enabled by using the SMCD parameter of the GLOBALCONFIG statement.

Report field descriptions

- **TCP Configuration Table**

  Display the following configured TCP information that is defined in the TCPCONFIG and SOMAXCONN profile statements. For more information about each field, see the TCPCONFIG or SOMAXCONN profile statement information in z/OS Communications Server: IP Configuration Reference.

DefaultRcvBufSize
The TCP receive buffer size that was defined using the TCPRCVBUFRSIZE parameter in the TCPCONFIG statement. The size is between 256 and TCPMAXRCVBUFRSIZE; the default size is 65536 (64 KB). This value is used as the default receive buffer size for those applications that do not explicitly set the buffer size using SETSOCKOPT(). If the TCPRCVBUFRSIZE parameter was not specified in the TCPCONFIG statement, then the default size 65536 (64 KB) is displayed.

DefaultSndBufSize
The TCP send buffer size that was defined using the TCPSENDBFRSIZE parameter in the TCPCONFIG statement. The size is between 256 bytes and TCPMAXSENDBUFRSIZE; the default size is 65536 (64 KB). This value is used as the default send buffer size for those applications that do not explicitly set the buffer size using SETSOCKOPT(). If the TCPSENDBFRSIZE parameter was not specified in the TCPCONFIG statement, then the default size 65536 (64 KB) is displayed.

DefltMaxRcvBufSize
The TCP maximum receive buffer size that was defined using the TCPMAXRCVBUFRSIZE parameter in the TCPCONFIG statement. The maximum receive buffer size is the maximum value that an application can set as its receive buffer size using SETSOCKOPT(). The minimum acceptable value is the value that is coded on the TCPRCVBUFRSIZE parameter, the maximum size is 2 MB, and the default size is 256 KB. If you do not have large bandwidth interfaces, you can use this parameter to limit the receive buffer size that an application can set. If the TCPMAXRCVBUFRSIZE parameter was not specified in the TCPCONFIG statement, then the default size 262144 (256 KB) is displayed.

SoMaxConn
The maximum number of connection requests that can be queued for any listening socket, as defined by the SOMAXCONN statement. The minimum value is 1, the maximum value is 2147483647, and the default value is 1024.

MaxReTransmitTime
The maximum retransmit interval in seconds. The range is 0 - 999.990. The default value is 120.

Rules :

– If none of the following parameters is specified, this MAXIMUMRETRANSMITTIME parameter is used and the MINIMUMRETRANSMITTIME parameters of the following statements are not used.
  - MAXIMUMRETRANSMITTIME on the BEGINROUTES statement
  - MAXIMUMRETRANSMITTIME on the GATEWAY statement
  - MAXIMUMRETRANSMITTIME on the ROUTETABLE statement
  - Max_Xmit_Time on the OSPF_INTERFACE statement
  - Max_Xmit_Time on the RIP_INTERFACE statement
– The TCPCONFIG version is used if no route parameter has been explicitly specified. If the TCPCONFIG version of maximum retransmit time is used, the MINIMUMRETRANSMITTIME value that is specified on the route parameter is not used, which means the value of the minimum retransmit time is 0.

DefaultKeepAlive
The default keepalive interval that was defined using the INTERVAL parameter in the TCPCONFIG statement. It is the number of minutes that TCP waits after it receives a packet for a connection before it sends a keepalive packet for that connection. The range is 0 - 35791 minutes; the default value is 120. The value 0 disables the keepalive function. If the INTERVAL parameter was not specified in the TCPCONFIG statement, then the default interval 120 is displayed.

DelayAck
Indicates whether the DELAYACKS option is enabled or disabled. The value Yes indicates that acknowledgments are delayed when a packet is received (the DELAYACKS parameter was defined in the TCPCONFIG profile statement or is in effect by default); the value No indicates that acknowledgments are not delayed when a packet is received (the NODELAYACKS parameter was defined in the TCPCONFIG statement).

RestrictLowPort
Indicates whether ports in the range 1 - 1023 are reserved for users by the PORT and PORTRANGE statements. The value Yes indicates that RESTRICTLOWPORTS is in effect (the RESTRICTLOWPORTS parameter was defined in the TCPCONFIG profile statement); the value No indicates that RESTRICTLOWPORTS is not in effect (the UNRESTRICTLOWPORTS parameter was defined in the TCPCONFIG statement or is in effect by default).

SendGarbage
Indicates whether the keepalive packets sent by TCP contain 1 byte of random data. The value Yes indicates that SENDGARBAGE TRUE is in effect (SENDGARBAGE TRUE was defined in the TCPCONFIG profile statement); the value No indicates that SENDGARBAGE TRUE is not in effect (SENDGARBAGE FALSE was defined in the TCPCONFIG statement or is in effect by default).

TcpTimeStamp
Indicates whether the TCP Timestamp Option is enabled or disabled. The value Yes indicates that TCPTIMESTAMP is in effect (the TCPTIMESTAMP parameter was defined in the TCPCONFIG profile statement or is in effect by default); the value No indicates that TCPTIMESTAMP is not in effect (the NOTCPTIMESTAMP parameter was defined in the TCPCONFIG statement).

FinWait2Time
The FinWait2Time number that was defined using the FINWAIT2TIME parameter in the TCPCONFIG statement. It is the number of seconds a TCP connection should remain in the FINWAIT2 state. The range is 60 - 3600 seconds; the default value is 600 seconds. When this timer expires, it is reset to 75 seconds; when this timer expires a second time, the connection is dropped. If the FINWAIT2TIME parameter was not specified in the TCPCONFIG statement, then the default value 600 is displayed.

TimeWaitInterval
The number of seconds that a connection remains in TIMEWAIT state. The range is 0 - 120. The default value is 60.

Note : For local connections, a TIMEWAITINTERVAL of 50 milliseconds is always used.

DefltMaxSndBufSize
The maximum send buffer size. The range is the value that is specified on TCPSENDBFRSIZE to 2 MB. The default value is 256K.

RetransmitAttempt
The number of times a segment is retransmitted before the connection is aborted. The range is 0 - 15. The default value is 15.

ConnectTimeOut
The total amount of time before the initial connection times out. This value also applies to TCP connections that are established over SMC-R links. The range is 5 - 190 seconds. The default value is 75.

ConnectInitIntval
The initial retransmission interval for the connect(). The range is 100 to 3000 milliseconds (ms). The default value is 3000.

KAProbeInterval
The interval in seconds between keepalive probes. The range is 1 - 75. The default value is 75.

This parameter does not change the initial keepalive timeout interval. It controls the time between the probes that are sent only after the initial keepalive interval has expired.

You can specify setsockopt() TCP_KEEPALIVE to override the parameter.

KeepAliveProbes
The number of keepalive probes to send before the connection is aborted. The range is 1 - 10. The default value is 10.

This parameter does not change the initial keepalive timeout interval. It controls the number of probes that are sent only after the initial keepalive interval has expired.

You can specify setsockopt() TCP_KEEPALIVE to override this parameter.

Nagle
Indicates whether the Nagle option is enabled or disabled. The value Yes indicates that packets with less than a full maximum segment size (MSS) of data are buffered unless all data on the send queue has been acknowledged.

QueuedRTT
The threshold at which outbound serialization is engaged. The range is 0 - 50 milliseconds. The default value is 20 milliseconds.

FRRTheshold
The threshold of duplicate ACKs for FRR to engage. The range is 1 - 2048. The default value is 3.

TTLS
Indicates whether Application Transparent Transport Layer Security (AT-TLS) is active in the TCP/IP stack. The value Yes indicates that AT-TLS is active (the TTLS parameter was specified in the TCPCONFIG profile statement). The value No indicates that AT-TLS is not active (the NOTTLS parameter was specified in the TCPCONFIG profile statement or is in effect by default).

EphemeralPorts
The range of ephemeral ports that was defined using the EPHEMERALPORTS parameter in the TCPCONFIG statement or by default. The range specified must be within the range of 1024 to 65535. If the EPHEMERALPORTS parameter was not specified in the TCPCONFIG statement, then the default range 1024 - 65535 is displayed.

SelectiveACK
Indicates whether Selective Acknowledgment (SACK) support is active in the TCP/IP stack. This field can have the following values:

Yes
Indicates that SACK options are exchanged with partners when transmitting data. The SELECTIVEACK parameter was specified on the TCPCONFIG profile statement.

No
Indicates that SACK options will not be exchanged. The NOSELECTIVEACK parameter was specified on the TCPCONFIG profile statement or is in effect by default.

Note: The values displayed in the MaxReTransmitTime, MinReTransmitTime, RoundTripGain, VarianceGain, VarianceMultiplier, and MaxSegLifeTime fields are actual default values that are assigned by the TCP/IP stack; you cannot configure them externally using the TCPCONFIG profile statement. You can override the MaxReTransmitTime, MinReTransmitTime, RoundTripGain, VarianceGain, VarianceMultiplier values on a per-destination basis using either the BEGINROUTES configuration statement, the old GATEWAY configuration statement, or the configuration file for OMPROUTE.

- **UDP Configuration Table**

Display the following configured UDP information defined in the UDPCONFIG profile statement. For more information about each UDP parameter, see UDPCONFIG profile statement information in the z/OS Communications Server: IP Configuration Reference.

DefaultRcvBufSize
The UDP receive buffer size that was defined using the UDPRCVBUFRSIZE parameter in the UDPCONFIG statement. The size is in the range 1 - 65535; the default size is 65535. If the UDPRCVBUFRSIZE parameter was not specified in the UDPCONFIG statement, then the default size 65535 is displayed.

DefaultSndBufSize
The UDP send buffer size that was defined using the UDPSENDBFRSIZE parameter in the UDPCONFIG statement. The size is in the range 1 - 65535; the default size is 65535. If the UDPSENDBFRSIZE parameter was not specified in the UDPCONFIG statement, then the default size 65535 is displayed.

CheckSum
Indicates whether UDP does check summing. The value Yes indicates that UDP check summing is in effect (the UDPCHKSUM parameter was defined in the UDPCONFIG profile statement or is in effect

by default); the value No indicates that UDP check summing is not in effect (the NOUDPCHKSUM parameter was defined in the UDPCONFIG statement).

EphemeralPorts
The range of ephemeral ports that was defined using the EPHEMERALPORTS parameter in the UDPCONFIG statement or by default. The range specified must be within the range of 1024 to 65535. If the EPHEMERALPORTS parameter was not specified in the UDPCONFIG statement, then the default range 1024 - 65535 is displayed.

RestrictLowPort
Indicates whether ports 1 - 1023 are reserved for users by the PORT and PORTRANGE statements. The value Yes indicates that ports in the range 1 - 1023 are reserved (the RESTRICTLOWPORTS parameter was defined in the UDPCONFIG profile statement); the value No indicates that the ports are not reserved (the UNRESTRICTLOWPORTS parameter was defined in the UDPCONFIG statement or is in effect by default).

UdpQueueLimit
Indicates whether UDP should have a queue limit on incoming datagrams. The value Yes indicates that there is a UDP queue limit in effect (the UDPQUEUELIMIT parameter was defined in the UDPCONFIG profile statement or is in effect by default); the value No indicates that a UDP queue limit is not in effect (the NOUDPQUEUELIMIT parameter was defined in the UDPCONFIG statement).

- **IP Configuration Table**

Displays the following configured IP information defined in the IPCONFIG profile statement. For more information about each IP parameter, see the IPCONFIG profile statement information in the z/OS Communications Server: IP Configuration Reference.

Forwarding
Indicates whether the transfer of data between networks is enabled for this TCP/IP stack. Possible values are:

Pkt
Indicates that packets that are received but not destined for this stack are forwarded and use multipath routes if they are available on a per-packet basis (the DATAGRAMFWD FWDMULTIPATH PERPACKET was specified in the IPCONFIG profile statement).

Yes
Indicates that packets that are received but not destined for this stack are forwarded but do not use multipath routes even if they are available. (the DATAGRAMFWD NOFWDMULTIPATH was specified in the IPCONFIG profile statement or is in effect by default).

No
Indicates that packets that are received but that are not destined for this stack are not forwarded in route to the destination (the NODATAGRAMFWD parameter was specified in the IPCONFIG profile statement).

TimeToLive
The time to live value that was defined using the TTL parameter in the IPCONFIG statement. The time to live value is the number of hops that packets originating from this host can travel before reaching the destination. Valid values are in the range 1 - 255; the default value is 64. If the TTL parameter was not specified in the IPCONFIG statement, then the default value 64 is displayed.

RsmTimeOut
The reassembly timeout value that was defined using the REASSEMBLYTIMEOUT parameter in the IPCONFIG statement. It is the amount of time (in seconds) that is allowed to receive all parts of a fragmented packet before discarding the packets received. Valid values are in the range 1 - 240; the default value is 60. If the REASSEMBLYTIMEOUT parameter was not specified in the IPCONFIG statement, then the default value 60 is displayed.

IpSecurity
Indicates whether the IP filtering and IPSec tunnel support is enabled. The value Yes indicates that IP security is in effect (the IPSECURITY parameter was defined on the IPCONFIG profile statement). The value No indicates that IP security is not in effect.

ArpTimeout
The ARP timeout value that was defined using the ARPTO parameter in the IPCONFIG statement. It indicates the number of seconds between creation or revalidation and deletion of ARP table entries. Valid values are in the range 60 - 86400; the default value is 1200. If the ARPTO parameter was not specified in the IPCONFIG statement, then the default value 1200 is displayed.

MaxRsmSize
The maximum packet size that can be reassembled. If an IP datagram is fragmented into smaller packets, the complete reassembled datagram cannot exceed this value. Valid values are in the range 576 - 65535; the default value is 65535.

Restriction: The value that is displayed in the MaxRsmSize field is the actual default value that was assigned by the TCP/IP stack; users cannot configure this value externally using the IPCONFIG profile statement.

Format
The stack-wide command format that was defined using the FORMAT parameter in the IPCONFIG statement or that was assigned by default by TCP/IP stack. This field can have the following values:

SHORT
Indicates that the command report is displayed in the short format (the FORMAT SHORT parameter was specified in the IPCONFIG profile statement).

LONG
Indicates that the command report is displayed in the long format (the FORMAT LONG parameter was specified in the IPCONFIG profile statement).

If the FORMAT parameter was not specified in the IPCONFIG profile statement, then the TCP/IP stack assigned the default format based on whether the stack was IPv6 enabled or not. If the stack is IPv6 enabled, then the format value LONG is assigned by default. If the stack is configured for IPv4-only operation, then the format value SHORT is assigned by default. You can override the stack-wide command format using the Netstat FORMAT/`-M` option.

IgRedirect
Indicates whether TCP/IP is to ignore ICMP Redirect packets. This field can have the following values:

Yes
Indicates that IGNOREREDIRECT is in effect. The IGNOREREDIRECT parameter was defined on the IPCONFIG profile statement, OMPROUTE has been started and IPv4 interfaces are configured to OMPROUTE, or intrusion detection services (IDS) policy is in effect to detect and discard ICMP Redirects.

No
Indicates that ICMP Redirects are not ignored.

SysplxRout
Indicates whether this TCP/IP host is part of an MVS sysplex domain and should communicate interface changes to the workload manager (WLM). This field can have the following values:

Yes
Indicates that SYSPLEXROUTING is in effect (the SYSPLEXROUTING parameter was specified in the IPCONFIG profile statement).

No
Indicates that SYSPLEXROUTING is not in effect (the NOSYSPLEXROUTING parameter was specified in the IPCONFIG profile statement or is in effect by default).

DoubleNop
Indicates whether to force channel programs for CLAW devices to have two NOP CCWs to end the channel programs. This field can have the following values:

Yes
Indicates that CLAWUSEDOUBLENOP is in effect (the CLAWUSEDOUBLENOP parameter was defined on the IPCONFIG profile statement).

No
Indicates that CLAWUSEDOUBLENOP is not in effect.

StopClawEr
Indicates whether to stop channel programs (HALTIO and HALTSIO) when a device error is detected. This field can have the following values:

Yes
Indicates that STOPONCLAWERROR is in effect (the STOPONCLAWERROR parameter was specified in the IPCONFIG profile statement).

No
Indicates that STOPONCLAWERROR is not in effect.

SourceVipa
Indicates whether the TCP/IP stack uses the corresponding virtual IP address in the HOME list as the source IP address for outbound datagrams that do not have an explicit source address. This field can have the following values:

Yes
Indicates that SOURCEVIPA is in effect (the SOURCEVIPA parameter was specified in the IPCONFIG profile statement).

No
Indicates that SOURCEVIPA is not in effect (the NOSOURCEVIPA parameter was specified in the IPCONFIG profile statement or is in effect by default).

MultiPath
Indicates whether the multipath routing selection algorithm for outbound IP traffic is enabled for this TCP/IP stack. Possible values are:

Pkt
Indicates that outbound traffic uses the multipath routes in a round-robin fashion for each outbound packet (the MULTIPATH PERPACKET parameter was specified in the IPCONFIG profile statement).

Conn
Indicates that outbound traffic uses the multipath routes in a round-robin fashion for each outbound connection request (the MULTIPATH PERCONNECTION parameter was specified in the IPCONFIG profile statement).

No
Indicates that outbound traffic always uses the first active route in a multipath group (the NOMULTIPATH parameter was specified in the IPCONFIG profile statement or is in effect by default).

PathMtuDsc
Indicates whether TCP/IP is to dynamically discover the PMTU, which is the smallest MTU of all the hops in the path. This field can have the following values:

Yes
Indicates that PATHMTUDISCOVERY is in effect (the PATHMTUDISCOVERY parameter was specified in the IPCONFIG profile statement),

No
Indicates that PATHMTUDISCOVERY is not in effect (the NOPATHMTUDISCOVERY parameter was specified in the IPCONFIG profile statement or is in effect by default).

DevRtryDur
The retry period duration (in seconds) for a failed device or interface that was defined using the DEVRETRYDURATION parameter in the IPCONFIG statement. TCP/IP performs reactivation attempts at 30 second intervals during this retry period. The default value is 90 seconds. The value 0 indicates an infinite recovery period; reactivation attempts are performed until the device or interface is either successfully reactivated or manually stopped. The maximum value is 4294967295. If the DEVRETRYDURATION parameter was not specified in the IPCONFIG statement, then the default value 90 is displayed.

DynamicXCF
Indicates whether IPv4 XCF dynamic support is enabled for this TCP/IP stack. This field can have the following values:

Yes
Indicates that XCF dynamic support is in effect (the DYNAMICXCF parameter was specified in the IPCONFIG profile statement).

No
Indicates that XCF dynamic support is not in effect (the NODYNAMICXCF parameter was specified in the IPCONFIG profile statement or is in effect by default).

When XCF dynamic support is in effect, the following information is displayed:

IpAddr
The IPv4 address that was specified for DYNAMICXCF in the IPCONFIG profile statement.

Subnet
The subnet mask that was specified for DYNAMICXCF in the IPCONFIG profile statement.

Guidelines:

1. If the IpAddr/PrefixLen format was used for DYNAMICXCF in the IPCONFIG profile statement, then it is displayed in the same format in the Netstat report. The PrefixLen is the integer value in the range 1 - 32 that represents the number of left-most significant bits for the address mask.
2. If the IPv6_address/prefix_route_len format was used for DYNAMICXCF in the IPCONFIG6 profile statement, then it is displayed in the same format in the Netstat report. The length of routing prefix is an integer value in the range 1 - 128.

Metric
The interface routing metric represents the configured cost_metric value to be used by dynamic routing daemons for routing preferences. It is configured using the cost_metric value in the IPCONFIG DYNAMICXCF statement.

SecClass
Indicates the IP Security security class value that is associated with the dynamic XCF link. Valid values are in the range 1 - 255.

SMCD
Indicates whether the HiperSockets interface that dynamic XCF generates supports Shared Memory Communications - Direct Memory Access (SMC-D). This field can have the following values:

Yes
Indicates that the HiperSockets interface that dynamic XCF generates can be used for new TCP connections with SMC-D. The SMCD parameter was specified on the IPCONFIG profile statement or the value was set by default.

No
Indicates that the HiperSockets interface that dynamic XCF generates cannot be used for new TCP connections with SMC-D. The NOSMCD parameter was specified on the IPCONFIG profile statement.

SrcVipaInt
The source VIPA interface name that was defined using the DYNAMICXCF SOURCEVIPAINTERFACE parameter in the IPCONFIG statement. It must be a VIRTUAL interface. This field indicates the value No if the SOURCEVIPAINTERFACE subparameter was not specified for the DYNAMICXCF in the IPCONFIG statement.

QDIOAccel
Indicates whether QDIO Accelerator is enabled for this TCP/IP stack. This field can have the following values:

Yes
Indicates that the QDIO Accelerator is enabled (the QDIOACCELERATOR parameter was specified in the IPCONFIG profile statement).

SD only
Indicates that the QDIO Accelerator is enabled (the QDIOACCELERATOR parameter was specified in the IPCONFIG profile statement), but only for Sysplex Distributor traffic and not for routed traffic. This might be the case if IP forwarding is disabled on this stack, or if IP filters or defensive filters require this stack to perform special processing for routed traffic. For more information, see QDIO Accelerator and IP security in z/OS Communications Server: IP Configuration Guide.

No
Indicates that the QDIO Accelerator is not enabled (the NOQDIOACCELERATOR parameter was specified in the IPCONFIG profile statement or is in effect by default).

QDIOAccelPriority
Indicates which QDIO outbound priority level should be used if the QDIO Accelerator is routing packets to a QDIO device. If the NOQDIOACCELERATOR parameter was specified in the IPCONFIG profile statement or is in effect by default, then the QDIOAccelPriority field is not displayed.

IQDIORoute
Indicates whether HiperSockets Accelerator is enabled for this TCP/IP stack. This field can have the following values:

Yes
Indicates that HiperSockets Accelerator is enabled (the IQDIOROUTING parameter was specified in the IPCONFIG profile statement).

No
Indicates that HiperSockets Accelerator is not enabled (the NOIQDIOROUTING parameter was specified in the IPCONFIG profile statement or is in effect by default).

n/a
Indicates that HiperSockets Accelerator does not apply because QDIO Accelerator is enabled.

QDIOPriority
Indicates which QDIO outbound priority level should be used if the HiperSockets Accelerator is routing packets to a QDIO device. If the NOIQDIOROUTING parameter was specified in the IPCONFIG profile statement or is in effect by default, then the QDIOPriority field is not displayed. This field is displayed only when the IQDIORoute field value is Yes.

TcpStackSrcVipa
The IPv4 address that was defined using the TCPSTACKSOURCEVIPA parameter in the IPCONFIG statement. It must be the source IP address for outbound TCP connections if SOURCEVIPA has been enabled. This field has the value No if the TCPSTACKSOURCEVIPA parameter was not specified in the IPCONFIG statement

ChecksumOffload
Indicates whether the IPv4 checksum offload function is enabled or disabled. This field can have the following values:

Yes
Indicates that the checksum processing for IPv4 packets is offloaded to OSA-Express interfaces that support the checksum offload function. The CHECKSUMOFFLOAD parameter was specified on the IPCONFIG profile statement or the value was set by default.

No
Indicates that the checksum processing is performed by the TCP/IP stack. The NOCHECKSUMOFFLOAD parameter was specified on the IPCONFIG profile statement.

SegOffload
Indicates whether the IPv4 TCP segmentation offload function is enabled or disabled. This field can have the following values:

Yes
Indicates that IPv4 TCP segmentation is performed by OSA-Express interfaces that support the segmentation offload function. The SEGMENTATIONOFFLOAD parameter was specified on the IPCONFIG profile statement.

No
Indicates that the segmentation is performed by the TCP/IP stack. The NOSEGMENTATIONOFFLOAD parameter was specified on the IPCONFIG profile statement or the value was set by default.

- **IPv6 Configuration Table if the TCP/IP stack is IPv6 enabled**

  Displays the following configured IPv6 information that is defined in the IPCONFIG6 profile statement For more information about each IPv6 IP parameter, see the IPCONFIG6 profile statement information in the z/OS Communications Server: IP Configuration Reference.

  Forwarding
  Indicates whether the transfer of data between networks is enabled for this TCP/IP stack. Possible values are:

  Pkt
  Indicates that packets that are received but that are not destined for this stack are forwarded and use multipath routes if available on a per-packet basis (the DATAGRAMFWD FWDMULTIPATH PERPACKET was specified in the IPCONFIG6 profile statement).

  Yes
  Indicates that packets that are received but that are not destined for this stack are forwarded but do not use multipath routes even if they are available. (the DATAGRAMFWD NOFWDMULTIPATH was specified in the IPCONFIG6 profile statement or is in effect by default).

  No
  Indicates that packets that are received but that are not destined for this stack are not forwarded in route to the destination (the NODATAGRAMFWD parameter was specified in the IPCONFIG6 profile statement).

  HopLimit
  The hop limit value that was defined using the HOPLIMIT parameter in the IPCONFIG6 statement. It is the number of hops that a packet that originates at this host can travel in route to the destination. Valid values are in the range 1 - 255; the default value is 255. If the HOPLIMIT parameter was not specified in the IPCONFIG6 statement, then the default value 255 is displayed.

  IgRedirect
  Indicates whether TCP/IP is to ignore ICMP Redirect packets. This field can have the following values:

  Yes
  Indicates that IGNOREREDIRECT is in effect. The IGNOREREDIRECT parameter was defined on the IPCONFIG6 profile statement, OMPROUTE has been started and IPv6 interfaces are configured to OMPROUTE, or intrusion detection services (IDS) policy is in effect to detect and discard ICMP Redirects.

  No
  Indicates that ICMP Redirects are not ignored.

  SourceVipa
  Indicates whether to use a virtual IP address that is assigned to the SOURCEVIPAINT interface as the source address for outbound datagrams that do not have an explicit source address. You must specify the SOURCEVIPAINT parameter on the INTERFACE profile statement for each interface where you want the SOURCEVIPA address to take effect. This field can have the following values:

  Yes
  Indicates that SOURCEVIPA is in effect (the SOURCEVIPA parameter was specified in the IPCONFIG6 profile statement).

No
Indicates that SOURCEVIPA is not in effect (the NOSOURCEVIPA parameter was specified in the IPCONFIG6 profile statement or is in effect by default).

MultiPath
Indicates whether the multipath routing selection algorithm for outbound IP traffic is enabled for this TCP/IP stack. Possible values are:

Pkt
Indicates that outbound traffic uses the multipath routes in a round-robin fashion for each outbound packet (the MULTIPATH PERPACKET parameter was specified in the IPCONFIG6 profile statement).

Conn
Indicates that outbound traffic uses the multipath routes in a round-robin fashion for each outbound connection request (the MULTIPATH PERCONNECTION parameter was specified in the IPCONFIG6 profile statement).

No
Indicates that outbound traffic always uses the first active route in a multipath group (the NOMULTIPATH parameter was specified in the IPCONFIG6 profile statement is in effect by default).

IcmperrLim
The ICMP error limit value that was defined using the ICMPERRORLIMIT parameter in the IPCONFIG6 statement. It controls the rate at which ICMP error messages can be sent to a particular IPv6 destination address. The number displayed is the number of messages per second. Valid values are in the range 1 - 20; the default value is 3. If the ICMPERRORLIMIT parameter was not specified in the IPCONFIG6 statement, then the default value 3 is displayed.

IgRtrHopLimit
Indicates whether the TCP/IP stack ignores a hop limit value that is received from a router in a router advertisement. This field can have the following values:

Yes
Indicates that IGNOREROUTERHOPLIMIT is in effect (the IGNOREROUTERHOPLIMIT parameter was defined on the IPCONFIG6 profile statement).

No
Indicates that IGNOREROUTERHOPLIMIT is not in effect (the NOIGNOREROUTERHOPLIMIT parameter was defined on the IPCONFIG6 profile statement or is in effect by default).

IpSecurity
Indicates whether the IP filtering and IPSec tunnel support is enabled.

Yes
Indicates that IP security is in effect (the IPSECURITY parameter was defined on the IPCONFIG6 profile statement). When IP security is in effect, the following information is displayed:

OSMSecClass
Indicates the IP Security security class value that is associated with the OSM interfaces. Valid values are in the range 1 - 255.

No
Indicates that IP security is not in effect.

DynamicXCF
Indicates whether IPv6 XCF dynamic support is enabled for this TCP/IP stack. This field can have the following values:

Yes
Indicates that XCF dynamic support is in effect (the DYNAMICXCF parameter was specified in the IPCONFIG6 profile statement).

No
Indicates that XCF dynamic support is not in effect (the NODYNAMICXCF parameter was specified in the IPCONFIG6 profile statement or is in effect by default).

When XCF dynamic support is in effect, the following information is displayed:

IpAddr
The IPv6 address that was specified for DYNAMICXCF in the IPCONFIG6 profile statement.

Tip: If the IpAddr/PrefixRouteLen format was used for DYNAMICXCF in the IPCONFIG6 profile statement, then it is displayed in the same format in the Netstat report. The PrefixRouteLen is the integer value in the range 1 - 128.

IntfId
The 64-bit interface identifier in colon-hexadecimal format that was specified using INTFID subparameter for DYNAMICXCF in the IPCONFIG6 profile statement. If the INTFID subparameter was not specified, then this field is not displayed.

SrcVipaInt
The source VIPA interface name that was defined using the DYNAMICXCF SOURCEVIPAINTERFACE parameter in the IPCONFIG6 statement. It must be a VIRTUAL6 interface. This field indicates the value No if the SOURCEVIPAINTERFACE subparameter was not specified for the DYNAMICXCF in the IPCONFIG6 statement.

SecClass
Indicates the IP Security security class value that is associated with the IPv6 dynamic XCF interfaces. Valid values are in the range 1 - 255.

SMCD
Indicates whether the HiperSockets interface that dynamic XCF generates supports SMC-D. This field can have the following values:

Yes
Indicates that the HiperSockets interface that dynamic XCF generates can be used for new TCP connections with SMC-D. The SMCD parameter was specified on the IPCONFIG6 profile statement or the value was set by default.

No
Indicates that the HiperSockets interface that dynamic XCF generates cannot be used for new TCP connections with SMC-D. The NOSMCD parameter was specified on the IPCONFIG6 profile statement.

TcpStackSrcVipa
The IPv6 interface name that was defined using the TCPSTACKSOURCEVIPA parameter in the IPCONFIG6 statement. It must be the source interface for outbound TCP connections if SOURCEVIPA has been enabled. This field indicates the value No if the TCPSTACKSOURCEVIPA parameter was not specified in the IPCONFIG6 statement

TempAddresses
Indicates whether the TCP/IP stack generates IPv6 temporary addresses for IPv6 interfaces for which stateless address autoconfiguration is enabled. This field can have the following values:

Yes
Indicates that this behavior is enabled (the TEMPADDRS parameter was defined on the IPCONFIG6 profile statement).

No
Indicates that this behavior is not enabled (the NOTEMPADDRS parameter was defined on the IPCONFIG6 profile statement or is in effect by default).

When TEMPADDRS support is in effect, the following information is displayed:

PreferredLifetime
The preferred lifetime for IPv6 temporary addresses, which was defined using the PREFLIFETIME parameter in the IPCONFIG6 statement.

At the expiration of the preferred lifetime, a new temporary address is generated and the existing address is deprecated. The number that is displayed is the preferred lifetime, in hours. Valid values are in the range of 1 - 720 hours (30 days). The default value is 24 hours.

ValidLifetime
The valid lifetime for IPv6 temporary addresses that was defined using the VALIDLIFETIME parameter in the IPCONFIG6 statement.

When the valid lifetime expires, the temporary address is deleted. The number displayed is the valid lifetime in hours. Valid values are in the range 2 - 2160 hours (90 days). The default value is 7 times the preferred lifetime value, with a maximum of 90 days.

ChecksumOffload
Indicates whether the IPv6 checksum offload function is enabled or disabled. This field can have the following values:

Yes
Indicates that the checksum processing for IPv6 packets is offloaded to OSA-Express interfaces that support the checksum offload function. The CHECKSUMOFFLOAD parameter was specified on the IPCONFIG6 profile statement or the value was set by default.

No
Indicates that the checksum processing is performed by the TCP/IP stack. The NOCHECKSUMOFFLOAD parameter was specified on the IPCONFIG6 profile statement.

SegOffload
Indicates whether the IPv6 TCP segmentation offload function is enabled or disabled. This field can have the following values:

Yes
Indicates that the IPv6 TCP segmentation is offloaded to OSA-Express interfaces that support the segmentation offload function. The SEGMENTATIONOFFLOAD parameter was specified on the IPCONFIG6 profile statement.

No
Indicates that the segmentation is performed by the TCP/IP stack. The NOSEGMENTATIONOFFLOAD parameter was specified on the IPCONFIG6 profile statement or the value was set by default.

- **SMF parameters**

Display the following configured SMF information defined in the SMFCONFIG profile statement. For more information about each SMF parameter, see SMFCONFIG profile statement information in the z/OS Communications Server: IP Configuration Reference.

Type 118

TcpInit
Indicates whether SMF subtype 1 records are created when TCP connections are established. A value of the subtype indicates that TYPE118 TCPINIT is in effect (the TCPINIT or TYPE118 TCPINIT was specified on the SMFCONFIG profile statement or a nonzero value of inittype was specified on the SMFPARMS profile statement).

The value 0 indicates that TYPE118 TCPINIT is not in effect (the NOTCPINIT or TYPE118 NOTCPINIT was specified in the SMFCONFIG profile statement (or is in effect by default), or zero value of inittype was specified on the SMFPARMS profile statement).

TcpTerm
Indicates whether SMF subtype 2 records are created when TCP connections are established. A value of the subtype indicates that TYPE118 TCPTERM is in effect (the TCPTERM or TYPE118 TCPTERM was specified on the profile SMFCONFIG statement or a non zero value of termtype was specified on the SMFPARMS profile statement).

The value 0 indicates that TYPE118 TCPTERM is not in effect (the NOTCPTERM or TYPE118 NOTCPTERM was specified in the SMFCONFIG profile statement (or is in effect by default), or zero value of termtype was specified on the SMFPARMS profile statement).

FTPClient
Indicates whether SMF subtype 3 records are created when TCP connections are established. A value of the subtype indicates that TYPE118 FTPCLIENT is in effect (the FTPCLIENT or TYPE118 FTPCLIENT was specified on the SMFCONFIG profile statement or a non zero value of clienttype was specified on the SMFPARMS profile statement).

The value 0 indicates that TYPE118 FTPCLIENT is not in effect (the NOFTPCLIENT or TYPE118 NOFTPCLIENT was specified in the SMFCONFIG profile statement (or is in effect by default), or zero value of clienttype was specified on the SMFPARMS profile statement).

TN3270Client
Indicates whether SMF subtype 4 records are created when TCP connections are established. A value of the subtype indicates TYPE118 TN3270CLIENT is in effect (the TN3270CLIENT or TYPE118 TN3270CLIENT was specified on the SMFCONFIG profile statement or a non zero value of clienttype was specified on the SMFPARMS profile statement).

The value 0 indicates that TYPE118 TN3270CLIENT is not in effect (the NOTN3270CLIENT or TYPE118 NOTN3270CLIENT was specified in the SMFCONFIG profile statement (or is in effect by default), or zero value of clienttype was specified on the SMFPARMS profile statement).

TcpIpStates
Indicates whether SMF subtype 5 records are created when TCP connections are established. A value of the subtype indicates that TYPE118 TCPIPSTATISTICS is in effect (the TCPIPSTATISTICS or TYPE118 TCPIPSTATISTICS was specified on the SMFCONFIG statement).

The value 0 indicates that TYPE118 TCPIPSTATISTICS is not in effect (the NOTCPIPSTATISTICS or TYPE118 NOTCPIPSTATISTICS was specified in the SMFCONFIG profile statement or is in effect by default).

Type 119

TcpInit
Indicates whether SMF records of subtype 1 are created when TCP connections are established. This field can have the following values:

Yes
Indicates that TYPE119 TCPINIT is in effect (the TYPE119 TCPINIT was specified on the SMFCONFIG statement).

No
Indicates that TYPE119 TCPINIT is not in effect (the TYPE119 NOTCPINIT was specified in the SMFCONFIG profile statement or is in effect by default).

TcpTerm
Indicates whether SMF subtype 2 records are created when TCP connections are established. This field can have the following values:

Yes
Indicates that TYPE119 TCPTERM is in effect (the TYPE119 TCPTERM was specified on the SMFCONFIG statement).

No
Indicates that TYPE119 TCPTERM is not in effect (the TYPE119 NOTCPTERM was specified in the SMFCONFIG profile statement or is in effect by default).

FTPClient
Indicates whether SMF subtype 3 records are created when TCP connections are established. This field can have the following values:

Yes
Indicates that TYPE119 FTPCLIENT is in effect (the TYPE119 FTPCLIENT was specified on the SMFCONFIG statement).

No
Indicates that TYPE119 FTPCLIENT is not in effect (the TYPE119 NOFTPCLIENT was specified in the SMFCONFIG profile statement or is in effect by default).

TcpIpStats
Indicates whether SMF subtype 5 records are created when TCP connections are established. This field can have the following values:

Yes
Indicates that TYPE119 TCPIPSTATISTICS is in effect (the TYPE119 TCPIPSTATISTICS was specified on the SMFCONFIG statement).

No
Indicates that TYPE119 TCPIPSTATISTICS is not in effect (the TYPE119 NOTCPIPSTATISTICS was specified in the SMFCONFIG profile statement or is in effect by default).

IfStats
Indicates whether SMF subtype 6 and subtype 44 records are created. This field can have the following values:

Yes
Indicates that TYPE119 IFSTATISTICS is in effect (the TYPE119 IFSTATISTICS was specified on the SMFCONFIG statement).

No
Indicates that TYPE119 IFSTATISTICS is not in effect (the TYPE119 NOIFSTATISTICS was specified in the SMFCONFIG profile statement or is in effect by default).

PortStats
Indicates whether SMF subtype 7 records are created when TCP connections are established. This field can have the following values:

Yes
Indicates that TYPE119 PORTSTATISTICS is in effect (the TYPE119 PORTSTATISTICS was specified on the SMFCONFIG statement).

No
Indicates that TYPE119 PORTSTATISTICS is not in effect (the TYPE119 NOPORTSTATISTICS was specified in the SMFCONFIG profile statement or is in effect by default).

Stack
Indicates whether SMF subtype 8 records are created when TCP connections are established. This field can have the following values:

Yes
Indicates that TYPE119 TCPSTACK is in effect (the TYPE119 TCPSTACK was specified on the SMFCONFIG statement).

No
Indicates that TYPE119 TCPSTACK is not in effect (the TYPE119 NOTCPSTACK was specified in the SMFCONFIG profile statement or is in effect by default).

UdpTerm
Indicates whether SMF subtype 10 records are created when TCP connections are established. This field can have the following values:

Yes
Indicates that TYPE119 UDPTERM is in effect (the TYPE119 UDPTERM was specified on the SMFCONFIG statement).

No
Indicates that TYPE119 UDPTERM is not in effect (the TYPE119 NOUDPTERM was specified in the SMFCONFIG profile statement or is in effect by default).

TN3270Client
Indicates whether SMF subtype 22 and 23 records are created when TCP connections are established. This field can have the following values:

Yes
Indicates that TYPE119 TN3270CLIENT is in effect (the TYPE119 TN3270CLIENT was specified on the SMFCONFIG statement).

No
Indicates that TYPE119 TN3270CLIENT is not in effect (the TYPE119 NOTN3270CLIENT was specified in the SMFCONFIG profile statement or is in effect by default).

IPSecurity
Indicates whether SMF records of subtypes 77, 78, 79, and 80 are created when dynamic tunnels are removed and when manual tunnels are activated and deactivated. This field can have the following values:

Yes
Indicates that TYPE119 IPSECURITY is in effect (the TYPE119 IPSECURITY was specified on the SMFCONFIG statement).

No
Indicates that TYPE119 IPSECURITY is not in effect (the TYPE119 NOIPSECURITY was specified or is in effect by default in the SMFCONFIG profile statement).

Profile
Indicates whether SMF subtype 4 event records are created when the TCP/IP stack is initialized or when a profile change occurs. This record provides TCP/IP stack profile information. This field can have the following values:

Yes
Indicates that this behavior is enabled (the TYPE119 PROFILE parameter was specified on the SMFCONFIG statement).

No
Indicates that this behavior is not enabled (the TYPE119 NOPROFILE parameter was specified on the SMFCONFIG statement or is in effect by default).

DVIPA
Indicates whether SMF subtypes 32, 33, 34, 35, 36, and 37 event records are created for sysplex events. These records provide information about changes to dynamic virtual IP addresses (DVIPAs), DVIPA targets, and DVIPA target servers. This field can have the following values:

Yes
Indicates that this behavior is enabled (the TYPE119 DVIPA parameter was specified on the SMFCONFIG statement).

No
Indicates that this behavior is not enabled (the TYPE119 NODVIPA parameter was specified on the SMFCONFIG statement or is in effect by default).

SmcrGrpStats
Indicates whether SMF subtype 41 records are created. These records are SMC-R link group statistics records. The records collect information about Shared Memory Communications over Remote Direct Memory Access (SMC-R) link groups and the SMC-R links within each group. This field can have the following values:

Yes
Indicates that this behavior is enabled. The TYPE119 SMCRGROUPSTATISTICS parameter was specified on the SMFCONFIG statement.

No
Indicates that this behavior is not enabled. The TYPE119 NOSMCRGROUPSTATISTICS parameter was specified on the SMFCONFIG statement or is in effect by default.

SmcrLnkEvent
Indicates whether SMF subtype 42 and 43 records are created. The SMF records of subtype 42 are created when SMC-R links are started, and the SMF records of subtype 43 are created when SMC-R links are ended. This field can have the following values:

Yes
Indicates that this behavior is enabled. The TYPE119 SMCRLINKEVENT parameter was specified on the SMFCONFIG statement.

No
Indicates that this behavior is not enabled. The TYPE119 NOSMCRLINKEVENT parameter was specified on the SMFCONFIG statement or is in effect by default.

SmcdLnkStats
Indicates whether SMF subtype 38 records are created. These records are SMC-D link statistics records. The records collect information about Shared Memory Communications - Direct Memory Access (SMC-D) links. This field can have the following values:

Yes
Indicates that this behavior is enabled. The TYPE119 SMCDLINKSTATISTICS parameter was specified on the SMFCONFIG statement.

No
Indicates that this behavior is not enabled. The TYPE119 NOSMCDLINKSTATISTICS parameter was specified on the SMFCONFIG statement or is in effect by default.

SmcdLnkEvent
Indicates whether SMF subtype 39 and 40 records are created. The SMF records of subtype 39 are created when SMC-D links are started, and the SMF records of subtype 40 are created when SMC-D links are ended. This field can have the following values:

Yes
Indicates that this behavior is enabled. The TYPE119 SMCDLINKEVENT parameter was specified on the SMFCONFIG statement.

No
Indicates that this behavior is not enabled. The TYPE119 NOSMCDLINKEVENT parameter was specified on the SMFCONFIG statement or is in effect by default.

ZertDetail
Indicates whether SMF subtype 11 records are created.

A subtype 11 record is created anytime one of the conditions occurs:

- When new TCP and Enterprise Extender connections are established
- When significant changes to a connection's cryptographic protection state occur
- When a connection terminates

You can also get these records when the z/OS Encryption Readiness Technology (zERT) function is enabled or disabled.

This field can have the following values:

Yes
Indicates that this behavior is enabled. The TYPE119 ZERTDETAIL parameter was specified on the SMFCONFIG statement.

No
Indicates that this behavior is not enabled. The TYPE119 NOZERTDETAIL parameter was specified on the SMFCONFIG statement or is in effect by default.

ZertSummary
Indicates whether SMF subtype 12 records are created. Subtype 12 records are interval records that report zERT summary statistics.

You also get subtype 12 event records when the zERT aggregation function is enabled or disabled.

This field can have the following values:

Yes
Indicates that this behavior is enabled. The TYPE119 ZERTSUMMARY parameter was specified on the SMFCONFIG statement.

No
Indicates that this behavior is not enabled. The TYPE119 NOZERTSUMMARY parameter was specified on the SMFCONFIG statement or is in effect by default.

Note: The TCPIP statistics field under SMF Parameters displays the subtype value used when creating the SMF type 118 record (if the value is nonzero). The TCPIP statistics field under Global Configuration Information indicates whether the TCP/IP stack will write statistics messages to the TCP/IP job log when TCP/IP is terminated. For the Type 119 fields, the subtype cannot be changed and the setting indicates if the record is requested (Yes) or not (No).

- **Global Configuration Information**

  Display the following global configured information defined in the GLOBALCONFIG profile statement. For more information about each global parameter, see GLOBALCONFIG profile statement information in the z/OS Communications Server: IP Configuration Reference.

  TcpIpStats
  Indicates whether the several TCP/IP counter values are to be written to the output data set designated by the CFGPRINT JCL statement. The value Yes indicates that TCPIPSTATISTICS is in effect (the TCPIPSTATISTICS parameter was specified in the GLOBALCONFIG profile statement). The value No indicates that TCPIPSTATISTICS is not in effect (the NOTCPIPSTATISTICS parameter was specified in the GLOBALCONFIG profile statement or is in effect by default).

  Tip: The TCPIPSTATS field that is shown under the SMF PARAMETERS section of the Netstat CONFIG/-f output reflects the TcpIpStatistics value or NoTcpIpStatistics value that is specified on the SMFCONFIG statement in the TCP/IP Profile or Obeyfile. The TCPIPSTATS field that is shown under the GLOBAL CONFIGURATION section of the Netstat CONFIG/-f output reflects the value from the GLOBALCONFIG statement in the TCP/IP Profile or Obeyfile.

  ECSALimit
  The maximum amount of extended common service area (ECSA) that was defined using the ECSALIMIT parameter in the GLOBALCONFIG statement. This limit can be expressed as a number followed by the letter K (which represents 1024 bytes), or a number followed by the letter M (which represents 1048576 bytes). If the K suffix is used, then the value displayed must be in the range 10240K - 2096128K inclusive, or 0K. If the M suffix is used, the value displayed must be in the range 10M - 2047M inclusive, or 0K. If the ECSALIMIT parameter was not specified in the GLOBALCONFIG statement, then the default value 0K is displayed (which means no limit).

  PoolLimit
  The maximum amount of authorized private storage that was defined using the POOLLIMIT parameter in the GLOBALCONFIG statement. This limit can be expressed as a number followed by the letter K (which represents 1024 bytes), or a number followed by the letter M (which represents 1048576 bytes). If the K suffix is used, then the value displayed must be in the range 10240K to 2096128K inclusive, or 0K. If the M suffix is used, value is displayed must be in the range 10M - 2047M inclusive, or 0K. If the POOLLIMIT parameter was not specified in the GLOBALCONFIG statement, then the default value 0K is displayed (which means no limit).

  MlsChkTerm
  Indicates whether the stack should be terminated when inconsistent configuration information is discovered in a multilevel-secure environment. The value Yes indicates that MLSCHKTERMINATE is in effect (the MLSCHKTERMINATE parameter was specified in the GLOBALCONFIG profile statement). The value No indicates that MLSCHKTERMINATE is not in effect (the NOMLSCHKTERMINATE parameter was specified in the GLOBALCONFIG profile statement or is in effect by default).

  XCFGRPID
  Displays the TCP 2-digit XCF group name suffix. The two digits displayed are used to generate the XCF group that the TCP/IP stack has joined. The group name is EZBT*vvtt*, where *vv* is the VTAM XCF group ID suffix (specified as a VTAM start option) and *tt* is the displayed XCFGRPID value. If no

VTAM XCF group ID suffix was specified, the group name is EZBTCP*tt*. You can use the D TCPIP,,SYSPLEX,GROUP command to display the group name that the TCP/IP stack has joined.

These digits are also used as a suffix for the EZBDVIPA and EZBEPORT structure names in the form EZBDVIPA*vvtt* and EZBEPORT*vvtt*. If no VTAM XCF group ID suffix was specified, the structure names are EZBDVIPA01*tt* and EZBEPORT01*tt*. If no XCFGRPID value was specified on the GLOBALCONFIG statement in the TCP/IP profile, then no value is displayed for XCFGRPID field in the Netstat output.

IQDVLANID
Displays the TCP/IP VLAN ID that is to be used when a HiperSockets link or interface is generated for dynamic XCF connectivity between stacks on the same CPC. The VLAN ID provides connectivity separation between TCP/IP stacks using HiperSockets for dynamic XCF when subplexing is being used (when XCFGRPID was specified on the GLOBALCONFIG statement). TCP/IP stacks with the same XCFGRPID value (stacks in the same subplex) should specify the same IQDVLANID value if the stacks are in the same CPC and use the same CHPID value. TCP/IP stacks with different XCFGRPID values should specify different IQDVLANID values if the stacks are in the same CPC and use the same CHPID value. If no IQDVLANID value was specified on the GLOBALCONFIG statement in the TCP/IP profile, then the value 0 (no value) is displayed for the IQDVLANID field in the Netstat output.

SysplexWLMPoll
The rate, in seconds, at which the sysplex distributor and its target servers poll WLM for new weight recommendations. A shorter rate indicates a quicker response; however, shorter rates might result in unneeded queries.

MaxRecs
The maximum number of records that are displayed by the DISPLAY TCPIP,,NETSTAT operator command, if the MAX parameter is not specified on that command. The maximum number of records is specified on the MAXRECS parameter of the GLOBALCONFIG profile statement. An asterisk (*) indicates that all records are displayed.

ExplicitBindPortRange
The range of ephemeral ports that is assigned uniquely across the sysplex when an explicit bind() is issued using INADDR_ANY or the unspecified IPv6 address (in6addr_any) and when the specified port is 0.

Tip: This range is the range that was configured on this stack. It might not be the actual range that is in use throughout the sysplex at this time, because another stack that was started later with a different explicit bind port range configured (or with a VARY OBEYFILE command specifying a file with a different EXPLICITBINDPORTRANGE value) can override the range that is configured by this stack. Use the Display TCPIP,,SYSPLEX,PORTS command to display the currently active port range.

AutoIQDC
Indicates whether to dynamically create Internal Queued Direct I/O (IQD) interfaces and transparently converge the dynamic IQD interfaces with the associated OSA interfaces for OSD CHPIDs. The IQD interface that is dynamically created and managed is logically converged with the OSA interface and is referred to as a HiperSockets Converged Interface (IQDC). This field can have the following values:

No
Do not use dynamic IQD (HiperSockets) converged interfaces. The NOAUTOIQDC parameter was specified on the GLOBALCONFIG statement.

AllTraffic
Indicates to use IQDC interfaces for all eligible outbound traffic flowing on the external IP data network. This value is the default value for the AutoIQDC field.

NoLargeData
Indicates to not use IQDC dynamic interfaces for outbound TCP socket data transmissions of length 32 KB or larger. Use dynamic IQDC interfaces for all other eligible outbound traffic.

AutoIQDX
Indicates whether dynamic Internal Queued Direct I/O extensions function (IQDX) interfaces are used for connectivity to the intraensemble data network (IEDN). This field can have the following values:

No
Indicates that access to the IEDN using HiperSockets (IQD CHPIDs) with the IQDX is disabled. The NOAUTOIQDX parameter was specified on the GLOBALCONFIG statement.

AllTraffic
Indicates that IQDX interfaces are used for all eligible outbound traffic to the IEDN. The AUTOIQDX ALLTRAFFIC parameter was specified on the GLOBALCONFIG statement. This value is the default value for the AutoIQDX field.

NoLargeData
Indicates that IQDX interfaces are used for all eligible outbound traffic to the IEDN, except for large outbound TCP protocol traffic. The AUTOIQDX NOLARGEDATA parameter was specified on the GLOBALCONFIG statement. Large outbound TCP traffic is sent to the IEDN by using OSX OSA-Express interfaces.

IQDMultiWrite
Indicates whether all HiperSockets interfaces are configured to move multiple output data buffers using a single write operation. You must stop and restart the interface for a change in this value to take effect for an active HiperSockets interface. This field can have the following values:

Yes
Indicates that the HiperSockets interfaces are configured to use HiperSockets multiple write support when this function is supported by the IBM Z®® environment (the IQDMULTIWRITE parameter was specified on the GLOBALCONFIG profile statement).

No
Indicates that the HiperSockets interfaces are not configured to use HiperSockets multiple write support (the NOIQDMULTIWRITE parameter was specified on the GLOBALCONFIG profile statement or the value was set by default).

WLMPriorityQ
Indicates whether OSA-Express QDIO write priority values are being assigned to outbound OSA-Express packets that are associated with Workload Manager (WLM) service classes, and to forwarded packets that are not being accelerated. The displayed priorities are applied only when the IPv4 type of service (ToS) byte or the IPv6 traffic class value in the IP header is 0 and the packet is sent from an OSA-Express device that is in QDIO mode. This field can have the following values:

Yes
Indicates that QDIO write priority values are assigned to outbound OSA-Express packets that are associated with Workload Manager (WLM) service classes, and to forwarded packets that are not being accelerated (the WLMPRIORITYQ parameter was specified on the GLOBALCONFIG profile statement). When the WLMPriorityQ field has the value Yes, the following information is displayed:

IOPRIn control_values
Indicates which QDIO priority value is assigned to each control value. The QDIO priority values are in the range of 1 - 4. These QDIO priority values are displayed as the identifiers IOPRI1, IOPRI2, IOPRI3, and IOPRI4. The values that follow the identifiers are the control values. The control values represent Workload Manager service classes and forwarded packets. Most of the control values correlate directly to Workload Manager service class importance levels. See the WLMPRIORITYQ parameter in the GLOBALCONFIG profile statement information in z/OS Communications Server: IP Configuration Reference for more details about the control values. If no control value was specified for a specific QDIO priority value, then the identifier for that QDIO priority value is not displayed.

No
Indicates that QDIO write priority values are not assigned to outbound OSA-Express packets that are associated with Workload Manager (WLM) service classes or to forwarded packets that

are not accelerated (the NOWLMPRIORITYQ parameter was specified on the GLOBALCONFIG profile statement or is in effect by default).

Sysplex Monitor

Displays the parameter values for the Sysplex Problem Detection and Recovery function.

TimerSecs

Displays the timer value (in seconds) that is used to determine how soon the sysplex monitor timer reacts to problems with needed sysplex resources. This value can be configured using the SYSPLEXMONITOR parameter on the GLOBALCONFIG profile statement. Valid values are in the range 10 - 3600 seconds; the default value is 60 seconds.

Recovery

Indicates the action that is to be taken when a sysplex problem is detected.

The value Yes indicates that when a problem is detected, the stack issues messages about the problem, leaves the sysplex group, and deactivates all DVIPA resources that are owned by this stack; the VIPADYNAMIC configuration is restored if the stack rejoins the sysplex group. The default value is No. The value Yes can be configured by specifying the RECOVERY keyword for the SYSPLEXMONITOR parameter on the GLOBALCONFIG profile statement.

The value No indicates that when a problem is detected, the stack issues messages regarding the problem but takes no other action. The value No can be configured by specifying the NORECOVERY keyword for the SYSPLEXMONITOR parameter on the GLOBALCONFIG profile statement.

DelayJoin

Indicates whether the TCP/IP stack delays joining the sysplex group during stack initialization or rejoining the sysplex group following a VARY TCPIP,,OBEYFILE command.

The value No indicates that TCP/IP immediately joins the sysplex group during stack initialization. The default value is No and can be configured by specifying the NODELAYJOIN keyword for the SYSPLEXMONITOR parameter on the GLOBALCONFIG profile statement.

The value Yes indicates that TCP/IP delays joining the sysplex group during stack initialization until the following conditions true:

- OMPROUTE is started and active
- At least one of monitored interfaces is defined and active (if MONINTERFACE is configured)
- At least one dynamic route over the monitored interfaces is available (if MONINTERFACE DYNROUTE is configured)

Any sysplex-related definitions within the TCP/IP profile (for example, VIPADYNAMIC or IPCONFIG/IPCONFIG6 DYNAMICXCF statements) are not processed until the sysplex group is joined. The value Yes can be configured by specifying the DELAYJOIN keyword for the SYSPLEXMONITOR parameter on the GLOBALCONFIG profile statement.

Join

Indicates whether the TCP/IP stack joins the sysplex group during stack initialization.

The value Yes indicates that the TCP/IP stack immediately attempts to join the sysplex group during stack initialization. This is the default setting.

The value No indicates that the TCP/IP stack does not join the sysplex group during stack initialization. You can configure the value No by specifying the NOJOIN keyword for the SYSPLEXMONITOR parameter on the GLOBALCONFIG profile statement.

If NOJOIN is configured, the TCP/IP stack does not process any VIPADYNAMIC block or DYNAMICXCF statements. Any other GLOBALCONFIG SYSPLEXMONITOR parameter settings (configured or default) are ignored, and the settings are saved in case you want the TCP/IP stack to join the sysplex group at a later time.

If you subsequently issue a VARY TCPIP,,SYSPLEX,JOINGROUP command, the NOJOIN setting is overridden and the saved GLOBALCONFIG SYSPLEXMONITOR parameter settings become active. For example, if you configure NOJOIN and DELAYJOIN, DELAYJOIN is initially ignored.

After you issue a V TCPIP,,SYSPLEX,JOINGROUP command, NOJOIN is overridden, DELAYJOIN becomes active, and the stack joins the sysplex group if OMPROUTE is initialized.

Any sysplex-related definitions within the TCP/IP profile, such as VIPADYNAMIC or IPCONFIG DYNAMICXCF statements, are not processed until the TCP/IP stack joins the sysplex group.

MonIntf
Indicates whether the TCP/IP stack is monitoring the status of specified network interfaces.

The value No indicates that the TCP/IP stack is not monitoring the status of network interfaces. The default value is No and it can be configured by specifying the NOMONINTERFACE keyword for the SYSPLEXMONITOR parameter on the GLOBALCONFIG profile statement.

The value Yes indicates that the TCP/IP stack is monitoring the status of network interfaces that have the MONSYSPLEX attribute specified on the LINK or INTERFACE profile statement. The value Yes can be configured by specifying the MONINTERFACE keyword for the SYSPLEXMONITOR parameter on the GLOBALCONFIG profile statement.

DynRoute
Indicates whether the TCP/IP stack is monitoring the presence of dynamic routes over the monitored network interfaces.

The value No indicates that the TCP/IP stack is not monitoring the presence of dynamic routes over monitored network interfaces. The default value is No and it can be configured by specifying the NODYNROUTE keyword for the SYSPLEXMONITOR MONINTERFACE parameter on the GLOBALCONFIG profile statement.

The value Yes indicates that the TCP/IP stack is monitoring the presence of dynamic routes over monitored network interfaces that have the MONSYSPLEX attribute specified on the LINK or INTERFACE statement. It can be configured by specifying the DYNROUTE keyword for the SYSPLEXMONITOR MONINTERFACE parameter on the GLOBALCONFIG profile statement.

AutoRejoin
Indicates whether the TCP/IP stack automatically rejoins the sysplex group when all detected problems that caused the stack to leave the group are relieved.

The value No indicates that the stack does not rejoin the group or restore its VIPADYNAMIC definitions when all detected problems have been relieved. The default value is No and it can be configured by specifying the NOAUTOREJOIN keyword for the SYSPLEXMONITOR parameter on the GLOBALCONFIG profile statement.

The value Yes indicates that the stack automatically rejoins the sysplex group and restores all of its VIPADYNAMIC configuration definitions. The value Yes can be configured by specifying the AUTOREJOIN keyword for the SYSPLEXMONITOR parameter on the GLOBALCONFIG profile statement.

Restriction: You can specify the AUTOREJOIN keyword only if the RECOVERY keyword is also specified (or is currently enabled) on the SYSPLEXMONITOR parameter on the GLOBALCONFIG profile statement.

zIIP
Displays information about displacing CPU cycles for various functions onto a System z Information Integration Processor (zIIP). The value Yes for a function indicates that cycles can be displaced to a zIIP when at least one zIIP device is online. Issue the MVS D M=CPU command to display zIIP status. See displaying system configuration information details in z/OS MVS System Commands for more information about displaying processor status.

IPSecurity
Indicates whether the stack is configured to displace CPU cycles for IPSec workload onto a zIIP. This field can have the following values:

Yes
Indicates that IPSec CPU cycles are displaced to a zIIP as long as at least one zIIP device is online.

No
Indicates that IPSec CPU cycles are not being displaced to a zIIP.

IQDIOMultiWrite
Indicates whether the stack is configured to displace CPU cycles for HiperSockets multiple write workload onto a zIIP. This field can have the following values:

Yes
Indicates that the stack is configured to permit HiperSockets multiple write CPU cycles to be displaced to a zIIP.

No
Indicates that the stack is configured to not permit HiperSockets multiple write CPU cycles to be displaced to a zIIP.

SMCGlobal
Displays the global settings for Shared Memory Communications (SMC). SMC includes Shared Memory Communications over Remote Direct Memory Access (RDMA), or SMC-R, for external data network communications and Shared Memory Communications - Direct Memory Access (SMC-D). This field has the following values:

AutoCache
Indicates whether AUTOCACHE support is enabled for this TCP/IP stack. The following values are valid:

Yes
Indicates that AUTOCACHE support is enabled. The AUTOCACHE subparameter was specified with the SMCGLOBAL parameter on the GLOBALCONFIG profile statement or AUTOCACHE support is enabled by default. This support is started only when SMC is enabled. SMC is enabled if the value of either the SMCR or the SMCD field is Yes.

No
Indicates that AUTOCACHE support is not enabled. The NOAUTOCACHE subparameter was specified with the SMCGLOBAL parameter on the GLOBALCONFIG profile statement.

AutoSMC
Indicates whether the AUTOSMC monitoring function is enabled for this TCP/IP stack. For more information about the AUTOSMC monitoring function, see AUTOSMC monitoring function in z/OS Communications Server: IP Configuration Guide. The following values are valid:

Yes
Indicates that the AUTOSMC monitoring function is enabled. The AUTOSMC subparameter was specified with the SMCGLOBAL parameter on the GLOBALCONFIG profile statement or the AUTOSMC monitoring function was enabled by default. This function is started only when SMC is enabled. SMC is enabled if the value of either the SMCR or the SMCD field is Yes.

No
Indicates that the AUTOSMC monitoring function is not enabled. The NOAUTOSMC subparameter was specified with the SMCGLOBAL parameter on the GLOBALCONFIG profile statement.

SMCR
Indicates whether this stack supports Shared Memory Communications over Remote Direct Memory Access (SMC-R) for intra-ensemble data network (IEDN) or external data network communications. This field can have the following values:

Yes
Indicates that this stack can communicate with other stacks on the IEDN or external data network by using SMC-R. The SMCR parameter was specified on the GLOBALCONFIG profile statement. When the SMCR field has the value Yes, the following information is displayed:

FixedMemory
Indicates the maximum amount, in megabytes, of 64-bit private storage that the stack can use for the send and receive buffers that are required for SMC-R communications. The fixed

memory value was defined by using the SMCR FIXEDMEMORY parameter on the GLOBALCONFIG. If the SMCR FIXEDMEMORY parameter was not specified on the GLOBALCONFIG statement, the default value of 256 is displayed.

TcpKeepMinInt
Indicates the minimum supported TCP keepalive interval for SMC-R links. Use the SMCR TCPKEEPMININTERVAL parameter on the GLOBALCONFIG statement to define the interval. For applications that are using the TCP_KEEPALIVE setsockopt() option, this interval indicates the minimum interval that TCP keepalive packets are sent on the TCP path of an SMC-R link. The range is 0 - 2147460 seconds. If the interval value is set to 0, TCP keepalive probe packets on the TCP path of an SMC-R link are disabled. If the SMCR TCPKEEPMININTERVAL parameter was not specified on the GLOBALCONFIG statement, then the default interval value of 300 is displayed.

PFID
Indicates the Peripheral Component Interconnect Express (PCIe) function ID (PFID) value that was defined using SMCR PFID parameter. The combination of PFID and port number uniquely identifies an "RoCE Express". The stack uses "RoCE Express" for SMC-R communications with other stacks on the IEDN or external data network. The PFID is a 2-byte hexadecimal value.

PortNum
Indicates the "RoCE Express" port number that is used for the associated PFID. The PortNum value was specified with the PFID value on the SMCR parameter of the GLOBALCONFIG statement in the TCP/IP profile. The port number can be 1 or 2; the default port is 1.

Note : When PFID represents a RoCE Express2 feature, the PortNum value is the port number configured for the PFID in the Hardware Configuration Definition (HCD). The port number is learned by VTAM during activation of the PFID and might be different from the value coded for PORTNUM for this PFID on the GLOBALCONFIG SMCR statement.

MTU
Indicates the configured maximum transmission unit (MTU) value that is used for the associated PFID. The MTU value can be 1024 or 2048 and the default MTU value is 1024.

No
Indicates that this stack cannot communicate with other stacks on the IEDN or external data network by using SMC-R communications. The NOSMCR parameter was specified on the GLOBALCONFIG profile statement or the value was set by default.

SMCD
Indicates whether this stack supports SMC-D. This field can have the following values:

Yes
Indicates that this stack can communicate with other stacks by using SMC-D. The SMCD parameter was specified on the GLOBALCONFIG profile statement. When the SMCD field has the value Yes, the following information is displayed:

FixedMemory
Indicates the maximum amount, in megabytes, of 64-bit private storage that the stack can use for the receive buffers that are required for SMC-D communications. The fixed memory value was defined by using the SMCD FIXEDMEMORY parameter on the GLOBALCONFIG statement. If the SMCD FIXEDMEMORY parameter was not specified on the GLOBALCONFIG statement, the default value of 256 is displayed.

TcpKeepMinInt
Indicates the minimum supported TCP keepalive interval for SMC-R links. Use the SMCD TCPKEEPMININTERVAL parameter on the GLOBALCONFIG statement to define the interval. For applications that are using the TCP_KEEPALIVE setsockopt() option, this interval indicates the minimum interval that TCP keepalive packets are sent on the TCP path of an SMC-D link. The range is 0 - 2147460 seconds. If the interval value is set to 0, TCP keepalive probe packets on the TCP path of an SMC-D link are disabled. If the SMCD

TCPKEEPMININTERVAL parameter was not specified on the GLOBALCONFIG statement, the default interval value of 300 is displayed.

No
Indicates that this stack cannot communicate with other stacks by using SMC-D communications. The NOSMCD parameter was specified on the GLOBALCONFIG profile statement or the value was set by default.

ZERT
Indicates whether this stack supports z/OS Encryption Readiness Technology (zERT).

The value Yes indicates that ZERT is in effect (the ZERT parameter was specified in the GLOBALCONFIG profile statement) and that the zERT discovery function is enabled for this TCP/IP stack. When the ZERT field has the value Yes, the following information is displayed:

Aggregation
Indicates whether the zERT aggregation function is enabled for this TCP/IP stack. The following values are valid:

Yes
Indicates that the zERT aggregation function is enabled. The AGGREGATION subparameter was specified with the ZERT parameter on the GLOBALCONFIG profile statement.

INTVAL
Indicates the interval at which zERT aggregation will write the data it has collected as SMF type 119 subtype 12 (zERT summary) records. A value of SMF indicates that the records will be written on the z/OS system's SMF interval. A numeric value indicates the recording interval in hours (from 1 to 24).

SYNCVAL

Indicates the reference time from which zERT aggregation will start writing SMF type 119 subtype 12 (zERT summary) records. This value is displayed in 24 hour clock format (hh:mm).

No
Indicates that the zERT aggregation function is not enabled. The NOAGGREGATION subparameter was specified with the ZERT parameter on the GLOBALCONFIG profile statement, or the zERT aggregation function was not enabled by default.

The value No indicates that zERT is not in effect (the NOZERT parameter was specified in the GLOBALCONFIG profile statement or is in effect by default).

- **Network Monitor Configuration information**

Display the following configured network monitor information defined in the NETMONITOR profile statement. For more information about each network monitor parameter, see the NETMONITOR profile statement information in the z/OS Communications Server: IP Configuration Reference.

PktTrcSrv
Indicates whether the packet trace service is enabled or disabled. The value Yes indicates that PKTTRCSERVICE is in effect (the PKTTRCSERVICE parameter was specified in the NETMONITOR profile statement). The value No indicates that PKTTRCSERVICE is not in effect (the NOPKTTRCSERVICE parameter was specified in the NETMONITOR profile statement or is in effect by default).

TcpCnnSrv
Indicates whether the TCP connection information service is enabled or disabled. The value Yes indicates that TCPCONNSERVICE is in effect (the TCPCONNSERVICE parameter was specified in the NETMONITOR profile statement). The value No indicates that TCPCONNSERVICE is not in effect (the NOTCPCONNSERVICE parameter was specified in the NETMONITOR profile statement or is in effect by default).

MinLifTim
The minimum lifetime for a new TCP connection to be reported by the service when the TCP connection information service is enabled. If the NOTCPCONNSERVICE parameter was specified in

the NETMONITOR profile statement or is in effect by default, then the MinLifTim field is not displayed.

NtaSrv
Indicates whether the OSAENTA trace service is enabled or disabled. The value Yes indicates that NTATRCSERVICE is in effect (the NTATRCSERVICE parameter was specified in the NETMONITOR profile statement). The value No indicates that NTATRCSERVICE is not in effect (the NONTATRCSERVICE parameter was specified in the NETMONITOR profile statement or is in effect by default).

SmfSrv
Indicates whether the real-time SMF information service is enabled or disabled. The value Yes indicates that SMFSERVICE is enabled (the SMFSERVICE parameter was specified in the NETMONITOR profile statement). The value No indicates that SMFSERVICE is disabled (the NOSMFSERVICE parameter was specified in the NETMONITOR profile statement or is disabled by default).

IPSecurity
Indicates whether the real-time SMF service is providing IPSec SMF records. The value Yes indicates that IPSec SMF records are being provided (either the SMFSERVICE parameter was specified with the IPSECURITY subparameter on the NETMONITOR profile statement or the SMFSERVICE parameter was specified without any subparameters). The value No indicates that IPSec SMF records are not being provided (the SMFSERVICE parameter was specified with the NOIPSECURITY subparameter on the NETMONITOR profile statement). This field is displayed only if the SmfSrv value is Yes.

Profile
Indicates whether the real-time SMF service is providing TCP/IP profile SMF records. The value Yes indicates that TCP/IP profile SMF records are being provided (either the SMFSERVICE parameter was specified with the PROFILE subparameter on the NETMONITOR profile statement, or the SMFSERVICE parameter was specified without any subparameters). The value No indicates that TCP/IP profile SMF records are not being provided (the SMFSERVICE parameter was specified with the NOPROFILE subparameter on the NETMONITOR profile statement). This field is displayed only if the SmfSrv value is Yes.

CSSMTP
Indicates whether the real-time SMF service is providing CSSMTP SMF 119 records for subtype 48, 49, 51 and 52. The value Yes indicates that CSSMTP SMF records are being provided (either the SMFSERVICE parameter was specified with the CSSMTP subparameter on the NETMOINTOR profile statement or the SMFSERVICE parameter was specified without any sub parameters). The value No indicates that CSSMTP SMF records are not being provided (the SMFSERVICE parameter was specified with the NOCSSMTP subparameter on the NETMONITOR profile statement). This field is displayed only if the SMFSrv value is Yes.

CSMAIL
Indicates whether the real-time SMF service is providing CSSMTP SMF 119 records for subtype 50. The value Yes indicates that CSSMTP SMF mail records are being provided (either the SMFSERVICE parameter was specified with the CSSMTP subparameter on the NETMOINTOR profile statement or the SMFSERVICE parameter was specified without any subparameters). The value No indicates that CSSMTP SMF mail records are not being provided (the SMFSERVICE parameter was specified with the NOCSSMTP subparameter on the NETMONITOR profile statement). This field is displayed only if the SMFSrv value is Yes.

DVIPA
Indicates whether the real-time SMF service is providing sysplex event SMF records. The value Yes indicates that sysplex event SMF records are being provided (either the SMFSERVICE parameter was specified with the DVIPA subparameter on the NETMONITOR profile statement, or the SMFSERVICE parameter was specified without any subparameters). The value No indicates that sysplex event SMF records are not being provided (the SMFSERVICE parameter was specified with the NODVIPA subparameter on the NETMONITOR profile statement). This field is displayed only if the SmfSrv value is Yes.

ZertSrv
Indicates whether the real-time zERT Detail SMF information service is enabled or disabled. The value Yes indicates that the zERT Detail service is enabled (the ZERTSERVICE parameter was specified on the NETMONITOR profile statement). The value No indicates that the zERT Detail service is disabled (the NOZERTSERVICE parameter was specified on the NETMONITOR profile statement or is disabled by default).

ZertSum
Indicates whether the real-time zERT Summary SMF information service is enabled or disabled. The value Yes indicates that the zERT Summary service is enabled (the ZERTSUMMARY parameter was specified on the NETMONITOR profile statement). The value No indicates that the zERT Summary service is disabled (the NOZERTSUMMARY parameter was specified on the NETMONITOR profile statement or is disabled by default).

- **Autolog Configuration Information**

WaitTime
The time, displayed in seconds, that is specified on the AUTOLOG statement that represents the length of time TCP/IP waits for a procedure to stop if the procedure is still active at startup and TCP/IP is attempting to start the procedure again. The procedure could still be active if it did not stop when TCP/IP was last shut down.

ProcName
The procedure that the TCP/IP address space starts.

JobName
The job name used for the PORT reservation statement. The job name might be identical to the procedure name; however, for z/OS UNIX jobs that spawn listener threads, the names are not the same.

ParmString
A string to be added following the START ProcName value. The ParmString value can be up to 115 characters in length and can span multiple lines. If the PARMSTRING parameter on the AUTOLOG profile statement was not specified or if the *parm_string* value was specified with a blank string, then this field displays blanks.

DelayStart
Indicates whether TCP/IP delays starting this procedure until the TCP/IP stack has completed one or more processing steps. This field can have the following values:

Yes
Indicates that the TCP/IP stack does not start this procedure until it has completed all of the processing steps identified by the following subparameters:

DVIPA
TCP/IP delays starting this procedure until after the TCP/IP stack has joined the sysplex group and processed its dynamic VIPA configuration (DELAYSTART was specified on the entry for this procedure in the AUTOLOG profile statement with no additional subparameters, or DELAYSTART was specified with the DVIPA subparameter).

TTLS
TCP/IP delays starting this procedure until after the Policy Agent has successfully installed the AT-TLS policy in the TCP/IP stack and AT-TLS services are available (DELAYSTART was specified with the TTLS subparameter on the entry for this procedure in the AUTOLOG profile statement).

No
Indicates that this procedure is started when TCP/IP is started (DELAYSTART was not specified on the entry for this procedure in the AUTOLOG profile statement).

- **Data Trace Settings if socket data trace is on**

JobName
The application address space name specified on the DATTRACE command or asterisk (*), if not specified.

TrRecCnt
The number of packets traced for this DATTRACE command.

Length
The value of the ABBREV keyword of the DATTRACE command or FULL to capture the entire packet.

IpAddr
The IP address from the IP keyword of the DATTRACE command or asterisk (*), if not specified.

SubNet
The subnet mask from the SUBNET keyword of the DATTRACE command or asterisk (*), if not specified.

PrefixLen
The prefix length specified on the DATTRACE command.

PortNum
The port number from the PORTNUM keyword of the DATTRACE command or an asterisk (*), if a value was not specified.

Not IPv6 enabled (SHORT format)

```
NETSTAT CONFIG MVS TCP/IP NETSTAT CS V2R2       TCPIP Name: TCPCS           11:37:31
TCP Configuration Table:
DefaultRcvBufSize:  00016384  DefaultSndBufSize: 00016384
DefltMaxRcvBufSize: 00262144  SoMaxConn:         0000001024
MaxReTransmitTime:  120.000   MinReTransmitTime: 0.500
RoundTripGain:      0.125     VarianceGain:      0.250
VarianceMultiplier: 2.000     MaxSegLifeTime:    30.000
DefaultKeepALive:   00000120  DelayAck:          Yes
RestrictLowPort:    Yes       SendGarbage:       No
TcpTimeStamp:       Yes       FinWait2Time:      010
TTLS:               No        EphemeralPorts:    1024-65535
SelectiveACK:       Yes       TimeWaitInterval:  30
DefltMaxSndBufSize  262144    RetransmitAttempt: 15
ConnectTimeOut:     0120      ConnectInitIntval: 1000
KeepAliveProbes:    10        KAProbeInterval:   060
Nagle:              No        QueuedRTT:         20
FRRThreshold:       3

UDP Configuration Table:
DefaultRcvBufSize: 00065535  DefaultSndBufSize: 00065535
CheckSum:          Yes       EphemeralPorts:    1024-65535
RestrictLowPort:   Yes       UdpQueueLimit:     No

IP Configuration Table:
Forwarding: Yes    TimeToLive: 00064  RsmTimeOut:  00060
IpSecurity: Yes
ArpTimeout: 01200  MaxRsmSize: 65535  Format:      Short
IgRedirect: Yes    SysplxRout: No     DoubleNop:   No
StopClawEr: No     SourceVipa: Yes
MultiPath:  Conn   PathMtuDsc: No     DevRtryDur:  0000000090
DynamicXCF: Yes
  IpAddr/PrefixLen: 193.9.200.3/28      Metric: 01
  SecClass: 008  SrcVipaInt: IPV4SRCVIPA
  SMCD:     Yes
QDIOAccel:  No
IQDIORoute: No
TcpStackSrcVipa: 201.1.10.10
ChecksumOffload: Yes             SegOffload: Yes

SMF Parameters:
Type 118:
  TcpInit:      00   TcpTerm:      02   FTPClient:    03
  TN3270Client: 04   TcpIpStats:   05
Type 119:
  TcpInit:      Yes  TcpTerm:      Yes  FTPClient:    Yes
  TcpIpStats:   Yes  IfStats:      Yes  PortStats:    Yes
  Stack:        Yes  UdpTerm:      Yes  TN3270Client: Yes
  IPSecurity:   No   Profile:      Yes  DVIPA:        Yes
  SmcrGrpStats: Yes  SmcrLnkEvent: Yes
  SmcdLnkStats: Yes  SmcdLnkEvent: Yes
  ZertDetail:   Yes   ZertSummary:  Yes
```

```
Global Configuration Information:
TcpIpStats: Yes  ECSALimit: 2096128K  PoolLimit: 2096128K
MlsChkTerm: No   XCFGRPID:  11        IQDVLANID: 27
SysplexWLMPoll: 060  MaxRecs:   100
ExplicitBindPortRange:  05000-06023   IQDMultiWrite: Yes
AutoIQDC: AllTraffic
AutoIQDX: AllTraffic                  AdjustDVIPAMSS: Auto
WLMPriorityQ: Yes
  IOPri1 0 1
  IOPri2 2
  IOPri3 3 4
  IOPri4 5 6 FWD
Sysplex Monitor:
  TimerSecs: 0060  Recovery: Yes  DelayJoin: No   AutoRejoin: Yes
  MonIntf:   Yes   DynRoute: Yes  Join:      Yes
zIIP:
  IPSecurity: Yes  IQDIOMultiWrite: Yes
SMCGlobal:
  AutoCache:  Yes  AutoSMC: Yes
SMCR: Yes
  FixedMemory: 100M  TcpKeepMinInt: 00000300
  PFID: 0018  PortNum: 1 MTU: 1024
  PFID: 0019  PortNum: 2 MTU: 1024
SMCD: Yes
  FixedMemory: 100M  TcpKeepMinInt: 00000300
ZERT: Yes
  Aggregation: Yes
  INTVAL: SMF
```

```
Network Monitor Configuration Information:
PktTrcSrv: Yes  TcpCnnSrv: Yes  MinLifTim: 3  NtaSrv: Yes
SmfSrv: Yes
  IPSecurity: Yes  Profile: Yes  CSSMTP: Yes  CSMAIL: Yes  DVIPA: Yes
ZertSrv: Yes    ZertSum: Yes

Autolog Configuration Information: Wait Time: 0300
ProcName: FTPD      JobName: FTPD
  ParmString:
  DelayStart: Yes
    DVIPA    TTLS
```

IPv6 enabled or request for LONG format

```
NETSTAT CONFIG MVS TCP/IP NETSTAT CS V2R2       TCPIP Name: TCPCS          19:54:08
TCP Configuration Table:
DefaultRcvBufSize:  00016384  DefaultSndBufSize: 00016384
DefltMaxRcvBufSize: 00262144  SoMaxConn:         0000001024
MaxReTransmitTime:  120.000   MinReTransmitTime: 0.500
RoundTripGain:      0.125     VarianceGain:      0.250
VarianceMultiplier: 2.000     MaxSegLifeTime:    30.000
DefaultKeepALive:   00000120  DelayAck:          Yes
RestrictLowPort:    Yes       SendGarbage:       No
TcpTimeStamp:       Yes       FinWait2Time:      010
TTLS:               No        EphemeralPorts:    1024-65535
SelectiveACK:       Yes       TimeWaitInterval:  30
DefltMaxSndBufSize  262144    RetransmitAttempt: 15
ConnectTimeOut:     0120      ConnectInitIntval: 1000
KeepAliveProbes:    10        KAProbeInterval:   060
Nagle:              No        QueuedRTT:         20
FRRThreshold:       3

UDP Configuration Table:
DefaultRcvBufSize: 00065535  DefaultSndBufSize: 00065535
CheckSum:          Yes       EphemeralPorts:    1024-65535
RestrictLowPort:   Yes       UdpQueueLimit:     No

IP Configuration Table:
Forwarding: Yes    TimeToLive: 00064  RsmTimeOut:  00060
IpSecurity: Yes
ArpTimeout: 01200  MaxRsmSize: 65535  Format:      Long
IgRedirect: Yes    SysplxRout: No     DoubleNop:   No
StopClawEr: No     SourceVipa: Yes
MultiPath:  Conn   PathMtuDsc: No     DevRtryDur:  0000000090
DynamicXCF: Yes
  IpAddr/PrefixLen: 193.9.200.3/28      Metric: 01
  SecClass: 008  SrcVipaInt: IPV4SRCVIPA
  SMCD:     Yes
QDIOAccel:  Yes       QDIOAccelPriority: 2
IQDIORoute: n/a
TcpStackSrcVipa: 201.1.10.10
ChecksumOffload: Yes             SegOffload: Yes

IPv6 Configuration Table:
Forwarding:    Yes  HopLimit:   00255  IgRedirect:  No
SourceVipa:    Yes  MultiPath:  Conn   IcmperrLim:  00003
IgRtrHopLimit: No
IpSecurity: Yes
  OSMSecClass: 255
DynamicXCF: Yes
  IpAddr: 2001:db8::9:67:115:5
  IntfID: 0009:0067:0011:0001
  SrcVipaInt: IPV6SRCVIPA
  SecClass: 008
  SMCD:     Yes
TcpStackSrcVipa: IPV6STKSRCVIPA
TempAddresses: Yes
  PreferredLifetime: 24   ValidLifetime: 168
ChecksumOffload: Yes             SegOffload: Yes

SMF Parameters:
Type 118:
  TcpInit:      00   TcpTerm:      02   FTPClient:    03
  TN3270Client: 04   TcpIpStats:   05
Type 119:
  TcpInit:      Yes  TcpTerm:      Yes  FTPClient:    Yes
  TcpIpStats:   Yes  IfStats:      Yes  PortStats:    Yes
  Stack:        Yes  UdpTerm:      Yes  TN3270Client: Yes
  IPSecurity:   No   Profile:      Yes  DVIPA:        Yes
  SmcrGrpStats: Yes  SmcrLnkEvent: Yes
  SmcdLnkStats: Yes  SmcdLnkEvent: Yes
  ZertDetail:   Yes   ZertSummary:  Yes
```

```
Global Configuration Information:
TcpIpStats: Yes  ECSALimit: 2096128K  PoolLimit: 2096128K
MlsChkTerm: No   XCFGRPID:  11        IQDVLANID: 27
SysplexWLMPoll: 060  MaxRecs:   100
ExplicitBindPortRange:  05000-06023   IQDMultiWrite: Yes
AutoIQDC: AllTraffic
AutoIQDX: AllTraffic                  AdjustDVIPAMSS: Auto
WLMPriorityQ: Yes
  IOPri1 0 1
  IOPri2 2
  IOPri3 3 4
  IOPri4 5 6 FWD
Sysplex Monitor:
  TimerSecs: 0060  Recovery: Yes  DelayJoin: No   AutoRejoin: Yes
  MonIntf:   Yes   DynRoute: Yes  Join:      Yes
zIIP:
  IPSecurity: Yes  IQDIOMultiWrite: Yes
SMCGlobal:
  AutoCache:  Yes  AutoSMC: Yes
SMCR: Yes
  FixedMemory: 100M  TcpKeepMinInt: 00000300
  PFID: 0018  PortNum: 1 MTU: 1024
  PFID: 0019  PortNum: 2 MTU: 1024
SMCD: Yes
  FixedMemory: 100M  TcpKeepMinInt: 00000300
ZERT: Yes
  Aggregation: Yes
    INTVAL: 2
    SYNCVAL: 12:00

Network Monitor Configuration Information:
PktTrcSrv: Yes  TcpCnnSrv: Yes  MinLifTim: 3  NtaSrv: Yes
SmfSrv:    Yes
  IPSecurity: Yes  Profile: Yes  CSSMTP: Yes  CSMAIL: Yes  DVIPA: Yes
ZertSrv:   Yes    ZertSum: Yes

Data Trace Setting:
JobName: *         TrRecCnt: 00000000  Length: FULL
IpAddr:  *                SubNet: *
PortNum: *

Autolog Configuration Information: Wait Time: 0300
ProcName: FTPD      JobName: FTPD
  ParmString:
  DelayStart: Yes
    DVIPA    TTLS
```

Chapter 5. IP Programmer's Guide and Reference

Format and details for poll-type requests

The following poll-type requests are provided by EZBNMIFR. The request constant, which is specified in the NWMType field in the NWMHeader data structure, follows the request name. Some requests support filters. See *Filter request section* for a description of each filter and the information about which filters are supported by each request. For more information about Shared Memory Communications over Remote Direct Memory Access (SMC-R) and Shared Memory Communications - Direct Memory Access (SMC-D), see Shared Memory Communications in z/OS Communications Server: IP Configuration Guide.

- **GetConnectionDetail (NWMTcpConnType)**

  Use this request to obtain information about active TCP connections, including SMC information for TCP connections that traverse SMC links.

  Guideline: When you use filters with this request, you can experience a performance improvement in retrieving the connection details if every filter contains a 4-tuple (local address, local port, remote address and remote port) for a connection. Additional filter values can be specified in each filter along with the 4-tuple.

- **GetDVIPAConnRTab (NWMDvConnRTabType)**

  Use this request to obtain information about dynamic virtual IP addresses (DVIPA) connections. This call returns a list of IPv4 and IPv6 DVIPA TCP connections. Entries are returned for the following:

  – All DVIPA interfaces for which MOVEABLE IMMEDIATE or NONDISRUPTIVE was specified.
  – On a sysplex distributor routing stack, every connection that is being routed through this distributor.
  – On a stack taking over a DVIPA, every connection to the DVIPA.
  – On a sysplex distributor target stack or a stack that is in the process of giving up a DVIPA, every connection for which the stack is an endpoint.

  If none of these apply, then an empty response buffer is returned with a successful reason value, return code, and reason code. If the invoked TCP/IP stack has not joined a sysplex, then return value -1, return code EAGAIN, and reason code JRMustBeSysplex are returned without any other data.

- **GetDVIPAList (NWMDvListType)**

  Use this request to obtain information about dynamic virtual IP addresses (DVIPAs). This request returns a list of all IPv4 and IPv6 DVIPAs for the invoked TCP/IP stack. For each DVIPA, the MVS system name, TCP/IP job name, and various status information are returned.

- **GetDVIPAPortDist (NWMDvPortDistType)**

  Use this request to obtain information about dynamic virtual IP address (DVIPA) port distribution. This request returns a list of IPv4 and IPv6 distributed DVIPAs and ports. For each distributed DVIPA and port pair, one or more entries are returned for each target TCP/IP stack. If the invoked TCP/IP stack has not joined a sysplex, then return value -1, return code EAGAIN, and reason code JRMustBeSysplex are returned without any other data. If the TCP/IP stack is not a distributing stack, then an empty response buffer is returned with a successful return value, return code, and reason code. If the same DVIPA and port pair are affected by more than one QOS Policy, then an entry with the same DVIPA and port is returned for each QOS policy.

- **GetDVIPARoute (NWMDvRouteType)**

  Use this request to obtain information about dynamic virtual IP address (DVIPA) routes. This request returns a list of information that is defined on VIPAROUTE profile statements. Each entry includes the dynamic XCF address of a target TCP/IP stack and the corresponding target IP address that is used to route connection requests to that TCP/IP stack. Output is returned only by a distributing TCP/IP stack, or by a backup TCP/IP stack for a distributed DVIPA when the backup TCP/IP stack is assuming

ownership of the distributed DVIPA. If the invoked TCP/IP stack has not joined a sysplex, then return value -1, return code EAGAIN, and reason code JRMustBeSysplex are returned without any other data. If the invoked TCP/IP stack is neither a distributing stack nor a backup stack, then an empty response buffer is returned with a successful return value, return code, and reason code.

- **GetFTPDaemonConfig (NWMFTPDConfigType)**

  Use this request to obtain configuration data from one active FTP daemon.

  Rules : You must supply only one filter when using this request type. If the filter number is not 1 in the request header, the following information is returned:

  – Return value -1
  – Return code EINVAL
  – Reason code JRInvalidValue

  The filter must contain the ASID of the specific FTP daemon for which you want to obtain the configuration data. If no ASID is specified in the filter, the following information is returned:

  – Return value -1
  – Return code EINVAL
  – Reason code JRInvalidFilter

  To obtain the ASID for the FTP daemon, take the following steps:

  – Invoke EZBNMIFR for the GetTCPListener request to each TCP/IP stack to obtain the active FTP daemons.
  – Specify a filter with the application data (NWMFilterApplData) value of EZAFTP0D in the first 8 bytes to filter the active FTP daemons from other listeners. A daemon might be listening on multiple stacks.
  – Extract the ASID (NWMTCPLAsid) of each FTP daemon returned by the GetTCPListener request for which the GetFTPDaemonConfig request is issued.
  – Invoke EZBNMIFR for the GetFTPDaemonConfig request.
  – Specify a filter that contains the ASID of the FTP daemon to obtain the configuration data of the FTP daemon.

- **GetGlobalStats (NWMGlobalStatsType)**

  Use this request to obtain TCP/IP stack global statistics for IP, ICMP, TCP, SMC, and UDP processing. The statistics that are returned by the request are similar to those in the output of the Netstat STATS/-S report. This request does not support filtering.

- **GetIfs (NWMIfsType)**

  Use this request to obtain TCP/IP stack interface attributes and IP addresses. The attributes and IP address information that are returned by the request are similar to those in the output of the Netstat DEVLINKS/-d and HOME/-h reports. Detailed attribute information is supported only for strategic interface types. The strategic interface types are:

  – Loopback
  – OSA-Express QDIO Ethernet
  – HiperSockets
  – Multipath Channel Point-to-Point
  – Static VIPA

  "RoCE Express" interfaces are also strategic interfaces. Some information about "RoCE Express" interfaces is reported on this request, but the majority of the "RoCE Express" attributes can be obtained from the GetRnics (NWMRnicType) NMI request.

  Internal shared memory (ISM) interfaces are also strategic interfaces. Some information about ISM interfaces is reported on this request, but the majority of the ISM attributes can be obtained from the GetIsms (NWMIsmType) NMI request.

Dynamic VIPA interfaces are also strategic interfaces but their attributes can be obtained from the dynamic VIPA (DVIPA) NMI requests that are previously described in this topic. For non-strategic interface types, only the following information is provided:

- Interface name from the LINK profile statement
- Interface index
- Associated device name from the DEVICE profile statement
- Interface type
- Interface status at the DEVICE and LINK level
- Time stamp of last interface status change at the LINK level

This request does not support filtering.

- **GetIfStats (NWMIfsStatsType)**

  Use this request to obtain TCP/IP stack interface statistics for all interface types except for "RoCE Express" interfaces and ISM interfaces. The statistics that are returned by the request are similar to those in the output of the Netstat DEVLINKS/-d report with the addition of SNMP counters that are defined in the IF-MIB. For information about the IF-MIB, see RFC 2233. For information about how to access RFCs, see *Related protocol specifications*. Statistics are provided for all strategic interface types except for VIPA interfaces; the stack does not maintain counters for VIPA interfaces. This request also provides a time stamp of when the counters were last reset. This request does not support filtering.

  See GetRnics (NWMRnicType) NMI request for information about "RoCE Express" interfaces. See GetIsms (NWMIsmType) NMI request for information about ISM interfaces.

- **GetIfStatsExtended (NWMIfsStatsExtType)**

  Use this request to obtain data link control (DLC) tuning statistics for datapath devices that are used by active OSA-Express QDIO Ethernet and HiperSockets interfaces. The statistics that are returned by the request are similar to those in the output of the VTAM TNSTATS function and the SMF type 50 record. Counters are provided for each read and write queue for each datapath device. Because of performance considerations, the counters are not maintained by default as part of TCP/IP stack initialization. The first GetIfStatsExtended request causes the counters to be maintained for all active interfaces. Therefore, the read and write queue counters can be 0 in the response for the first request.

  This request also provides a time stamp of when the counters were last reset. The counters are reset if all the interfaces that are using a datapath device are deactivated. This request does not support filtering.

- **GetIsms (NWMIsmType)**

  Use this request to obtain information about internal shared memory (ISM) interfaces. The ISM interface information that the request returns is similar to the information that is provided in the Netstat DEvlinks/-d report. The NWMIsmDevIntr counter value is reset each time when the interfaces are deactivated. This request does not support filtering.

- **GetProfile (NWMProfileType)**

  Use this request to obtain information about the current TCP/IP profile statement settings. This request does not support filtering. To detect changes to the profile statement settings, callers can use this callable request to obtain an initial set of current profile settings, and then do one of the following actions:

  - Repeat the request, over a time interval, comparing returned data from a previous response to the returned data from the last response.
  - Obtain the SMF Type 119 subtype 4 TCP/IP profile event records. These records provide information about changes to the profile settings that are made by using the VARY TCPIP,,OBEYFILE command processing.
    - If the records are requested on the SMFCONFIG or NETMONITOR profile statements, they are created.

- If the records are requested on the SMFCONFIG profile statement, they are written to the MVS SMF data sets.
- If the records are requested on the NETMONITOR profile statement, they can be obtained from the real-time SMF data network management interface (NMI).

For more information about the real-time SMF NMI, see *Real-time TCP/IP network monitoring NMI*. For more information about the TCP/IP profile SMF record, see *TCP/IP profile event record (subtype 4)*. The SMF record might be created even if some errors occurred during the VARY TCPIP,,OBEYFILE command processing. To determine whether profile changes actually occurred, application programs that process these records must compare the sections of changed information to the previous profile settings.

- **GetRnics (NWMRnicType)**

  Use this request to obtain information about "RoCE Express" interfaces.

  – The "RoCE Express" interface information that the request returns is similar to the information that is provided in the Netstat DEvlinks/-d report.
  – The VTAM tuning statistics that the request returns are for active "RoCE Express" interfaces only. These statistics are similar to those in the output of the VTAM TNSTAT function and the SMF type 50 record. Because of performance considerations, the counters are not maintained by default as part of VTAM or TCP/IP stack initialization. The first GetRnics request causes the counters to be maintained for all active "RoCE Express" interfaces. Therefore, the counters can be 0 in the response for the first request. The counter values are reset each time when the "RoCE Express" interfaces are deactivated.

  This request does not support filtering.

- **GetSmcDLinks (NWMSmcDLinkType)**

  Use this request to obtain information about Shared Memory Communications - Direct Memory Access (SMC-D) links. The SMC-D link information that the request returns is similar to the information that is provided in the Netstat DEvlinks/-d report. This request does not support filtering.

- **GetSmcLinks (NWMSmcLinkType)**

  Use this request to obtain information about SMC-R link groups and the SMC-R links in each group. The SMC-R link group and SMC-R link information that is returned by the request is similar to the information provided in the Netstat DEvlinks/-d report. This request does not support filtering.

- **GetStorageStatistics (NWMStgStatsType)**

  Use this request to obtain information about TCP/IP storage utilization, including zERT Aggregation Records storage utilization, SMC storage utilization, and SMC-R send and receive buffer utilization. This request does not support filtering.

- **GetSysplexXCF (NWMSyXcfType)**

  Use this request to obtain information about all TCP/IP stacks in the subplex. This request returns a list of all TCP/IP stacks in the same subplex as the invoked TCP/IP stack. For each TCP/IP stack, the MVS system name and one or more dynamic XCF IP addresses are returned. There are no filters defined for this request. If the invoked TCP/IP stack has not joined a sysplex, then return value -1, return code EAGAIN, and reason code JRMustBeSysplex are returned without any other data.

- **GetTCPListeners (NWMTcpListenType)**

  Use this request to obtain information about active TCP listeners, including SMC information for TCP listeners that traverse SMC links.

- **GetTnConnectionData (NWMTnConnType)**

  Use this request to obtain information about TN3270E Telnet server connection performance data.

- **GetTnMonitorGroups (NWMTnMonGrpType)**

  Use this request to obtain information about TN3270E Telnet server monitor groups.

- **GetTnProfile (NWMTnProfileType)**

Use this request to obtain information about the current TN3270E Telnet server profile statement settings.

This request does not support filtering. To detect changes to the profile statement settings, callers can use this request to obtain an initial set of the current profile settings, and then do one of the following actions:

- Repeat the request, over a time interval, comparing returned data from a previous response to the returned data from the last response.
- Obtain the SMF Type 119 subtype 24 TN3270E Telnet server profile event records. These records provide information about changes to the profile settings that are made by using the VARY TCPIP,*tnproc*,OBEYFILE command processing.
  - If the records are requested by the TELNETGLOBALS SMFPROFILE profile statement or the TCP/IP stack NETMONITOR profile statement, they are created.
  - If the records are requested by the TELNETGLOBALS SMFCONFIG profile statement, they are written to the MVS SMF data sets.
  - If the records are requested by the NETMONITOR profile statement, they can be obtained from the real-time SMF data network management interface (NMI).

  For more information about the real-time SMF NMI, see *Real-time TCP/IP network monitoring NMI*. For more information about the TCP/IP profile SMF record, see *TN3270E Telnet server profile event record (subtype 24)*. The SMF record might be created even if some errors occurred during the VARY TCPIP,*tnproc*,OBEYFILE command processing. To determine whether profile changes occurred, application programs that process these records must compare the sections of information in the new record with the previous profile settings.
- The NWMTnGrpDtl option flag allows the caller to obtain all the range data in the various groups that a Telnet profile defines. The call can return multiple entries and can use SMF119TN_*xx*RngNum to determine the number of ranges that are returned in each entry. If the flag is not set, the call returns one entry that contains only the first SMF119TN_*xx*RngMax ranges for a group. Based on the profile, specifying NWMTnGrpDtl can require a large amount of memory to hold the entire profile.

  Tip : Regardless of the number of entries that are returned for a group, the SMF119TN_*xx*RngCnt field indicates the total number of ranges that the group defines, and the SMF119TN_*xx*Count field indicates the total number of LUs or elements in the group.

- **GetUDPTable (NWMUdpConnType)**

  Use this request to obtain information about active UDP sockets.

The general format of the request consists of the request header and the request section descriptors (triplets), which define the input data. A triplet describes the input filters and contains the offset, in bytes, of the request section relative to the beginning of the request buffer, the number of elements in the request section, and the length of an element in the request section.

TCP/IP statistics record (subtype 5)

The TCP/IP statistics record is collected at user-specified intervals. The record provides data about IP, TCP, UDP, and ICMP activity in the reporting TCP stack during the previous recording interval, including TCP activity for Shared Memory Communications over Remote Direct Memory Access (SMC-R) processing, and Shared Memory Communications - Direct Memory Access (SMC-D) processing. For those fields that provide an interval value, the cumulative value for each statistic reported can be obtained by adding the values reported for the statistic in the individual TCP/IP statistics interval records. Other fields provide the current value of a statistic and are not interval values. If TCP/IP statistics recording is turned off dynamically, or the TCP stack terminates, a final TCP/IP statistics record is generated to report close-out statistics.

The Type 119 TCP/IP statistics record is generated using the same user specified interval time value as the equivalent Type 118 TCPIPSTATISTICS record.

See *Common TCP/IP identification section* for the contents of the TCP/IP stack identification section. For the TCP/IP statistics record, the TCP/IP stack identification section indicates STACK as the subcomponent and X'08' (event record), X'20' (recording stop), or X'10' (recording shutdown) as the record reason.

Table 8 on page 78 shows the TCP/IP statistics record self-defining section:

Table 8. SMF records: TCP/IP statistics record self-defining section

Offset	Name	Length	Format	Description
0(X'0')	Standard SMF Header	24		Standard SMF header; subtype is 5(X'5')
Self-defining section				
24(X'5')	SMF119SD_TRN	2	Binary	Number of triplets in this record (7)
26(X'1A')		2	Binary	Reserved
28(X'1C')	SMF119IDOff	4	Binary	Offset to TCP/IP identification section
32(X'20')	SMF119IDLen	2	Binary	Length of TCP/IP identification section
34(X'22')	SMF119IDNum	2	Binary	Number of TCP/IP identification sections
36(X'24')	SMF119S1Off	4	Binary	Offset to IPv4 IP statistics section
40(X'28')	SMF119S1Len	2	Binary	Length of IPv4 IP statistics section
42(X'2A')	SMF119S1Num	2	Binary	Number of IPv4 IP statistics sections
44(X'2C')	SMF119S2Off	4	Binary	Offset to TCP statistics section
48(X'30')	SMF119S2Len	2	Binary	Length of TCP statistics section
50(X'32')	SMF119S2Num	2	Binary	Number of TCP statistics sections
52(X'34')	SMF119S3Off	4	Binary	Offset to UDP statistics section
56(X'38')	SMF119S3Len	2	Binary	Length of UDP statistics section
58(X'3A')	SMF119S3Num	2	Binary	Number of UDP statistics sections
60(X'3C')	SMF119S4Off	4	Binary	Offset to IPv4 ICMP statistics section
64(X'40')	SMF119S4Len	2	Binary	Length of IPv4 ICMP statistics section
66(X'42')	SMF119S4Num	2	Binary	Number of IPv4 ICMP statistics sections
68 (X'44')	SMF119S5Off	4	Binary	Offset to IPv6 IP statistics section
72 (X'48')	SMF119S5Len	2	Binary	Length of IPv6 IP statistics section
74 (X'4A')	SMF119S5Num	2	Binary	Number of IPv6 IP statistics sections
76 (X'4C')	SMF119S6Off	4	Binary	Offset to IPv6 ICMP statistics section
80 (X'50')	SMF119S6Len	2	Binary	Length of IPv6 ICMP statistics section

Table 8. SMF records: TCP/IP statistics record self-defining section (continued)				
Offset	**Name**	**Length**	**Format**	**Description**
82 (X'52')	SMF119S6Num	2	Binary	Number of IPv6 ICMP statistics sections
84 (X'54')	SMF119S7Off	4	Binary	Offset to storage statistics section
88 (X'58')	SMF119S7Len	2	Binary	Length of storage statistics section
90 (X'5A')	SMF119S7Num	2	Binary	Number of storage statistics sections

Table 9 on page 79 shows the IP statistics section:

Table 9. IP statistics section				
Offset	**Name**	**Length**	**Format**	**Description**
0(X'0')	SMF119AP_TSIPDuration	8	Binary	Duration of recording interval in microseconds, where bit 51 is equivalent to one microsecond
8(X'8')	SMF119AP_TSIPRecData	4	Binary	Number of datagrams received
12(X'C')	SMF119AP_TSIPDscData	4	Binary	Number of input datagrams discarded due to errors in their IP headers
16(X'10')	SMF119AP_TSIPDscDAddr	4	Binary	Number of input datagrams discarded because the IP address in their IP header's destination field was not valid
20(X'14')	SMF119AP_TSIPAttFwdData	4	Binary	Number of attempts to forward datagrams
24(X'18')	SMF119AP_TSIPDscDUnkPr	4	Binary	Number of datagrams discarded because of an unknown or unsupported protocol
28(X'1C')	SMF119AP_TSIPDscDOth	4	Binary	Number of input datagrams discarded that are not accounted for in another input discard counter
32(X'20')	SMF119AP_TSIPDlvData	4	Binary	Number of datagrams delivered
36(X'24')	SMF119AP_TSIPXData	4	Binary	Number of datagrams transmitted
40(X'28')	SMF119AP_TSIPXDscOth	4	Binary	Number of outbound transmitted datagrams discarded, due to reasons other than no route being available
44(X'2C')	SMF119AP_TSIPXDscRoute	4	Binary	Number of outbound transmitted datagrams discarded, due to no route being available
48(X'30')	SMF119AP_TSIPTimeouts	4	Binary	Number of reassembly timeouts
52(X'34')	SMF119AP_TSIPRecDRsbm	4	Binary	Number of received datagrams requiring assembly
56(X'38')	SMF119AP_TSIPRsmb	4	Binary	Number of datagrams reassembled

Table 9. IP statistics section (continued)				
Offset	**Name**	**Length**	**Format**	**Description**
60(X'3C')	SMF119AP_TSIPFailRsmb	4	Binary	Number of failed reassembly attempts
64(X'40')	SMF119AP_TSIPRecFgmt	4	Binary	Number of fragmented datagrams received
68(X'44')	SMF119AP_TSIPDscDFgmt	4	Binary	Number of discarded datagrams due to fragmentation failures
72(X'48')	SMF119AP_TSIPXFgmt	4	Binary	Number of fragments generated
76(X'4C')	SMF119AP_TSIPRouteDisc	4	Binary	Number of routing discards
80(X'50')	SMF119AP_TSIPMaxRsmb	4	Binary	Maximum active number of reassemblies
84(X'54')	SMF119AP_TSIPCurRsmb	4	Binary	Number of currently active reassemblies
88(X'58')	SMF119AP_TSIPRsmbFlags	4	Binary	Reassembly flags
92(X'5C')	SMF119AP_TSIPInCalls	4	Binary	Number of inbound calls from device layer
96(X'60')	SMF119AP_TSIPInUerrs	4	Binary	Number of received frame unpacking
100(X'64')	SMF119AP_TSIPIDMem	4	Binary	Number of discarded datagrams, due to memory shortages
104(X'68')	SMF119AP_TSIPODSync	4	Binary	Number of transmitted datagrams discarded, due to Sync errors
108(X'6C')	SMF119AP_TSIPODAsyn	4	Binary	Number of transmitted datagrams discarded, due to Async errors
112(X'70')	SMF119AP_TSIPODMem	4	Binary	Number of transmitted datagrams discarded due to memory shortages

Table 10 on page 80 shows the TCP statistics section:

Table 10. TCP statistics section				
Offset	**Name**	**Length**	**Format**	**Description**
0(X'0')	SMF119AP_TSTCDuration	8	Binary	Duration of recording interval in microseconds, where bit 51 is equivalent to one microsecond
8(X'8')	SMF119AP_TSTCAlg	4	Binary	Retransmission algorithm
12(X'C')	SMF119AP_TSTCMinRet	4	Binary	Minimum retransmission time, in milliseconds

Table 10. TCP statistics section (continued)

Offset	Name	Length	Format	Description
16(X'10')	SMF119AP_TSTCMxRet	4	Binary	Maximum retransmission time, in milliseconds
20(X'14')	SMF119AP_TSTCMxCon	4	Binary	Maximum TCP connections
24(X'18')	SMF119AP_TSTCOpenConn	4	Binary	Number of active open connections, including active open connections across SMC links
28(X'1C')	SMF119AP_TSTCPassConn	4	Binary	Number of passive open connections, including passive open connections across SMC links
32(X'20')	SMF119AP_TSTCOFails	4	Binary	Number of open connection failures
36(X'24')	SMF119AP_TSTCConReset	4	Binary	Number of connection resets, including resets for connections across SMC links
40(X'28')	SMF119AP_TSTCEstab	4	Binary	Number of current establishments, including establishments for connections across SMC links
44(X'2C')	SMF119AP_TSTCInSegs	4	Binary	Number of input TCP segments, including input TCP segments for connections across SMC links
48(X'30')	SMF119AP_TSTCOSegs	4	Binary	Number of output TCP segments, including output TCP segments for connections across SMC links
52(X'34')	SMF119AP_TSTCRxSegs	4	Binary	Number of retransmitted segments
56(X'38')	SMF119AP_TSTCInErrs	4	Binary	Number of input errors

Table 10. TCP statistics section (continued)

Offset	Name	Length	Format	Description
60(X'3C')	SMF119AP_TSTCReset	4	Binary	Number of resets sent, including resets for connections across SMC links
64(X'40')	SMF119AP_TSTCConCls	4	Binary	Number of TCP connections closed, including connections across SMC links
68(X'44')	SMF119AP_TSTCConAttD	4	Binary	Number of TCP connection attempts discarded
72(X'48')	SMF119AP_TSTCTWRef	4	Binary	Number of TCP Timewait connections assassinated
76(X'4C')	SMF119AP_TSTCHOKAck	4	Binary	Number of header predictions (OK for ACK)
80(X'50')	SMF119AP_TSTCHOKDat	4	Binary	Number of header predictions (OK for Data)
84(X'54')	SMF119AP_TSTCIDupAck	4	Binary	Number of duplicate ACKs received
88(X'58')	SMF119AP_TSTCDscChecksum	4	Binary	Number of received packets discarded due to bad checksum values
92(X'5C')	SMF119AP_TSTCDscLen	4	Binary	Number of received packets discarded due to bad header length
96(X'60')	SMF119AP_TSTCDscInsData	4	Binary	Number of received packets discarded due to insufficient data
100(X'64')	SMF119AP_TSTCDscOldTime	4	Binary	Number of received packets discarded due to old timestamp information

Table 10. TCP statistics section (continued)

Offset	Name	Length	Format	Description
104(X'68')	SMF119AP_TSTCICmpDupSeg	4	Binary	Number of received complete duplicate segments
108(X'6C')	SMF119AP_TSTCIPartDupSeg	4	Binary	Number of received partial duplicate segments
112(X'70')	SMF119AP_TSTCICmpSegsWin	4	Binary	Number of complete segments received after window closure
116(X'74')	SMF119AP_TSTCIPartSegsWin	4	Binary	Number of partial segments received after window closure
120(X'78')	SMF119AP_TSTCIOOrder	4	Binary	Number of out-of-order segments received
124(X'7C')	SMF119AP_TSTCISegCls	4	Binary	Number of segments received after the TCP connection closed
128(X'80')	SMF119AP_TSTCIWinPr	4	Binary	Number of received window probes
132(X'84')	SMF119AP_TSTCIWinUp	4	Binary	Number of received window updates
136(X'88')	SMF119AP_TSTCOWinPr	4	Binary	Number of transmitted window probes
140(X'8C')	SMF119AP_TSTCOWinUp	4	Binary	Number of transmitted window updates
144(X'90')	SMF119AP_TSTCODlAck	4	Binary	Number of transmitted delayed ACKs
148(X'94')	SMF119AP_TSTCOKApr	4	Binary	Number of transmitted keepalive probes, including keepalive probes sent on the TCP path for connections across SMC links

Table 10. TCP statistics section (continued)

Offset	Name	Length	Format	Description
152(X'98')	SMF119AP_TSTCRxTim	4	Binary	Number of retransmitted timeouts
156(X'9C')	SMF119AP_TSTCRxMTU	4	Binary	Number of retransmitted Path MTU discovery packets
160(X'A0')	SMF119AP_TSTCPathM	4	Binary	Number of Path MTUs beyond retransmit limit
164(X'A4')	SMF119AP_TSTCDropPr	4	Binary	Number of TCP connections dropped due to probes
168(X'A8')	SMF119AP_TSTCDropKA	4	Binary	Number of TCP connections dropped by KeepAlive, including connections across SMC links
172(X'AC')	SMF119AP_TSTCDropF2	4	Binary	Number of TCP connections dropped because the FINWAIT2 timer expired before receiving FIN segments, including connections across SMC links
176(X'B0')	SMF119AP_TSTCDropRx	4	Binary	Number of TCP connections dropped due to retransmits
180(X'B4')	SMF119AP_TSTCEphPortExh	4	Binary	Number of bind() requests that failed because no TCP ephemeral ports were available
184(X'B8')	SMF119AP_TSTCEphPortAvail	2	Binary	Number of available TCP ephemeral ports
186(X'BA')	SMF119AP_TSTCEphPortInUse	2	Binary	Number of TCP ephemeral ports currently in use

Table 10. TCP statistics section (continued)

Offset	Name	Length	Format	Description
188(X'BC')	SMF119AP_TSTCEphPortMxUse	2	Binary	Maximum number of TCP ephemeral ports that are used
190(X'BE')	SMF119AP_TSrsvd1	2	Binary	Reserved
192(X'C0')	SMF119AP_TSSMCRCurrEstabLnks	4	Binary	Number of current active SMC-R links
196(X'C4')	SMF119AP_TSSMCRLnkActTimeOut	4	Binary	Number of SMC-R link activation attempts for which a timeout occurred
200(X'C8')	SMF119AP_TSSMCRActLnkOpened	4	Binary	Number of active SMC-R links that have been opened
204(X'CC')	SMF119AP_TSSMCRPasLnkOpened	4	Binary	Number of passive SMC-R links that have been opened
208(X'D0')	SMF119AP_TSSMCRLnksClosed	4	Binary	Number of SMC-R links that have been closed
212(X'D4')	SMF119AP_TSSMCRCurrEstab	4	Binary	Current number of TCP connections that are across SMC-R links
216(X'D8')	SMF119AP_TSSMCRActiveOpened	4	Binary	Number of active TCP connections that have been opened across SMC-R links
220(X'DC')	SMF119AP_TSSMCRPassiveOpened	4	Binary	Number of passive TCP connections that have been opened across SMC-R links
224(X'E0')	SMF119AP_TSSMCRConnClosed	4	Binary	Number of closed TCP connections that were across SMC-R links
228(X'E4')	SMF119AP_TSrsvd2	4	Binary	Reserved
232(X'E8')	SMF119AP_TSSMCRInSegs	8	Binary	Number of SMC-R inbound write operations
240(X'F0')	SMF119AP_TSSMCROutSegs	8	Binary	Number of SMC-R outbound write operations

Table 10. TCP statistics section (continued)

Offset	Name	Length	Format	Description
248(X'F8')	SMF119AP_TSSMCRInRsts	4	Binary	Number of SMC-R inbound write operations that contained the abnormal close flag
252(X'FC')	SMF119AP_TSSMCROutRsts	4	Binary	Number of SMC-R outbound write operations that contained the abnormal close flag
256(X'100')	SMF119AP_TSSMCDCurrEstabLnks	4	Binary	Number of current active SMC-D links
260(X'104')	SMF119AP_TSSMCDActLnkOpened	4	Binary	Number of active SMC-D links that have been opened
264(X'108')	SMF119AP_TSSMCDPasLnkOpened	4	Binary	Number of passive SMC-D links that have been opened
268(X'10C')	SMF119AP_TSSMCDLnksClosed	4	Binary	Number of SMC-D links that have been closed
272(X'110')	SMF119AP_TSSMCDCurrEstab	4	Binary	Current number of TCP connections that are across SMC-D links
276(X'114')	SMF119AP_TSSMCDActiveOpened	4	Binary	Number of active TCP connections that have been opened across SMC-D links
280(X'118')	SMF119AP_TSSMCDPassiveOpened	4	Binary	Number of passive TCP connections that have been opened across SMC-D links
284(X'11C')	SMF119AP_TSSMCDConnClosed	4	Binary	Number of closed TCP connections that were across SMC-D links
288(X'120')	SMF119AP_TSSMCDInSegs	8	Binary	Number of SMC-D inbound write operations
296(X'128')	SMF119AP_TSSMCDOutSegs	8	Binary	Number of SMC-D outbound write operations

Table 10. TCP statistics section (continued)				
Offset	**Name**	**Length**	**Format**	**Description**
304(X'130')	SMF119AP_TSSMCDInRsts	4	Binary	Number of SMC-D inbound write operations that contained the abnormal close flag
308(X'134')	SMF119AP_TSSMCDOutRsts	4	Binary	Number of SMC-D outbound write operations that contained the abnormal close flag

Table 11 on page 87 shows the UDP statistics section:

Table 11. UDP statistics section				
Offset	**Name**	**Length**	**Format**	**Description**
0(X'0')	SMF119AP_TSUDDuration	8	Binary	Duration of recording interval in microseconds, where bit 51 is equivalent to one microsecond
8(X'8')	SMF119AP_TSUDRecData	8	Binary	Number of UDP datagrams received
16(X'10')	SMF119AP_TSUDRecNoPort	4	Binary	Number of UDP datagrams received with no port defined
20(X'14')	SMF119AP_TSUDNoRec	4	Binary	Number of other UDP datagrams not received
24(X'18')	SMF119AP_TSUDXmtData	8	Binary	Number of UDP datagrams sent
32(X'20')	SMF119AP_TSUDEphPortExh	4	Binary	Number of bind() requests that failed because no UDP ephemeral ports were available
36(X'24')	SMF119AP_TSUDEphPortAvail	2	Binary	Number of available UDP ephemeral ports
38(X'26')	SMF119AP_TSUDEphPortInUse	2	Binary	Number of UDP ephemeral ports currently in use

Table 11. UDP statistics section (continued)				
Offset	**Name**	**Length**	**Format**	**Description**
40(X'28')	SMF119AP_TSUDEphPortMxUse	2	Binary	Maximum number of UDP ephemeral ports that are used

Table 12 on page 88 shows the ICMP statistics section:

Table 12. ICMP statistics section				
Offset	**Name**	**Length**	**Format**	**Description**
0(X'0')	SMF119AP_TSICDuration	8	Binary	Duration of recording interval in microseconds, where bit 51 is equivalent to one microsecond
8(X'8')	SMF119AP_TSICInMsg	4	Binary	Number of inbound ICMP messages
12(X'C')	SMF119AP_TSICInError	4	Binary	Number of inbound ICMP error messages
16(X'10')	SMF119AP_TSICInDstUnreach	4	Binary	Number of inbound ICMP destination unreachable messages
20(X'14')	SMF119AP_TSICInTimeExcd	4	Binary	Number of inbound ICMP time exceeded messages
24(X'18')	SMF119AP_TSICInParmProb	4	Binary	Number of inbound ICMP parameter problem messages
28(X'1C')	SMF119AP_TSICInSrcQuench	4	Binary	Number of inbound ICMP source quench messages
32(X'20')	SMF119AP_TSICInRedirect	4	Binary	Number of inbound ICMP redirect messages
36(X'24')	SMF119AP_TSICInEcho	4	Binary	Number of inbound ICMP echo request messages
40(X'28')	SMF119AP_TSICInEchoRep	4	Binary	Number of inbound ICMP echo reply messages
44(X'2C')	SMF119AP_TSICInTstamp	4	Binary	Number of inbound ICMP timestamp request messages
48(X'30')	SMF119AP_TSICInTstampRep	4	Binary	Number of inbound ICMP timestamp reply messages
52(X'34')	SMF119AP_TSICInAddrMask	4	Binary	Number of inbound ICMP address mask request messages
56(X'38')	SMF119AP_TSICInAddrMRep	4	Binary	Number of inbound ICMP address mask reply messages
60(X'3C')	SMF119AP_TSICOutMsg	4	Binary	Number of outbound ICMP messages

Table 12. ICMP statistics section (continued)				
Offset	**Name**	**Length**	**Format**	**Description**
64(X'40')	SMF119AP_TSICOutError	4	Binary	Number of outbound ICMP error messages
68(X'44')	SMF119AP_TSICOutDstUnreach	4	Binary	Number of outbound ICMP destination unreachable messages
72(X'48')	SMF119AP_TSICOutTimeExcd	4	Binary	Number of outbound ICMP time exceeded messages
76(X'4C')	SMF119AP_TSICOutParmProb	4	Binary	Number of outbound ICMP parameter problem messages
80(X'50')	SMF119AP_TSICOutSrcQuench	4	Binary	Number of outbound ICMP source quench messages
84(X'54')	SMF119AP_TSICOutRedirect	4	Binary	Number of outbound ICMP redirect messages
88(X'58')	SMF119AP_TSICOutEcho	4	Binary	Number of outbound ICMP echo request messages
92(X'5C')	SMF119AP_TSICOutEchoRep	4	Binary	Number of outbound ICMP echo reply messages
96(X'60')	SMF119AP_TSICOutTstamp	4	Binary	Number of outbound ICMP timestamp request messages
100(X'64')	SMF119AP_TSICOutTstampRep	4	Binary	Number of outbound ICMP timestamp reply messages
104(X'68')	SMF119AP_TSICOutAddrMask	4	Binary	Number of outbound ICMP address mask request messages
108(X'6C')	SMF119AP_TSICOutAddrMRep	4	Binary	Number of outbound ICMP address mask reply messages

Table 13 on page 89 shows the IPv6 IP statistics section:

Table 13. IPv6 IP statistics section				
Offset	**Name**	**Length**	**Format**	**Description**
0 (X'00')	SMF119AP_TSP6Duration	8	Binary	Duration of recording interval in microseconds, where bit 51 is equivalent to one microsecond
8(X'08')	SMF119AP_TSP6RecData	4	Binary	Number of IPv6 datagrams received
12(X'0C')	SMF119AP_TSP6DscData	4	Binary	Number of input IPv6 datagrams discarded due to errors in their IP header
16(X'10')	SMF119AP_TSP6DscAddr	4	Binary	Number of input IPv6 datagrams discarded because the IP address in their IP header's destination field was not valid

Table 13. IPv6 IP statistics section (continued)				
Offset	**Name**	**Length**	**Format**	**Description**
20(X'14')	SMF119AP_TSP6AttFwdData	4	Binary	Number of attempts to forward IPv6 datagrams
24(X'18')	SMF119AP_TSP6DscDUnkPr	4	Binary	Number of IPv6 datagrams discarded because of an unknown or unsupported protocol
28(X'1C')	SMF119AP_TSP6DscDOth	4	Binary	Number of input IPv6 datagrams discarded that are not accounted for in another input discard counter
32(X'20')	SMF119AP_TSP6DlvData	4	Binary	Number of IPv6 datagrams delivered
36(X'24')	SMF119AP_TSP6XData	4	Binary	Number of IPv6 datagrams transmitted
40(X'28')	SMF119AP_TSP6XDscOth	4	Binary	Number of IPv6 outbound datagrams discarded, due to reasons other than no route being available
44(X'2C')	SMF119AP_ TSP6XDscRoute	4	Binary	Number of IPv6 outbound datagrams discarded, due to no route being available
48(X'30')	SMF119AP_TSP6Timeouts	4	Binary	Number of IPv6 reassembly timeouts
52(X'34')	SMF119AP_ TSP6RecDRsmb	4	Binary	Number of received IPv6 datagrams requiring reassembly
56(X'38')	SMF119AP_TSP6Rsmb	4	Binary	Number of received IPv6 datagrams reassembled
60(X'3C')	SMF119AP_TSP6FailRsmb	4	Binary	Number of failed reassembly attempts on IPv6 datagrams
64(X'40')	SMF119AP_TSP6RecFgmt	4	Binary	Number of fragmented IPv6 datagrams received
68(X'44')	SMF119AP_ TSP6DscDFgmt	4	Binary	Number of IPv6 datagrams discarded due to fragmentation failure
72(X'48')	SMF119AP_TSP6XFgmt	4	Binary	Number of IPv6 datagram fragments generated
76(X'4C')	SMF119AP_ TSP6RouteDisc	4	Binary	Number of IPv6 routing discards

Table 14 on page 90 shows the IPv6 ICMP statistics section:

Table 14. IPv6 ICMP statistics section				
Offset	**Name**	**Length**	**Format**	**Description**
0 (X'00')	SMF119AP_TSC6Duration	8	Binary	Duration of recording interval in microseconds, where bit 51 is equivalent to one microsecond

Table 14. IPv6 ICMP statistics section (continued)

Offset	Name	Length	Format	Description
8(X'08')	SMF119AP_TSC6InMsg	4	Binary	Number of inbound IPv6 ICMP messages
12(X'0C')	SMF119AP_TSC6InError	4	Binary	Number of inbound IPv6 ICMP error messages
16(X'10')	SMF119AP_ TSC6InDstUnreach	4	Binary	Number of inbound IPv6 ICMP destination unreachable messages
20(X'14')	SMF119AP_ TSC6InTimeExcd	4	Binary	Number of inbound IPv6 ICMP time exceeded messages
24(X'18')	SMF119AP_ TSC6InParmProb	4	Binary	Number of inbound IPv6 ICMP parameter problem messages
28(X'1C')	SMF119AP_ TSC6InAdmProhib	4	Binary	Number of inbound IPv6 ICMP administratively prohibited messages
32(X'20')	SMF119AP_ TSC6InPktTooBig	4	Binary	Number of inbound IPv6 ICMP packet too big messages
36(X'24')	SMF119AP_TSC6InEcho	4	Binary	Number of inbound IPv6 ICMP echo request messages
40(X'28')	SMF119AP_ TSC6InEchoRep	4	Binary	Number of inbound IPv6 ICMP echo reply messages
44(X'2C')	SMF119AP_ TSC6InRtSolicit	4	Binary	Number of inbound IPv6 ICMP router solicitation messages
48(X'30')	SMF119AP_TSC6InRtAdv	4	Binary	Number of inbound IPv6 ICMP router advertisement messages
52(X'34')	SMF119AP_ TSC6InNbSolicit	4	Binary	Number of inbound IPv6 ICMP neighbor solicitation messages
56(X'38')	SMF119AP_TSC6InNbAdv	4	Binary	Number of inbound IPv6 ICMP neighbor advertisement messages
60(X'3C')	SMF119AP_ TSC6InRedirect	4	Binary	Number of inbound IPv6 ICMP redirect messages
64(X'40')	SMF119AP_TSC6InGrpMemQry	4	Binary	Number of inbound IPv6 ICMP multicast listener discovery membership query messages
68(X'44')	SMF119AP_ TSC6InGrpMemRsp	4	Binary	Number of inbound IPv6 ICMP multicast listener discovery membership reply messages
72(X'48')	SMF119AP_ TSC6InGrpMemRed	4	Binary	Number of inbound IPv6 ICMP multicast listener discovery membership reduction messages

Table 14. IPv6 ICMP statistics section (continued)

Offset	Name	Length	Format	Description
76(X'4C')	SMF119AP_TSC6OutMsg	4	Binary	Number of outbound IPv6 ICMP messages
80 (X'50')	SMF119AP_TSC6OutError	4	Binary	Number of outbound IPv6 ICMP error messages
84 (X'54')	SMF119AP_ TSC6OutDstUnrch	4	Binary	Number of outbound IPv6 ICMP destination unreachable messages
88 (X'58')	SMF119AP_ TSC6OutTimeExcd	4	Binary	Number of outbound IPv6 ICMP time exceeded messages
92 (X'5C')	SMF119AP_ TSC6OutParmProb	4	Binary	Number of outbound IPv6 ICMP parameter problem messages
96 (X'60')	SMF119AP_ TSC6OutAdmProhib	4	Binary	Number of outbound IPv6 ICMP administratively prohibited messages
100 (X'64')	SMF119AP_TSC6OutPktTooBig	4	Binary	Number of outbound IPv6 ICMP packet too big messages
104 (X'68')	SMF119AP_TSC6OutEcho	4	Binary	Number of outbound IPv6 ICMP echo request messages
108 (X'6C')	SMF119AP_ TSC6OutEchoRep	4	Binary	Number of outbound IPv6 ICMP echo reply messages
112 (X'70')	SMF119AP_ TSC6OutRtSolicit	4	Binary	Number of outbound IPv6 ICMP router solicitation messages
116 (X'74')	SMF119AP_TSC6OutRtAdv	4	Binary	Number of outbound IPv6 ICMP router advertisement messages
120 (X'78')	SMF119AP_ TSC6OutNbSolicit	4	Binary	Number of outbound IPv6 ICMP neighbor solicitation messages
124 (X'7C')	SMF119AP_ TSC6OutNbAdv	4	Binary	Number of outbound IPv6 ICMP neighbor advertisement messages
128 (X'80')	SMF119AP_ TSC6OutRedirect	4	Binary	Number of outbound IPv6 ICMP redirect messages
132 (X'84')	SMF119AP_ TSC6OutGrpMemQry	4	Binary	Number of outbound IPv6 ICMP multicast listener discovery membership query messages
136 (X'88')	SMF119AP_ TSC6OutGrpMemRsp	4	Binary	Number of outbound IPv6 ICMP multicast listener discovery membership report messages

Table 14. IPv6 ICMP statistics section (continued)				
Offset	**Name**	**Length**	**Format**	**Description**
140 (X'8C')	SMF119AP_ TSC6OutGrpMemRed	4	Binary	Number of outbound IPv6 ICMP multicast listener discovery membership reduction messages

Table 15 on page 93 shows the storage statistics section.

Table 15. Storage statistics section				
Offset	**Name**	**Length**	**Format**	**Description**
0(X'0')	SMF119AP_TSSTECSACurrent	8	Binary	Current number of ECSA storage bytes allocated
8(X'8')	SMF119AP_TSSTECSAFree	8	Binary	Current number of ECSA storage bytes allocated but not in use
16(X'10')	SMF119AP_TSSTPrivateCurrent	8	Binary	Current number of authorized private subpool storage bytes allocated
24(X'18')	SMF119AP_TSSTPrivateFree	8	Binary	Current number of authorized private subpool storage bytes allocated but not in use.
32(X'20')	SMF119AP_TSSTSMCRFixedCurrent	8	Binary	Current amount of fixed 64-bit storage bytes allocated for SMC-R
40(X'28')	SMF119AP_TSSTSMCRFixedMax	8	Binary	Maximum amount of fixed 64-bit storage bytes ever allocated for SMC-R
48(X'30')	SMF119AP_TSSTSMCRSendCurrent	8	Binary	Current amount of fixed 64-bit storage bytes allocated for SMC-R outbound processing
56(X'38')	SMF119AP_TSSTSMCRSendMax	8	Binary	Maximum amount of fixed 64-bit storage bytes ever allocated for SMC-R outbound processing
64(X'40')	SMF119AP_TSSTSMCRRecvCurrent	8	Binary	Current amount of fixed 64-bit storage bytes allocated for SMC-R inbound processing
72(X'48')	SMF119AP_TSSTSMCRRecvMax	8	Binary	Maximum amount of fixed 64-bit storage bytes ever allocated for SMC-R inbound processing
80(X'50')	SMF119AP_TSSTSMCDFixedCurrent	8	Binary	Current amount of fixed 64-bit storage bytes allocated for SMC-D

Table 15. Storage statistics section (continued)				
Offset	**Name**	**Length**	**Format**	**Description**
88(X'58')	SMF119AP_TSSTSMCDFixedMax	8	Binary	Maximum amount of fixed 64- bit storage bytes ever allocated for SMC-D
96(X'60')	SMF119AP_TSSTZAGGCURRENT	8	Binary	Current amount of fixed 64-bit storage bytes allocated for zERT Aggregation Records
104(X'68')	SMF119AP_TSSTZAGGMAX	8	Binary	Maximum amount of fixed 64-bit storage bytes allocated for zERT Aggregation Records

zERT Summary record (subtype 12)

zERT summary records function as both interval and event records for the z/OS Encryption Readiness Technology (zERT) aggregation function.

As interval records, the zERT summary records are generated at user specified intervals. The record provides statistical data about an individual security session that provided cryptographic protection for one or more TCP or Enterprise Extender (EE) connections during the previous recording interval. The record also provides information describing the cryptographic characteristics of the security session.

Each record reports statistical data about the security session for the previous recording interval. The starting and ending values for the previous recording interval are reported for each statistic.

If zERT aggregation is turned off dynamically or the TCP stack terminates, a final complete set of subtype 12 records is generated to report close out data. These records are reported to the z/OS System Management Facility or the real-time zERT Summary SMF NMI service, or both, depending on the SMF record destination in effect.

In addition, if recording of zERT summary records to the z/OS System Management Facility is turned off dynamically, a final complete set of subtype 12 records is reported to the z/OS System Management Facility to report close out data. No records are reported to the real-time zERT Summary SMF NMI service for this condition.

As event records, zERT summary records are written for two events:

- The zERT aggregation function is enabled.
- The zERT aggregation function is disabled dynamically.

The format of the zERT summary record is the same for both interval and event usage, although the zERT summary event records include just the TCP/IP Identification section and the zERT common section.

See Table 16 on page 95 for the contents of the TCP/IP stack identification section.

- For all zERT summary records, the TCP/IP stack identification section indicates STACK as the subcomponent.
- zERT summary event records indicate X'08' (event record) for the record reason.
- zERT summary interval records indicate one of three possible interval record reason settings, depending on whether the reporting is because of interval expiration, statistics collection termination, or collection shutdown.

Note : The interval data for a single security session is always contained within a single SMF record. Because of that, each SMF record is marked as “last record in set”.

Table 16 on page 95 shows the zERT summary record self-defining section:

Table 16. zERT summary record self-defining section				
Offset	**Name**	**Length**	**Format**	**Description**
0(X'0')	Standard SMF Header	24		Standard SMF header
Self-defining section				
24(X'18')	SMF119DS_TRN	2	Binary	Number of triplets in this record (6)
26(X'1A')		2	Binary	Reserved
28(X'1C')	SMF119IDOff	4	Binary	Offset to TCP/IP identification section
32(X'20')	SMF119IDLen	2	Binary	Length of TCP/IP identification section
34(X'22')	SMF119IDNum	2	Binary	Number of TCP/IP identification sections
36(X'24')	SMF119S1Off	4	Binary	Offset to zERT common section
40(X'28')	SMF119S1Len	2	Binary	Length of zERT common section
42(X'2A')	SMF119S1Num	2	Binary	Number of zERT common section
44(X'2C')	SMF119S2Off	4	Binary	Offset to TLS-specific section
48(X'30')	SMF119S2Len	2	Binary	Length of TLS-specific section
50(X'32')	SMF119S2Num	2	Binary	Number of TLS section
52(X'34')	SMF119S3Off	4	Binary	Offset to SSH-specific section
56(X'38')	SMF119S3Len	2	Binary	Length of SSH-specific section
58(X'3A')	SMF119S3Num	2	Binary	Number of SSH-specific sections
60(X'3C')	SMF119S4Off	4	Binary	Offset to IPSec-specific section
64(X'40')	SMF119S4Len	2	Binary	Length of IPSec-specific section
66(X'42')	SMF119S4Num	2	Binary	Number of IPSec-specific section
68(X'44')	SMF119S5Off	4	Binary	Offset to certificate DN section
72(X'48')	SMF119S5Len	2	Binary	Length of certificate DN section
74(X'4A')	SMF119S5Num	2	Binary	Number of certificate DN section

Table 17 on page 95 shows the zERT summary common section. Every zERT summary record has one of these sections.

Unless noted in the field description, all TCP and Enterprise Extender (EE) connection statistics reported in the common section represent activity from the time the zERT aggregation function began tracking this security session until the time that the zERT aggregation function stops tracking it. The zERT aggregation function stops tracking a security session when one complete SMF/INTVAL recording interval passes without any connections being protected by the security session. The TCP and Enterprise Extender (EE) connection statistics counts are approximate.

Table 17. zERT summary record common section				
Offset	**Name**	**Length**	**Format**	**Description**
0(X'0')	SMF119SS_SAIntervalDuration	8	Binary	Duration of recording interval in microseconds, where bit 51 is equivalent to 1 microsecond.

Table 17. zERT summary record common section (continued)

Offset	Name	Length	Format	Description
8(X'8')	SMF119SS_SAEvent_Type	1	Binary	Event type: 1. Summary interval record 2. zERT aggregation function enabled event record 3. zERT aggregation function disabled event record
9(X'9')	SMF119SS_SAFlags	1	Binary	Flags: • X'80': The session uses IPv6 addresses • X'40': The local socket of this session is acting as the server (only meaningful when SMF119SS_SAIPProto indicates TCP) • X'20': The local socket of this session is acting as the client (only meaningful when SMF119SS_SAIPProto indicates TCP) • X'10': This security session represents Enterprise Extender connections (only meaningful when SMF119SS_SAIPProto indicates UDP) • X'08': This security session represents IPv4 outbound data connections that are established by the FTP server to the FTP client. • X'04': AT-TLS cryptographic data protection operations are bypassed for this security session as part of a stack optimization for intra-host connections. Only AT-TLS peer authentication operations are executed in this case. • X'02': A zERT Aggregation recording interval separate from the SMF interval is specified with the GLOBALCONFIG ZERT AGGREGATION INTVAL parameter.

Table 17. zERT summary record common section (continued)

Offset	Name	Length	Format	Description
10(X'A')	SMF119SS_SASecProtos	1	Binary	Cryptographic security protocol. Only one value is set. Possible values are: • X'00': No recognized cryptographic protection • X'80': TLS/SSL • X'40': SSH • X'20': IPSec
11(X'B')	SMF119SS_SAJobname	8	EBCDIC	Jobname that is associated with the socket.
19(X'13')	SMF119SS_SAUserID	8	EBCDIC	z/OS user ID associated with the socket **Note :** The value *FTPUSR* is specified when this security session represents an aggregation of FTP data connections and we are reporting at the FTP server (SMF119SS_SAFlags = x'40').
27(X'1B')	SMF119SS_SAIPProto	1	Binary	IP Protocol value. Possible values are: • 6: TCP • 17: UDP
28(X'1C')	SMF119SS_SASrvIP	16	Binary	Server IP address. If SMF119SS_Flags indicates IPv6, then this is a 16-byte IPv6 address. Otherwise, it is a 4-byte IPv4 address in the first 4 bytes of the field.
44(X'2C')	SMF119SS_SACltIP	16	Binary	Client IP address. If SMF119SS_Flags indicates IPv6, then this is a 16-byte IPv6 address. Otherwise, it is a 4-byte IPv4 address in the first 4 bytes of the field.
60(X'3C')	SMF119SS_SASrvPortStart	2	Binary	Starting value for server port range. For information on this field, see How does zERT aggregation determine the server port? in z/OS Communications Server: IP Configuration Guide.
62(X'3E')	SMF119SS_SASrvPortEnd	2	Binary	Ending value for server port range. If this security session represents a single-server port, then the ending value equals the starting value for the port range.

Table 17. zERT summary record common section (continued)

Offset	Name	Length	Format	Description
64(X'40')	SMF119SS_SASessionID	42	EBCDIC	Session identifier that uniquely identifies a security session based on the server and client endpoints plus the significant security attributes for the session. The session identifier is in the form *p-value*, where • *p* represents the cryptographic protocol. Possible values for *p* are: – C = No recognized cryptographic protection – I = IPSec – T = TLS/SSL – S = SSH • "-" is a separator character • *value* is a 20-character hexadecimal string
106(X'6A')		2		Reserved (alignment)
108(X'6C')	SMF119SS_SAInitLifeConnCnt	4	Binary	Count of connections for the life of this security session at the beginning of the summary interval.
112(X'70')	SMF119SS_SAInitLifePartialConnCnt	4	Binary	Count of the partial connections for the life of this security session at the beginning of the summary interval. This is a subset of the connections reported in SMF119SS_SAInitLifeConnCnt. A connection is considered to be a "partial connection" if one or more of these conditions is met: • The connection was in existence before it was associated with this security session • The security session stopped being associated with the connection, but the connection continued to exist.
116(X'74')	SMF119SS_SAInitLifeShortConnCnt	4	Binary	Count of short connections for the life of this security session at the beginning of the summary interval. Short connections are connections that last less than 10 seconds. This value is only meaningful when SMF119SS_SAIPProto indicates TCP.

Table 17. zERT summary record common section (continued)

Offset	Name	Length	Format	Description
120(X'78')	SMF119SS_ SAInitActiveConnCnt	4	Binary	Number of active connections that are associated with this security session at the beginning of the summary interval.
124(X'7C')	SMF119SS_SAInitLifeInBytes	8	Binary	Inbound byte count for the life of this security session at the beginning of the summary interval.
132(X'84')	SMF119SS_SAInitLifeOutBytes	8	Binary	Outbound byte count for the life of this security session at the beginning of the summary interval.
140(X'8C')	SMF119SS_SAInitLifeInSegDG	8	Binary	Inbound TCP segment or UDP datagram count for the life of this security session at the beginning of the summary interval.
148(X'94')	SMF119SS_ SAInitLifeOutSegDG	8	Binary	Outbound TCP segment or UDP datagram count for the life of this security session at the beginning of the summary interval.
156(X'9C')	SMF119SS_SAEndLifeConnCnt	4	Binary	Count of connections for the life of this security session at the end of the summary interval.
160(X'A0')	SMF119SS_ SAEndLifePartialConnCnt	4	Binary	Count of partial connections for the life of this security session at the end of the summary interval. This is a subset of the connections reported in SMF119SS_SAEndLifeConnCnt that were associated with the security session for only part of their existence, using the same conditions described for SMF119SS_SAInitLifePartialConnCnt.
164(X'A4')	SMF119SS_ SAEndLifeShortConnCnt	4	Binary	Count of short connections for the life of this security session at the end of the summary interval. Short connections are ones that last less than 10 seconds. This value is only meaningful when SMF119SS_SAIPProto indicates TCP.
168(X'A8')	SMF119SS_ SAEndActiveConnCnt	4	Binary	Number of active connections that are associated with this security session at the end of the summary interval.
172(X'AC')	SMF119SS_SAEndLifeInBytes	8	Binary	Inbound byte count for the life of this security session at the end of the summary interval.

Table 17. zERT summary record common section (continued)				
Offset	**Name**	**Length**	**Format**	**Description**
180(X'B4')	SMF119SS_SAEndLifeOutBytes	8	Binary	Outbound byte count for the life of this security session at the end of the summary interval.
188(X'BC')	SMF119SS_SAEndLifeInSegDG	8	Binary	Inbound TCP segment or UDP datagram count for the life of this security session at the end of the summary interval.
196(X'C4')	SMF119SS_ SAEndLifeOutSegDG	8	Binary	Outbound TCP segment or UDP datagram count for the life of this security session at the end of the summary interval.

Table 18 on page 100 shows the zERT summary TLS protocol attributes section. This section is presented in a zERT summary interval record when the SMF119SS_SecProto field of the zERT summary common section indicates that this is a TLS or SSL security session (that is, when it contains the value X'80'):

Table 18. zERT summary record TLS protocol attributes section				
Offset	**Name**	**Length**	**Format**	**Description**
0(X'0')	SMF119SS_TLS_Source	1	Binary	Source of the information in this record. Can be one of the following values: • X'01': Stream observation • X'02': Cryptographic protocol provider
1(X'1')	SMF119SS_TLS_CryptoFlags	1	Binary	Cryptographic operations flags: • X'80': Encrypt-then-MAC processing is used • X'40': Raw public key authentication is used • X'10': Pre-shared key authentication is used
2(X'2')	SMF119SS_TLS_Prot_Ver	2	Binary	Protocol version: • X'0000': Unknown version • X'0200': SSLv2 • X'0300': SSLv3 • X'0301': TLSv1.0 • X'0302': TLSv1.1 • X'0303': TLSv1.2 • X'0304': TLSv1.3

Table 18. zERT summary record TLS protocol attributes section (continued)

Offset	Name	Length	Format	Description
4(X'4')	SMF119SS_TLS_Neg_Cipher	6	EBCDIC	Negotiated cipher suite identifier. • If the TLS version is SSLv3 or higher, this is a four character value in the first 4 bytes of this field. Refer to the TLS Cipher Suite registry at http://www.iana.org/assignments/tls-parameters/tls-parameters.xhtml for a complete list of the 4-hexadecimal-character values. • If the TLS version is SSLv2, then all 6 bytes are used: – 010080: 128-bit RC4 with MD5 – 020080: 40-bit RC4 with MD5 – 030080: 128-bit RC2 with MD5 – 040080: 40-bit RC2 with MD5 – 050080: 128-bit IDEA with MD5 – 060040: DES with MD5 – 0700C0: 3DES with MD5
10(X'A') (continued)				• X'0025': Camellia 128 CBC • X'0026': Camellia 256 CBC • X'0027': Camellia 128 GCM • X'0028': Camellia 256 GCM • X'0029': ChaCha20 Poly1305 • X'002A': IDEA CBC • X'002B': SEED CBC • X'002C': Fortezza • X'002D': GOST28147 • X'002E': TwoFish CBC 256 • X'002F': TwoFish CBC • X'0030': TwoFish CBC 192 • X'0031': TwoFish CBC 128 • X'0032': Serpent CBC 256 • X'0033': Serpent CBC 192 • X'0034': Serpent CBC 128

Table 18. zERT summary record TLS protocol attributes section (continued)

Offset	Name	Length	Format	Description
10(X'A')	SMF119SS_TLS_CS_Enc_Alg	2	Binary	The symmetric encryption algorithm that is used by the cipher suite: • X'0000': Unknown • X'0001': None • X'0002': DES • X'0003': DES 40 • X'0004': 3DES • X'0005': RC2 40 • X'0006': RC2 128 • X'0007': RC2 • X'0008': RC4 40 • X'0009': RC4 128 • X'000A': RC4 256 • X'000B': RC4 • X'000C': AES CBC 128 • X'000D': AES CBC 192 • X'000E': AES CBC 256 • X'000F': AES CTR 128 • X'0010': AES CTR 192 • X'0011': AES CTR 256 • X'0012': AES GCM 128 • X'0013': AES GCM 256 • X'0014': AES CCM 128 • X'0015': AES CCM 256 • X'0016': AES CCM8 128 • X'0017': AES CCM8 256 • X'0018': AES 256 • X'0019': Blowfish • X'001A': Blowfish CBC • X'001B': CAST 128 CBC • X'001C': ARCFOUR 128 • X'001D': ARCFOUR 256 • X'001E': ARCFOUR • X'001F': Rijndael CBC • X'0020': ACSS • X'0021': ARIA 128 CBC • X'0022': ARIA 256 CBC • X'0023': ARIA 128 GCM • X'0024': ARIA 256 GCM

Table 18. zERT summary record TLS protocol attributes section (continued)

Offset	Name	Length	Format	Description
10(X'A') (continued)				• X'0025': Camellia 128 CBC • X'0026': Camellia 256 CBC • X'0027': Camellia 128 GCM • X'0028': Camellia 256 GCM • X'0029': ChaCha20 Poly1305
10(X'A')	SMF119SS_TLS_CS_Enc_Alg	2	Binary	The symmetric encryption algorithm that is used by the cipher suite: • X'002A': IDEA CBC • X'002B': SEED CBC • X'002C': Fortezza • X'002D': GOST28147 • X'002E': TwoFish CBC 256 • X'002F': TwoFish CBC • X'0030': TwoFish CBC 192 • X'0031': TwoFish CBC 128 • X'0032': Serpent CBC 256 • X'0033': Serpent CBC 192 • X'0034': Serpent CBC 128

Table 18. zERT summary record TLS protocol attributes section (continued)

Offset	Name	Length	Format	Description
12(X'C')	SMF119SS_TLS_CS_Msg_Auth	2	Binary	The message authentication algorithm that is used by the cipher suite: • X'0000': Unknown • X'0001': No message authentication, or uses authenticated encryption algorithm like AES-GCM • X'0002': MD2 • X'0003': HMAC-MD5 • X'0004': HMAC-SHA1 • X'0005': HMAC-SHA2-224 • X'0006': HMAC-SHA2-256 • X'0007': HMAC-SHA2-384 • X'0008': HMAC-SHA2-512 • X'0009': AES-GMAC-128 • X'000A': AES-GMAC-256 • X'000B': AES-128-XCBC-96 • X'000C': HMAC-SHA2-256-128 • X'000D': HMAC-SHA2-384-192 • X'000E': HMAC-SHA2-512-256 • X'000F': HMAC-MD5-96 • X'0010': HMAC-SHA1-96 • X'0011': UMAC-64 • X'0012': UMAC-128 • X'0013': RIPEMD-160

Table 18. zERT summary record TLS protocol attributes section (continued)

Offset	Name	Length	Format	Description
14(X'E')	SMF119SS_TLS_CS_Kex_Alg	2	Binary	The key exchange algorithm that is used by the cipher suite: • X'0000': Unknown • X'0001': None • X'0002': RSA • X'0003': RSA_EXPORT • X'0004': RSA_PSK • X'0005': DH_RSA • X'0006': DH_RSA_EXPORT • X'0007': DH_DSS • X'0008': DH_ANON • X'0009': DH_ANON_EXPORT • X'000A': DH_DSS_EXPORT • X'000B': DHE_RSA • X'000C': DHE_RSA_EXPORT • X'000D': DHE_DSS • X'000E': DHE_DSS_EXPORT • X'000F': DHE_PSK • X'0010': ECDH_ECDSA • X'0011': ECDH_RSA • X'0012': ECDH_ANON • X'0013': ECDHE_ECDSA • X'0014': ECDHE_RSA • X'0015': ECDHE_PSK • X'0016': KRB5 • X'0017': KRB5_EXPORT • X'0018': PSK • X'0019': SRP_SHA_RSA • X'001A': SRP_SHA_DSS • X'001B': SRP_SHA • X'001C': ECDHE • X'001D': DHE
Server certificate information				

Table 18. zERT summary record TLS protocol attributes section (continued)

Offset	Name	Length	Format	Description
16(X'10')	SMF119SS_TLS_SCert _Signature_Method	2	Binary	Server certificate signature method: • X'0000': Unknown • X'0001': None • X'0002': RSA with MD2 • X'0003': RSA with MD5 • X'0004': RSA with SHA1 • X'0005': DSA with SHA1 • X'0006': RSA with SHA-224 • X'0007': RSA with SHA-256 • X'0008': RSA with SHA-384 • X'0009': RSA with SHA-512 • X'000A': ECDSA with SHA1 • X'000B': ECDSA with SHA-224 • X'000C': ECDSA with SHA-256 • X'000D': ECDSA with SHA-384 • X'000E': ECDSA with SHA-512 • X'000F': DSA with SHA-224 • X'0010': DSA with SHA-256 • X'0011': RSA PSS RSAE with SHA-256 • X'0012': RSA PSS RSAE with SHA-384 • X'0013': RSA PSS RSAE with SHA-512 • X'0014': ED 25519 • X'0015': ED 448 • X'0016': RSA PSS PSS with SHA-256 • X'0017': RSA PSS PSS with SHA-384 • X'0018': RSA PSS PSS with SHA-512
18(X'12')	SMF119SS_TLS_SCert _Enc_Method	2	Binary	Server certificate encryption method: • X'0000': Unknown • X'0001': None • X'0002': RSA • X'0003': DSA • X'0004': ECDSA

Table 18. zERT summary record TLS protocol attributes section (continued)

Offset	Name	Length	Format	Description
20(X'14')	SMF119SS_TLS_SCert _Digest_Alg	2	Binary	Server certificate digest algorithm: • X'0000': Unknown • X'0001': None • X'0002': MD2 • X'0003': MD5 • X'0004': SHA1 • X'0005': SHA-224 • X'0006': SHA-256 • X'0007': SHA-384 • X'0008': SHA-512
22(X'16')	SMF119SS_TLS_SCert _Key_Type	2	Binary	Server certificate key type: • X'0000': Unknown • X'0001': None • X'0002': RSA • X'0003': DSA • X'0004': Diffie-Hellman (DH) • X'0005': Elliptic Curve Cryptography (ECC)
24(X'18')	SMF119SS_TLS_SCert _Key_Len	2	Binary	Server certificate key length
Client certificate information				
26(X'1A')	SMF119SS_TLS_CCert _Signature_Method	2	Binary	Client certificate signature method. Same values as SMF119SS_TLS_SCert_Signature _Method.
28(X'1C')	SMF119SS_TLS_CCert _Enc_Method	2	Binary	Client certificate encryption method. Same values as SMF119SS_TLS_SCert_Enc_Method.
30(X'1E')	SMF119SS_TLS_CCert _Digest_Alg	2	Binary	Client certificate digest algorithm. Same values as SMF119SS_TLS_SCert_Digest_Alg.
32(X'20')	SMF119SS_TLS_CCert _Key_Type	2	Binary	Client certificate key type. Same values as SMF119SS_TLS_SCert_Key_Type.
34(X'22')	SMF119SS_TLS_CCert _Key_Len	2	Binary	Client certificate key length
Additional connection specific information				

Table 18. zERT summary record TLS protocol attributes section (continued)

Offset	Name	Length	Format	Description
36(X'24')	SMF119SS_TLS_ Server_HS_Sig_Method	2	Binary	Server-specified signature method used to encrypt certain TLS handshake messages. Same values as SMF119SS_TLS_SCert_Signature_Method. **Note :** Only valid for TLSv1.2 and later connections.
38(X'26')	SMF119SS_TLS_ Client_HS_Sig_Method	2	Binary	Client-specified signature method used to encrypt certain TLS handshake messages. Same values as SMF119SS_TLS_SCert_Signature_Method. **Note :** Only valid for TLSv1.2 and later connections.
40(X'28')	SMF119SS_TLS_ Neg_Key_Share	2	Binary	Negotiated key share: • X'0000': Unknown • X'0001': None • X'0002': SECP-256R1 • X'0003': SECP-384R1 • X'0004': SECP-521R1 • X'0005': X-25519 • X'0006': X-448 • X'0007': FFDHE with 2048 • X'0008': FFDHE with 3072 • X'0009': FFDHE with 4096 • X'000A': FFDHE with 6144 • X'000B': FFDHE with 8192
TLSv1.3 specific information				
36(X'24')	SMF119SS_TLS_ Handshake_Sig_Method	2	Binary	Signature method used for the handshake certificate. Same values as SMF119SS_TLS_SCert_Signature_Method.

Table 18. zERT summary record TLS protocol attributes section (continued)				
Offset	**Name**	**Length**	**Format**	**Description**
38(X'26')	SMF119SS_TLS_ Neg_Key_Share	2	Binary	Negotiated key share: • X'0000': Unknown • X'0001': None • X'0002': SECP-256R1 • X'0003': SECP-384R1 • X'0004': SECP-521R1 • X'0005': X-25519 • X'0006': X-448 • X'0007': FFDHE with 2048 • X'0008': FFDHE with 3072 • X'0009': FFDHE with 4096 • X'000A': FFDHE with 6144 • X'000B': FFDHE with 8192

Table 19 on page 109 shows the zERT summary SSH protocol attributes section. This section is presented in a zERT summary interval record when the SMF119SS_SecProto field of the zERT summary common section indicates that this is an SSH security session (i.e., when it contains the value X'40'):

Table 19. zERT summary record SSH protocol attributes section				
Offset	**Name**	**Length**	**Format**	**Description**
0(X'0')	SMF119SS_SSH_Source	1	Binary	Source of the information in this record. Can be one of the following values: • X'01': Stream observation • X'02': Cryptographic protocol provider
1(X'1')		1		Unused
2(X'2')	SMF119SS_SSH_Prot_Ver	1	Binary	Protocol version : 1. Protocol version 1 2. Protocol version 2
3(X'3')	SMF119SS_SSH_CryptoFlags	1	Binary	Cryptographic operations flags: • X'80': Encrypt-then-MAC processing is used for inbound traffic • X'40': Encrypt-then-MAC processing is used for outbound traffic

Table 19. zERT summary record SSH protocol attributes section (continued)

Offset	Name	Length	Format	Description
4(X'4')	SMF119SS_SSH_Auth_Method	2	Binary	First or only peer authentication method that is used for this security session: • X'0000': Unknown • X'0001': None • X'0002': Password • X'0003': Public key • X'0004': Host-based • X'0005': Rhosts • X'0006': RhostsRSA • X'0007': RSA • X'0008': Keyboard-interactive • X'0009': Challenge-response • X'000A': Control socket 1 • X'000B': GSSAPI with MIC • X'000C': GSSAPI Key exchange
6(X'6')	SMF119SS_SSH_Auth_Method2	2	Binary	If not 0, the last of multiple authentication methods used for this connection. Values are the same as those for SMF119SS_SSH_Auth_Method
8(X'8')	SMF119SS_SSH_In_Enc_Alg	2	Binary	Encryption algorithm for inbound traffic. Same values as SMF119SS_TLS_CS_Enc_Alg in Table 18 on page 100.
10(X'A')	SMF119SS_SSH_In_Msg_Auth	2	Binary	Message authentication algorithm for inbound traffic. Same values as SMF119SS_TLS_CS_Msg_Auth in Table 18 on page 100.

Table 19. zERT summary record SSH protocol attributes section (continued)

Offset	Name	Length	Format	Description
12(X'C')	SMF119SS_SSH_Kex_Method	2	Binary	Key exchange method. • X'0000' Unknown • X'0001' None • X'0002' Diffie-Hellman-group-exchangeSHA256 • X'0003' Diffie-Hellman-group-exchangeSHA1 • X'0004' Diffie-Hellman-group14-SHA1 • X'0005' Diffie-Hellman-group1-SHA1 • X'0006' ECDH-SHA2-NISTP256 • X'0007' ECDH-SHA2-NISTP384 • X'0008' ECDH-SHA2-NISTP521 • X'0009' GSS-GROUP1-SHA1 • X'000A' GSS-GROUP14-SHA1 • X'000B' GSS-GEX-SHA1 • X'000C' ECMQV-SHA2 • X'000D' GSS-* • X'000E' RSA1024-SHA1 • X'000F' RSA2048-SHA256 • X'0010' Diffie-Hellman-group14-SHA256 • X'0011' Diffie-Hellman-group16-SHA512 • X'0012' Diffie-Hellman-group18-SHA512 • X'0013' Curve 25519-SHA256
14(X'E')	SMF119SS_SSH_Out_Enc_Alg	2	Binary	Encryption algorithm for outbound traffic. Same values as SMF119SS_TLS_CS_Enc_Alg in Table 18 on page 100.
16(X'10')	SMF119SS_SSH_Out_Msg_Auth	2	Binary	Message authentication algorithm for outbound traffic. Same values as SMF119SS_TLS_CS_Msg_Auth in Table 18 on page 100.

Table 19. zERT summary record SSH protocol attributes section (continued)

Offset	Name	Length	Format	Description
18(X'12')	SMF119SS_SSH_SKey_Type	2	Binary	Type of raw server key: • X'0000': Unknown • X'0001': None • X'0002': RSA • X'0003': DSA • X'0004': Diffie-Hellman (DH) • X'0005': Elliptic Curve Cryptography (ECC) • X'0006': RSA1 (SSHV1 only) • X'0007': RSA_CERT (from OpenSSH certificate) • X'0008': DSA_CERT (from OpenSSH certificate) • X'0009': ECDSA_CERT (from OpenSSH certificate) • X'000A': ED 25519 • X'000B': ED 25519 (from OpenSSH certificate)
20(X'14')	SMF119SS_SSH_SKey_Len	2	Binary	Length of raw server key in bits.
22(X'16')	SMF119SS_SSH_CKey_Type	2	Binary	Type of raw client key. Same values as SMF119SS_SSH_Server_Key_Type.
24(X'18')	SMF119SS_SSH_CKey_Len	2	Binary	Length of raw client key in bits.
Server X.509 certificate information				
26(X'1A')	SMF119SS_SSH_ SCert_Signature_Method	2	Binary	Server certificate signature method. Same values as SMF119SS_TLS_SCert_Signature_Method in Table 18 on page 100.
28(X'1C')	SMF119SS_SSH_SCert _Enc_Method	2	Binary	Server certificate encryption method. Same values as SMF119SS_TLS_SCert_Enc_Method in Table 18 on page 100.
30(X'1E')	SMF119SS_SSH_SCert _Digest_Alg	2	Binary	Server certificate digest algorithm. Same values as SMF119SS_TLS_SCert_Digest_Alg in Table 18 on page 100.
32(X'20')	SMF119SS_SSH_SCert _Key_Type	2	Binary	Server certificate key type. Same values as SMF119SS_TLS_SCert_Key_Type in Table 18 on page 100.
34(X'22')	SMF119SS_SSH_SCert _Key_Len	2	Binary	Server certificate key length
Client X.509 certificate information				

Table 19. zERT summary record SSH protocol attributes section (continued)				
Offset	**Name**	**Length**	**Format**	**Description**
36(X'24')	SMF119SS_SSH_CCert_Signature_Method	2	Binary	Client certificate signature method. Same values as SMF119SS_TLS_SCert_Signature_Method in Table 18 on page 100.
38(X'26')	SMF119SS_SSH_CCert_Enc_Method	2	Binary	Client certificate encryption method. Same values as SMF119SS_TLS_SCert_Enc_Method in Table 18 on page 100.
40(X'28')	SMF119SS_SSH_CCert_Digest_Alg	2	Binary	Client certificate digest algorithm. Same values as SMF119SS_TLS_SCert_Digest_Alg in Table 18 on page 100.
42(X'2A')	SMF119SS_SSH_CCert_Key_Type	2	Binary	Client certificate key type. Same values as SMF119SS_TLS_SCert_Key_Type in Table 18 on page 100.
44(X'2C')	SMF119SS_SSH_CCert_Key_Len	2	Binary	Client certificate key length

Table 20 on page 113 shows the zERT summary IPSec attributes section. This section is presented in a zERT summary interval record when the SMF119SS_SecProto field of the zERT summary common section indicates that this is an IPSec security session (that is, when it contains the value X'20'):

Table 20. zERT summary record IPSec protocol attributes section				
Offset	**Name**	**Length**	**Format**	**Description**
0(X'0')	SMF119SS_IPSec_IKEMajVer	1	Binary	Major version of the IKE protocol in use. Only the low-order 4 bits are used.
1(X'1')	SMF119SS_IPSec_IKEMinVer	1	Binary	Minor version of the IKE protocol in use. Only the low-order 4 bits are used.
2(X'2')	SMF119SS_IPSec_IKETunLclEndpt	16	Binary	Local IP address of tunnel endpoint. If SMF119SS_SAFlags indicates IPv6, then this is a 16-byte IPv6 address. Otherwise, it is a 4-byte IPv4 address in the first 4 bytes of the field.
18(X'12')	SMF119SS_IPSec_IKETunRmtEndpt	16	Binary	Remote IP address of tunnel endpoint. If SMF119SS_SAFlags indicates IPv6, then this is a 16-byte IPv6 address. Otherwise, it is a 4-byte IPv4 address in the first 4 bytes of the field.

Table 20. zERT summary record IPSec protocol attributes section (continued)				
Offset	**Name**	**Length**	**Format**	**Description**
34(X'22')	SMF119SS_IPSec _IKETunLclAuthMeth	2	Binary	The authentication method for the local endpoint. One of the following values: • 0: Unknown • 1: None • 2: RSA signature • 3: Preshared key • 4: ECDSA-256 signature • 5: ECDSA-384 signature • 6: ECDSA-521 signature • 7: Digital signature
36(X'24')	SMF119SS_IPSec _IKETunRmtAuthMeth	2	Binary	The authentication method for the remote endpoint. Same values as SMF119SS_IPSec_IKETunLclAuthMeth.
38(X'26')	SMF119SS_IPSec _IKETunAuthAlg	2	Binary	Tunnel authentication algorithm. Same values as SMF119SS_TLS_CS_Msg_Auth in Table 18 on page 100.
40(X'28')	SMF119SS_IPSec _IKETunEncAlg	2	Binary	Tunnel encryption algorithm. Same values as SMF119SS_TLS_CS_Enc_Alg in Table 18 on page 100.
42(X'2A')	SMF119SS_IPSec _IKETunDHGroup	2	Binary	Diffie-Hellman group that is used to generate the keying material for this IKE tunnel. One of the following values: • X'00': Unknown or manual tunnel • X'01': Group1 • X'02': Group 2 • X'05': Group 5 • X'0E': Group 14 • X'13': Group 19 • X'14': Group 20 • X'15': Group 21 • X'18': Group 24 • X'FF': No DH group used (only possible for SMF119SS_IPSec_PFSGroup, where these values are also used)

Table 20. zERT summary record IPSec protocol attributes section (continued)

Offset	Name	Length	Format	Description
44(X'2C')	SMF119SS_IPSec _IKETunPseudoRandFunc	2	Binary	Pseudo-random function that is used for seeding keying material. One of the following values: • 0: Unknown • 1: None • 2: HMAC-SHA2-256 • 3: HMAC-SHA2-384 • 4: HMAC-SHA2-512 • 5: AES-128-XCBC • 6: HMAC-MD5 • 7: HMAC-SHA1
IKE Local certificate information. This information is populated if SMF119SS_IPSec_IKETunLocalAuthMeth is not “preshared key” (or not a value of 3). Otherwise, all fields are set to zero.				
46(X'2E')	SMF119SS_IPSec_LclCert _Sign_Meth	2	Binary	Local IKE certificate signature method. Same values as SMF119SS_TLS_SCert_Signature_Method in Table 18 on page 100.
48(X'30')	SMF119SS_IPSec_LclCert _Enc_Meth	2	Binary	Local IKE certificate encryption method. Same values as SMF119SS_TLS_SCert_Enc_Method in Table 18 on page 100.
50(X'32')	SMF119SS_IPSec_LclCert _Digest_Alg	2	Binary	Local IKE certificate digest algorithm. Same values as SMF119SS_TLS_SCert_Digest_Alg in Table 18 on page 100.
52(X'34')	SMF119SS_IPSec_LclCert _Key_Type	2	Binary	Local IKE certificate key type. Same values as SMF119SS_TLS_SCert_Key_Type in Table 18 on page 100.
54(X'36')	SMF119SS_IPSec_LclCert _Key_Len	2	Binary	Local IKE certificate key length in bits
IKE Peer certificate information. This information is populated if SMF119SS_IPSec_IKETunRmtAuthMeth is not “preshared key” (or not a value of 3). Otherwise, all fields set to zero.				
56(X'38')	SMF119SS_IPSec_RmtCert _Sign_Meth	2	Binary	Remote IKE certificate signature method. Same values as SMF119SS_TLS_SCert_Signature_Method in Table 18 on page 100.
58(X'3A')	SMF119SS_IPSec_RmtCert _Enc_Meth	2	Binary	Remote IKE certificate encryption method. Same values as SMF119SS_TLS_SCert_Enc_Method in Table 18 on page 100.
60(X'3C')	SMF119SS_IPSec_RmtCert _Digest_Alg	2	Binary	Remote IKE certificate digest algorithm. Same values as SMF119SS_TLS_SCert_Digest_Alg in Table 18 on page 100.

Table 20. zERT summary record IPSec protocol attributes section (continued)				
Offset	**Name**	**Length**	**Format**	**Description**
62(X'3E')	SMF119SS_IPSec_RmtCert_Key_Type	2	Binary	Remote IKE certificate key type. Same values as SMF119SS_TLS_SCert_Key_Type in Table 18 on page 100.
64(X'40')	SMF119SS_IPSec_RmtCert_Key_Len	2	Binary	Remote IKE certificate key length in bits
IPsec (Phase 2) tunnel information				
66(X'42')	SMF119SS_IPSec_PFSGroup	2	Binary	Diffie-Hellman group that is used for perfect forward secrecy. Same values as SMF119SS_IPSec_IKETunDHGroup.
68(X'44')	SMF119SS_IPSec_EncapMode	1	Binary	Tunnel encapsulation mode. One of the following values: 1. Tunnel Mode 2. Transport Mode
69(X'45')	SMF119SS_IPSec_AuthProto	1	Binary	The protocol that is used for message authentication. One of the following values: • 50 Encapsulating Security Payload (ESP) • 51: Authentication Header (AH)
70(X'46')	SMF119SS_IPSec_AuthAlg	2	Binary	The tunnel authentication algorithms. Same values as SMF119SS_TLS_CS_Msg_Auth in Table 18 on page 100.
72(X'48')	SMF119SS_IPSec_EncAlg	2	Binary	The tunnel encryption algorithms. Same values as SMF119SS_TLS_CS_Enc_Alg in Table 18 on page 100.

The zERT summary Distinguished Names (DN) section contains one or more variable length X.500 DNs from relevant X.509 certificates. Subject and issuer DNs from the certificates are included in the zERT DNs section.

If any DNs exist, there is one zERT summary DN section that contains all the DNs. For each DN included in the section, there is a 2-byte length field, a 2-byte DN type field, and a variable length DN. The following structure is used to describe the fields present for each DN.

Table 21 on page 116 illustrates the format of the data structure for each DN in a zERT summary record DNs section.

Table 21. Data structure for each DN included in a zERT summary record Distinguished Name section				
Offset	**Name**	**Length**	**Format**	**Description**
0(X'0')	SMF119SS_DN_Len	2	Binary	Length of the DN structure (includes the length of SMF119SS_DN_Len, SMF119SS_DN_Type, and SMF119SS_DN)

Table 21. Data structure for each DN included in a zERT summary record Distinguished Name section (continued)				
Offset	**Name**	**Length**	**Format**	**Description**
2(X'2')	SMF119SS_DN_Type	2	Binary	Type of Distinguished Name: • X'0001': IPSec Local Certificate Subject DN • X'0002': IPSec Local Certificate Issuer DN • X'0003': IPSec Remote Certificate Subject DN • X'0004': IPSec Remote Certificate Issuer DN • X'0005': TLS Server Certificate Subject DN • X'0006': TLS Server Certificate Issuer DN • X'0007': TLS Client Certificate Subject DN • X'0008': TLS Client Certificate Issuer DN • X'0009': SSH Server Certificate Subject DN • X'000A': SSH Server Certificate Issuer DN • X'000B': SSH Client Certificate Subject DN • X'000C': SSH Client Certificate Issuer DN
4(X'4')	SMF119SS_DN	1 to 1024	EBCDIC	The variable length DN value.

TCP/IP profile record Global configuration section

This section provides Global configuration information from the GLOBALCONFIG profile statement. There is only one of these sections in the record.

Table 22 on page 117 shows the TCP/IP profile record Global configuration section.

Table 22. TCP/IP profile record Global configuration section				
Offset	**Name**	**Length**	**Format**	**Description**
0(X'0')	NMTP_GBCFEye	4	EBCDIC	GBCF eyecatcher

Table 22. TCP/IP profile record Global configuration section (continued)

Offset	Name	Length	Format	Description
4(X'4')	NMTP_GBCFFlags	2	Binary	Flags: **X'8000', NMTP_GBCFExpBindPortRange:** If set, fields NMTP_GBCFExpBindPortRangeBegNum and NMTP_GBCFExpBindPortRangeEndNum contain the beginning and ending port numbers of the range of reserved TCP ports in the sysplex. **X'4000', NMTP_GBCFIqdMultiWrite:** If set, multiple write support is enabled for HiperSockets interfaces. **X'2000', NMTP_GBCFMlsCheckTerminate:** If set, the stack terminates if multi-level secure configuration inconsistencies are encountered. **X'1000', NMTP_GBCFSegOffload:** If set, TCP segmentation is offloaded to an OSA-Express feature. **Guideline :** This flag is deprecated. Use NMTP_V4CFSegOffload instead. **X'0800', NMTP_GBCFTcpipStats:** If set, several counters are written to the CFGPRINT DD data set when the TCP/IP stack terminates. **X'0400', NMTP_GBCFZiip:** If set, field NMTP_GBCFZiipOptions indicates for which workloads CPU cycles are displaced to a zIIP. **X'0200', NMTP_GBCFWlmPriorityQ:** If set, the following fields indicate the OSA-Express QDIO priority values that are assigned for packets associated with WLM service classes and for forwarded packets according to the control values for the WLMPRIORITYQ parameter: • NMTP_GBCFWPQCV0Pri • NMTP_GBCFWPQCV1Pri • NMTP_GBCFWPQCV2Pri • NMTP_GBCFWPQCV3Pri • NMTP_GBCFWPQCV4Pri • NMTP_GBCFWPQCV5Pri • NMTP_GBCFWPQCV6Pri • NMTP_GBCFWPQFwdPri **X'0100', NMTP_GBCFSMCR:** If set, this stack is enabled for SMC-R communications. **X'0080', NMTP_GBCFSMCD:** If set, this stack is enabled for SMC-D communications. **X'0040', NMTP_GBCFZERT:** If set, this stack is enabled for the zERT discovery function. The NMTP_GBCFZertParms field identifies additional ZERT subparameter settings.

Table 22. TCP/IP profile record Global configuration section (continued)

Offset	Name	Length	Format	Description
6(X'6')	NMTP_GBCFSysMonOptions	2	Binary	The following are sysplex monitor subparameter settings: **X'8000', NMTP_GBCFSysMonAutoRejoin:** If set, the stack automatically rejoins the sysplex group after problems that caused it to leave the sysplex group are resolved. **X'4000', NMTP_GBCFSysMonDelayJoin:** If set, the stack delays joining the sysplex group until OMPROUTE is active. **X'2000', NMTP_GBCFSysMonDynRoute:** If set, the TCP/IP stack monitors the presence of dynamic routes over those network interfaces for which the MONSYSPLEX parameter was specified. This setting is dynamically changed if the MONINTERFACE or NOMONINTERFACE subparameters are specified. **X'1000', NMTP_GBCFSysMonMonIntf:** If set, the TCP/IP stack monitors the status of network interfaces for which the MONSYSPLEX parameter was specified. **X'0800', NMTP_GBCFSysMonRecovery:** If set, the TCP/IP stack issues error messagess, leaves the sysplex group, and deletes all DVIPA interfaces when a sysplex problem is detected. **X'0400', NMTP_GBCFSysMonNoJoin:** If set, the TCP/IP stack does not join the sysplex group until the V TCPIP,,SYSPLEX,JOINGROUP command is issued. **X'0200', NMTP_GBCFSysMonDelayJoinI:** If set, the TCP/IP stack delays joining the sysplex group until the IPsec infrastructure is active and operational. **X'0100', NMTP_GBCFSysMonIpsec:** If set, the TCP/IP stack monitors the status of the IPsec infrastructure.
8(X'8')	NMTP_GBCFIqdVlanId	2	Binary	VLAN ID for the dynamic XCF HiperSockets interface. If not specified the value is 0.
10(X'A')	NMTP_GBCFSysWlmPoll	1	Binary	The number of seconds used by the sysplex distributor and its target servers, when polling WLM for new weight values.
11(X'B')	NMTP_GBCFZiipOptions	1	Binary	Workloads whose CPU cycles should be displaced to a zIIP. This field is valid only if the NMTP_GBCFZiip flag is set. The following are valid values: **X'80', NMTP_GBCFZiipIPSecurity:** If set, CPU cycles for IPSec workloads are displaced to a zIIP, when possible. **X'40', NMTP_GBCFZiipIqdioMultiWrite:** If set, CPU cycles for large TCP outbound messages are displaced to a zIIP.
12(X'C')	NMTP_GBCFSysMonTimerSecs	2	Binary	The number of seconds used by the sysplex monitor function to react to problems with needed sysplex resources.
14(X'E')	NMTP_GBCFXcfGroupId	2	EBCDIC	The 2-digit suffix used to generate the sysplex group name that the TCP/IP stack joins. If not specified the value is zero.
16(X'10')	NMTP_GBCFExpBindPortRangeBegNum	2	Binary	If flag NMTP_GBCFExpBindPortRange is set, this field contains the beginning port number in the reserved range.

Table 22. TCP/IP profile record Global configuration section (continued)

Offset	Name	Length	Format	Description
18(X'12')	NMTP_GBCFExpBindPortRangeEndNum	2	Binary	If flag NMTP_GBCFExpBindPortRange is set, this field contains the ending port number in the reserved range.
20(X'14')	NMTP_GBCFMaxRecs	4	Binary	Configured maximum records value for the D TCPIP,,NETSTAT command. The value range is 1 - 65535. The value 65536 indicates that the * (asterisk) value was specified. This means all records.
24(X'18')	NMTP_GBCFEcsaLimit	4	Binary	The maximum ECSA storage size in bytes that can be used by the TCP/IP stack.
28(X'1C')	NMTP_GBCFPoolLimit	4	Binary	The maximum private storage size in bytes that can be used in the TCP/IP address space.
32(X'20')	NMTP_GBCFWPQCV0Pri	1	Binary	The OSA-Express QDIO priority value that is assigned to packets represented by control value 0. This field is valid only if flag NMTP_GBCFWlmPriorityQ is set.
33(X'21')	NMTP_GBCFWPQCV1Pri	1	Binary	The OSA-Express QDIO priority value that is assigned to packets represented by control value 1. This field is valid only if flag NMTP_GBCFWlmPriorityQ is set.
34(X'22')	NMTP_GBCFWPQCV2Pri	1	Binary	The OSA-Express QDIO priority value that is assigned to packets represented by control value 2. This field is valid only if flag NMTP_GBCFWlmPriorityQ is set.
35(X'23')	NMTP_GBCFWPQCV3Pri	1	Binary	The OSA-Express QDIO priority value that is assigned to packets represented by control value 3. This field is valid only if flag NMTP_GBCFWlmPriorityQ is set.
36(X'24')	NMTP_GBCFWPQCV4Pri	1	Binary	The OSA-Express QDIO priority value that is assigned to packets represented by control value 4. This field is valid only if flag NMTP_GBCFWlmPriorityQ is set.
37(X'25')	NMTP_GBCFWPQCV5Pri	1	Binary	The OSA-Express QDIO priority value that is assigned to packets represented by control value 5. This field is valid only if flag NMTP_GBCFWlmPriorityQ is set.
38(X'26')	NMTP_GBCFWPQCV6Pri	1	Binary	The OSA-Express QDIO priority value that is assigned to packets represented by control value 6. This field is valid only if flag NMTP_GBCFWlmPriorityQ is set.
39(X'27')	NMTP_GBCFWPQFwdPri	1	Binary	The OSA-Express QDIO priority value that is assigned to forwarded packets. This field is valid only if flag NMTP_GBCFWlmPriorityQ is set.
40(X'28')	NMTP_GBCFAutoIQDX	1	Binary	AutoIQDX settings. If no flag bits are set, the NOAUTOIQDX parameter value is in effect. **X'02', NMTP_GBCFAutoIQDX_NoLargeData:** If this flag bit is set, dynamic IQDX interfaces are used for all eligible traffic, except for TCP data traffic that is sent with socket transmissions of 32 K or larger. **X'01', NMTP_GBCFAutoIQDX_AllTraffic:** If this flag bit is set, dynamic IQDX interfaces are used for all eligible traffic to the intraensemble data network.
41(X'29')	NMTP_GBCFPFidCnt	1	Binary	SMCR PFID count - the current number of configured PFID, port, and MTU entries in the NMTP_GBCFPFs array.

Table 22. TCP/IP profile record Global configuration section (continued)

Offset	Name	Length	Format	Description
42(X'2A')	NMTP_GBCFSMCGFlags	1	Binary	SMCGlobal flags: **x'80', NMTP_GBCFAUTOCACHE** AUTOCACHE is configured. This function is active only when flag NMTP_GBCFSMCR is set and field NMTP_GBCFPFidCnt is not zero, or flag NMTP_GBCFSMCD is set. **x'40', NMTP_GBCFAUTOSMC** AUTOSMC is configured.
43(x'2B')	NMTP_GBCFAdjDVMSS	1	Binary	ADJUSTDVIPAMSS settings. **x'80', NMTP_GBCFAdjDVMSS_AUTO** If this flag is set, TCP/IP automatically adjusts the MSS size to avoid fragmentation for TCP connections that use VIPAROUTE and distributed DVIPAs. **x'40', NMTP_GBCFAdjDVMSS_ALL** If this flag is set, TCP/IP automatically adjusts the MSS size to avoid fragmentation for TCP connections that use any DVIPA, distributed or not, as the source IP address. **x'20', NMTP_GBCFAdjDVMSS_NONE** If this flag is set, TCP/IP does not adjust the MSS for any TCP connections.
44(X'2C')	NMTP_GBCFFixedMemory	4	Binary	SMCR FIXEDMEMORY value in megabytes
48(X'30')	NMTP_GBCFTcpKeepMinInt	4	Binary	SMCR TCPKEEPMININTERVAL value in seconds
52(X'34')	NMTP_GBCFPFs(16)	96	Binary	SMCR PFID array that contains up to 16 entries. Each entry contains the following information: • PFID (2-byte hexadecimal value) • PortNum • MTU value **Note :** When PFID represents a RoCE Express2 feature, the PortNum value is the port number configured for the PFID in the Hardware Configuration Definition (HCD). This port number is learned by VTAM and TCP/IP during activation of the PFID and might be different from the value coded for PORTNUM for this PFID on the GLOBALCONFIG SMCR statement.
148(X'94')	NMTP_GBCFZertParms	1	Binary	Additional ZERT subparameters. **X'80' NMTP_GBCFZERTAGG:** If set, this stack is enabled for the zERT aggregation function. **X'40' NMTP_GBCFZERTINTV:** If set, a zERT Aggregation recording interval separate from the SMF interval is specified. **X'20' NMTP_GBCFZERTSYNC:** If set, the reference time for the beginning of the INVAL interval is specified. The reference time is in NMTP_GBCFzAGGtim_SYNCVAL_HH and NMTP_GBCFzAGGtim_SYNCVAL_MM.

Table 22. TCP/IP profile record Global configuration section (continued)

Offset	Name	Length	Format	Description
149(X'95')	NMTP_GBCFAutoIQDC	1	Binary	AutoIQDC settings. If no flag bits are set, the NOAUTOIQDC parameter value is in effect. **x'02', NMTP_GBCFAutoIQDC_NoLargeData** If this flag bit is set, dynamic IQDC interfaces are used for all eligible traffic, except for TCP data traffic that is sent with socket transmissions of 32 K or larger. **x'01', NMTP_GBCFAutoIQDC_AllTraffic** If this flag bit is set, dynamic IQDC interfaces are used for all eligible traffic.
150(X'96')	NMTP_GBCFResv1	2	Binary	Reserved
152(X'98')	NMTP_GBCFFixedMemoryD	4	Binary	SMCD FIXEDMEMORY value in megabytes
156(X'9C')	NMTP_GBCFTcpKeepMinIntD	4	Binary	SMCD TCPKEEPMININTERVAL value in seconds
160(X'A0')	NMTP_GBCFzAGGtim_INTVAL	1	Binary	zERT Aggregation recording interval (INTVAL) in hours. This value is set if the NMTP_GBCFZERTINTV flag is on. Otherwise, the field will be 0.
161(X'A1')	NMTP_GBCFzAGGtim_SYNCVAL_HH	1	Binary	Hours portion of the SYNCVAL value. SYNCVAL is the reference time for which the zERT Aggregation recording interval (INTVAL) is applied. This value is set if the NMTP_GBCFZERTSYNC flag is on. Otherwise, the field will be 0.
162(X'A2')	NMTP_GBCFzAGGtim_SYNCVAL_MM	1	Binary	Minutes portion of the SYNCVAL value. SYNCVAL is the reference time for which the zERT Aggregation recording interval (INTVAL) is applied. This value is set if the NMTP_GBCFZERTSYNC flag is on. Otherwise, the field will be 0.
163(X'A3')	NMTP_GBCFSMCEIDCount	1	Binary	Count of user configured EIDs.
164(X'A4')	NMTP_GBCFUEIDList	128	EBCDIC	List of user configured EIDs.
292(X'124')	NMTP_GBCFSYSTEMEIDSTR	32	EBCDIC	System generated EID.
324(X'144')	rsvd	12	Binary	Reserved

Chapter 6. IP Messages: Volume 4 (EZZ, SNM)

EZZ8453I ***jobtype* STORAGE**

Explanation

TCP/IP issues this message as part of a group of messages in response to a DISPLAY TCPIP,*procname*,STOR command. This is the first message in the group. A complete description of the message group follows:

```
EZZ8453I jobtype STORAGE

EZZ8454I jobname STORAGE     CURRENT MAXIMUM  LIMIT
EZD2018I location
EZZ8455I         storagetype current maximum  limit
EZD2024I         type        current maximum
EZZ8459I DISPLAY TCPIP STOR COMPLETED SUCCESSFULLY
```

EZZ8453I

This message identifies the type of information shown in the message group.

jobtype is the type of job. Possible values are:

TCPIP
The job is a TCP/IP job.

TELNET
The job is a TN3270 job.

EZZ8454I

This message is a header message for EZZ8455I.
jobname is the job name associated with the procedure used to start the job.

EZD2018I

This message identifies the storage location for the storage described in the subsequent message EZZ8455I.

location is the location of the storage. Possible values are:

31-BIT
The storage is 31-bit storage located below the 2 GB bar.

64-BIT
The storage is 64-bit storage located above the 2 GB bar.

EZZ8455I

This message contains storage totals.

storagetype is the storage type. Possible values are:

ECSA
The amount of extended common storage area in use.

PRIVATE
The amount of pooled private storage in use.

ECSA MODULES
The amount of common storage in use for load modules loaded by dynamic LPA.

HVCOMMON
The amount of 64-bit common storage in use.

HVPRIVATE
The amount of 64-bit private storage in use.

TRACE HVCOMMON
The amount of 64 bit common storage that was obtained for tracing.

TRACE HVPRIVATE
The amount of 64 bit common storage that was obtained for tracing.

ZERTAGG HVPRIVATE
The amount of 64 bit private storage in use for ZERT Aggregation records. An instance of message specifying ZERTAGG HVPRIVATE is only included in the message group if ZERT Aggregation is enabled for this TCP/IP stack by specifying the ZERT AGGREGATION parameter on the GLOBALCONFIG profile statement.

SMC-R FIXEDMEMORY
The amount of 64-bit private fixed storage in use for Shared Memory Communications over Remote Direct Memory Access (SMC-R). An instance of message EZZ8455I specifying SMC-R FIXEDMEMORY is only included in the message group if SMC-R is or was previously enabled for this TCP/IP stack by specifying the SMCR parameter on the GLOBALCONFIG profile statement.

SMC-D FIXEDMEMORY
The amount of 64-bit private fixed storage in use for Shared Memory Communications - Direct Memory Access (SMC-D). An instance of message EZZ8455I specifying SMC-D FIXEDMEMORY is included in the message group only if SMC-D is or was previously enabled for this TCP/IP stack by specifying the SMCD parameter on the GLOBALCONFIG profile statement.

current is the amount of storage currently allocated. The value ends with either K to indicate 1024 bytes or M to indicate 1048576 bytes. The *current* value for SMC-R FIXEDMEMORY is the sum of the SMC-R SEND MEMORY and SMC-R RECV MEMORY *current* values in message EZD2024I.

maximum is the maximum amount of storage ever allocated since the job was started. The value ends with either K to indicate 1024 bytes or M to indicate 1048576 bytes. The *maximum* value for SMC-R FIXEDMEMORY is the maximum amount of storage ever allocated for SMC-R send and receive buffers combined, but can be less than the sum of the *maximum* values in message EZD2024I for SMC-R SEND MEMORY and SMC-R RECV MEMORY.

limit is the storage limit that the job allows.

- When *jobtype* on EZZ8453I is TELNET, the storage does not have a limit.
- When *storagetype* is ZERTAGG HVPRIVATE, the storage does not have a limit.
- When *storagetype* is SMC-R FIXEDMEMORY, *limit* is defined using the SMCR FIXEDMemory keyword value on the GLOBALCONFIG profile statement. The FIXEDMemory value represents the limit for all SMC-R storage, regardless of whether it is used for send or receive buffers.
- When *storagetype* is SMC-D FIXEDMEMORY, *limit* is defined using the SMCD FIXEDMemory keyword value on the GLOBALCONFIG profile statement.
- Otherwise, *limit* is defined on the GLOBALCONFIG profile statement for TCP/IP.

The value ends with either K to indicate 1024 bytes or M to indicate 1048576 bytes, is NOLIMIT if the storage does not have a limit, or is N/A for SMC-R FIXEDMEMORY when the SMC-R function was previously enabled on this TCP/IP stack but is not currently enabled. See z/OS Communications Server: IP Configuration Reference for more information.

EZD2024I

- This message contains storage totals. This message is only included in the message group if SMC-R is or was previously enabled for this TCP/IP stack by specifying the SMCR parameter on the GLOBALCONFIG profile statement.
- *type* is the storage type. Possible values are:

  SMC-R RECV MEMORY
  The amount of 64-bit private storage allocated as SMC-R receive buffers for all SMC-R link groups associated with this TCP stack.

  SMC-R SEND MEMORY
  The amount of 64-bit private storage allocated for SMC-R send buffers by this TCP/IP stack.

- *current* is the amount of storage currently allocated. The value ends with either K to indicate 1024 bytes or M to indicate 1048576 bytes.
- *maximum* is the maximum amount of storage ever allocated since the job was started. The value ends with either K to indicate 1024 bytes or M to indicate 1048576 bytes.

EZZ8459I

This message is displayed when the DISPLAY TCPIP,procname,STOR command completed.

System action

The job continues.

Operator response

None.

System programmer response

None.

User response

Not applicable.

Problem determination

None.

Module

EZACDDSU

Routing code

0

Descriptor code

5, 8, 9

Automation

Not applicable.

Example

```
EZZ8453I TCPIP STORAGE
EZZ8454I TCPCS    STORAGE            CURRENT   MAXIMUM     LIMIT
EZD2018I 31-BIT
EZZ8455I          ECSA                 2701K     3156K   NOLIMIT
EZZ8455I          PRIVATE              8557K     8561K   NOLIMIT
EZZ8455I          ECSA MODULES         8639K     8639K   NOLIMIT
EZD2018I 64-BIT
EZZ8455I          HVCOMMON                1M        1M   NOLIMIT
EZZ8455I          HVPRIVATE              50M       50M   NOLIMIT
EZZ8455I          TRACE HVCOMMON       2048M     2048M   NOLIMIT
EZZ8455I          SMC-R FIXEDMEMORY      12M       16M       40M
EZD2024I            SMC-R SEND MEMORY     4M        4M
EZD2024I            SMC-R RECV MEMORY     8M       12M
EZZ8455I          SMC-D FIXEDMEMORY      12M       16M       40M
EZZ8459I DISPLAY TCPIP STOR COMPLETED SUCCESSFULLY
```

Chapter 7. z/OS Summary of Message and Interface Changes

PROFILE.TCPIP statement and parameter changes

Table 23 on page 127 lists the new and updated Communications Server PROFILE.TCPIP configuration statements and parameters. See z/OS Communications Server: IP Configuration Reference for more detailed information.

Table 23. New and changed Communications Server PROFILE.TCPIP configuration statements and parameters for z/OS V2R3

Statement	Description	Reason for change
DEVICE	The MPCOSA DEVICE profile statement and its corresponding LINK profile statements are no longer supported.	Removal of support for legacy devices
GLOBALCONFIG	The AUTOIQDC parameter is defined to enable and configure the HiperSockets Converged Interface function. The AUTOIQDC parameter includes the ALLTRAFFIC and NOLARGEDATA sub-parameters. The NOAUTOIQDC parameter is defined to disable the HiperSockets Converged Interface function.	HiperSockets Converged Interface support
GLOBALCONFIG	ZERT AGGREGATION INTVAL SYNCVAL • INTVAL is the recording interval that would permit a minimum of 1 hour to a maximum of 24 hours (1 day). The default setting is SMF and this indicates the zERT Aggregation interval is determined by the SMF interval. • SYNCVAL indicates a reference time for which zERT Aggregation records will begin to record. It is in the 24 hour clock format hh:mm (hour and minute value separated by a colon) and the default value is midnight or 00:00.	z/OS Encryption Readiness Technology (zERT) aggregation recording interval
GLOBALCONFIG	New subparameters AGGREGATION and NOAGGREGATION are defined on the GLOBALCONFIG ZERT parameter	z/OS Encryption Readiness Technology (zERT) aggregation

Table 23. New and changed Communications Server PROFILE.TCPIP configuration statements and parameters for z/OS V2R3 (continued)

Statement	Description	Reason for change
GLOBALCONFIG	The ZERT parameter is defined to enable the z/OS Encryption Readiness Technology (zERT) function. The NOZERT parameter is defined to disable the zERT function.	z/OS Encryption Readiness Technology (zERT)
	The SMCD parameter is defined to enable and configure the Shared Memory Communications - Direct Memory Access (SMC-D) function. The SMCD parameter includes the FIXEDMEMORY and TCPKEEPMININTERVAL subparameters. The NOSMCD parameter is defined to disable SMC-D function	Shared Memory Communications - Direct Memory Access
	Added SMCGLOBAL parameter to provide global settings for the Shared Memory Communications over Remote Direct Memory Access (SMC-R) function and Shared Memory Communications - Direct Memory Access (SMC-D) function. The following subparameters can be specified: • AUTOCACHE and NOAUTCACHE Control caching of unsuccessful attempts to use SMC-R or SMC-D. • AUTOSMC and NOAUTOSMC Control monitoring incoming TCP connections to determine whether they would benefit from SMC-R or SMC-D.	• Shared Memory Communications over RDMA enhancements • Shared Memory Communications - Direct Memory Access
INTERFACE	The SMCD parameter is defined to enable the SMC-D function for the following statements: • IPAQENET, when CHPIDTYPE OSD is specified • IPAQENET6, when CHPIDTYPE OSD is specified • IPAQIDIO • IPAQIDIO6 The NOSMCD parameter is defined to disable the SMC-D function.	Shared Memory Communications - Direct Memory Access
IPCONFIG	The SMCD subparameter is defined on this statement for the DYNAMICXCF parameter to enable the SMC-D function. The NOSMCD subparameter is defined to disable SMC-D function.	Shared Memory Communications - Direct Memory Access
IPCONFIG6	The SMCD subparameter is defined on this statement for the DYNAMICXCF parameter to enable the SMC-D function. The NOSMCD subparameter is defined to disable SMC-D function.	Shared Memory Communications - Direct Memory Access

Table 23. New and changed Communications Server PROFILE.TCPIP configuration statements and parameters for z/OS V2R3 (continued)

Statement	Description	Reason for change
LINK	The following LINK profile statements are no longer supported: • FDDI and IBMTR • IPAQTR	Removal of support for legacy devices
NETMONITOR	New subparameters ZERTSUMMARY and NOZERTSUMMARY are added to control the real-time zERT Summary SMF NMI service (SYSTCPES).	z/OS Encryption Readiness Technology (zERT) aggregation
NETMONITOR	New ZERTSERVICE and NOZERTSERVICE parameters are added to control the real-time zERT NMI service (SYSTCPER).	z/OS Encryption Readiness Technology (zERT)
PORT	• Support added for specifying an asterisk in any position of the jobname parameter to indicate zero or more unspecified characters. • Support added for specifying a question mark in any position of the jobname parameter to indicate a single unspecified character.	Enhanced wildcard support for jobnames on PORT and PORTRANGE statements
	• The SMC parameter is enhanced to enable SMC-D function for the specified port. • The NOSMC parameter is enhanced to disable SMC-D function for the specified port.	Shared Memory Communications - Direct Memory Access
PORTRANGE	• Support added for specifying an asterisk in any position of the jobname parameter to indicate zero or more unspecified characters. • Support added for specifying a question mark in any position of the jobname parameter to indicate a single unspecified character.	Enhanced wildcard support for jobnames on PORT and PORTRANGE statements
	• The SMC parameter is enhanced to enable the SMC-D function. • The NOSMC parameter is enhanced to disable the SMC-D function.	Shared Memory Communications - Direct Memory Access
SMFCONFIG	New subparameters ZERTSUMMARY and NOZERTSUMMARY are defined as TYPE119 values.	z/OS Encryption Readiness Technology (zERT) aggregation

Table 23. New and changed Communications Server PROFILE.TCPIP configuration statements and parameters for z/OS V2R3 (continued)		
Statement	**Description**	**Reason for change**
SMFCONFIG	New ZERTDETAIL and NOZERTDETAIL parameters are added to control creation of zERT-related SMF 119 subtype 11 records.	z/OS Encryption Readiness Technology (zERT)
	• New SMCDLINKSTATISTICS and NOSMCDLINKSTATISTICS parameters are updated to control the creation of SMF 119 subtype 38 interval records for SMC-D link statistics. • New SMCDLINKEVENT and NOSMCDLINKEVENT parameters are added to create SMF 119 subtype 39 and 40 event records for SMC-D link state start and end events. • IFSTATISTICS and NOIFSTATISTICS parameters are updated to control the creation of the SMF 119 subtype 45 interval records for ISM interface statistics.	Shared Memory Communications - Direct Memory Access
TRANSLATE	The FDDI and IBMTR parameters are no longer supported.	Removal of support for legacy devices

Netstat operator commands (DISPLAY TCPIP,,NETSTAT)

Table 24 on page 130 lists the new and updated Communications Server IP Netstat operator command DISPLAY TCPIP,,NETSTAT. See Table 1 for the other Communications Server IP operator command entries.

See z/OS Communications Server: IP System Administrator's Commands for more detailed information about the Communications Server IP operator commands.

All parameters in the following table are for the DISPLAY TCPIP,,NETSTAT operator command.

Table 24. New and changed Communications Server Netstat operator commands (DISPLAY TCPIP,,NETSTAT) for z/OS V2R3		
Parameters	**Description**	**Reason for change**
ALL	• Displays Shared Memory Communications - Direct Memory Access (SMC-D) information for TCP connections. • Accepts the SMCID filter to display the TCP connections that are associated with a specific local SMC-D link identifier.	Shared Memory Communications - Direct Memory Access
ALLCONN	Accepts the SMCID filter to display the TCP connections that are associated with a specific local SMC-D link identifier.	Shared Memory Communications - Direct Memory Access
ARP	Displays ARP information for the HiperSockets Converged Interfaces.	HiperSockets Converged Interface support

Table 24. New and changed Communications Server Netstat operator commands (DISPLAY TCPIP,,NETSTAT) for z/OS V2R3 (continued)

Parameters	Description	Reason for change
CONFIG	Displays the setting of the AUTOIQDC parameter.	HiperSockets Converged Interface support
CONFIG	Displays new ZERT Aggregation sub parameter information with INTVAL and SYNCVAL in the GLOBALCONFIG section.	z/OS Encryption Readiness Technology (zERT) aggregation recording interval
CONFIG	• Displays new ZERTSUMMARY subparameter information in the SMFCONFIG section. • Displays new ZERT Aggregation subparameter information in the GLOBALCONFIG section. • Displays new ZERTSUMMARY subparameter information in the NETMONITOR section.	z/OS Encryption Readiness Technology (zERT) aggregation
CONFIG	• New SMF Parameters Type119 field ZertDetail • New Global Configuration Information field ZERT • New Network Monitor Configuration Information field ZertSrv	z/OS Encryption Readiness Technology (zERT)
	• Displays new SMCD parameter information in the GLOBALCONFIG section. • Displays new DYNAMICXCF SMCD subparameter information in the IPCONFIG and IPCONFIG6 section. • Displays new SMCDLINKSTATISTICS and SMCDLINKEVENT subparameter information in the SMFCONFIG section.	• Shared Memory Communications - Direct Memory Access
CONFIG	In the Global Configuration section, the SMCR PORTNUM represents the configured or learned port number used for the PFID.	Communications Server support for RoCE Express2 features
CONN	Accepts the SMCID filter to display the TCP connections that are associated with a specific local SMC-D link identifier.	Shared Memory Communications - Direct Memory Access

Table 24. New and changed Communications Server Netstat operator commands (DISPLAY TCPIP,,NETSTAT) for z/OS V2R3 (continued)

Parameters	Description	Reason for change
DEvlinks	• Displays SMC-D information for OSD and HiperSockets interfaces. • Accepts the SMCID filter to display devices that are associated with a specific SMC-D local link identifier. • Accepts the SMC modifier to display detailed SMC-D information about active internal shared memory (ISM) interfaces and their associated SMC-D links. • Accepts the new PNETID modifier to display information about interfaces with a PNETID value, or information about interfaces with a specific PNETID value.	• Shared Memory Communications - Direct Memory Access
DEvlinks	The card generation level and speed information are displayed for RNIC interfaces representing "RoCE Express" features.	Communications Server support for RoCE Express2 features
DEvlinks	Displays the name of the HiperSockets Converged Interface, if any, that is associated with an OSD and statistics related to that associated interface.	HiperSockets Converged Interface support
ND	Displays ND information for the HiperSockets Converged Interfaces.	HiperSockets Converged Interface support
PORTLIST	Displays a new flag, M, to indicate whether the port or port range is explicitly enabled for SMC-R and SMC-D.	• Shared Memory Communications over RDMA enhancements • Shared Memory Communications - Direct Memory Access
	Flag N is enhanced to indicate whether the port or the port range is explicitly disabled for SMC-R and SMC-D.	Shared Memory Communications - Direct Memory Access
STATS	Displays a new SMCD statistics section. The SMC-D statistics are displayed when no PROTOCOL modifier is specified, or when PROTOCOL=TCP is specified as the modifier value.	Shared Memory Communications - Direct Memory Access

Table 24. New and changed Communications Server Netstat operator commands (DISPLAY TCPIP,,NETSTAT) for z/OS V2R3 (continued)		
Parameters	**Description**	**Reason for change**
TTLS	• The report output can have a new value of Level1, Level2, or Level3 for the FIPS140 parameter. • The report output can have a new value of 128Min or 192Min for the SuiteBProfile parameter. • New field ServerCertificateLabel • New field 3DesKeyCheck • New field ClientEDHGroupSize • New field ServerEDHGroupSize • New field PeerMinCertVersion • New field PeerMinDHKeySize • New field PeerMinDsaKeySize • New field PeerMinECCKeySize • New field PeerMinRsaKeySize • New field OcspResponseSigAlgPairs • New field OcspServerStapling • New field ServerScsv	AT-TLS currency with System SSL

NETSTAT TSO commands

Table 25 on page 133 lists the new and updated Communications Server NETSTAT TSO command.

See z/OS Communications Server: IP System Administrator's Commands for more detailed information about the Communications Server TSO commands.

Table 25. New and changed Communications Server NETSTAT TSO commands for z/OS V2R3		
Parameter	**Description**	**Reason for change**
ALL	• Displays Shared Memory Communications - Direct Memory Access (SMC-D) information for TCP connections. • Accepts the SMCID filter to display the TCP connections that are associated with a specific local SMC-D link identifier.	Shared Memory Communications - Direct Memory Access
ALLCONN	Accepts the SMCID filter to display the TCP connections that are associated with a specific local SMC-D link identifier.	Shared Memory Communications - Direct Memory Access
ARP	Displays ARP information for the HiperSockets Converged Interfaces.	HiperSockets Converged Interface support
CONFIG	Displays the setting of the AUTOIQDC parameter.	HiperSockets Converged Interface support

Table 25. New and changed Communications Server NETSTAT TSO commands for z/OS V2R3 (continued)		
Parameter	**Description**	**Reason for change**
CONFIG	Displays new ZERT Aggregation subparameter information with INTVAL and SYNCVAL in the GLOBALCONFIG section.	z/OS Encryption Readiness Technology (zERT) aggregation
CONFIG	• Displays new ZERTSUMMARY subparameter information in the SMFCONFIG section. • Displays new ZERT aggregation subparameter information in the GLOBALCONFIG section. • Displays new ZERTSUMMARY subparameter information in the NETMONITOR section.	z/OS Encryption Readiness Technology (zERT) aggregation
CONFIG	• New SMF Parameters Type119 field ZertDetail • New Global Configuration Information field ZERT • New Network Monitor Configuration Information field ZertSrv	z/OS Encryption Readiness Technology (zERT)
	• Displays new SMCD parameter information in the GLOBALCONFIG section. • Displays new DYNAMICXCF SMCD subparameter information in the IPCONFIG and IPCONFIG6 section. • Displays new SMCDLINKSTATISTICS and SMCDLINKEVENT subparameter information in the SMFCONFIG section.	• Shared Memory Communications - Direct Memory Access
CONFIG	In the Global Configuration section, the SMCR PORTNUM represents the configured or learned port number used for the PFID.	Communications Server support for RoCE Express2 features
CONN	Accepts the SMCID filter to display the TCP connections that are associated with a specific local SMC-D link identifier.	Shared Memory Communications - Direct Memory Access
DEvlinks	• Displays SMC-D information for OSD and HiperSockets interfaces. • Accepts the SMCID filter to display devices that are associated with a specific SMC-D local link identifier. • Accepts the SMC modifier to display detailed SMC-D information about active internal shared memory (ISM) interfaces and their associated SMC-D links. • Accepts the new PNETID modifier to display information about interfaces with a PNETID value, or information about interfaces with a specific PNETID value.	• Shared Memory Communications - Direct Memory Access
DEvlinks	The card generation level and speed information are displayed for RNIC interfaces representing "RoCE Express" features.	Communications Server support for RoCE Express2 features

Table 25. New and changed Communications Server NETSTAT TSO commands for z/OS V2R3 (continued)		
Parameter	**Description**	**Reason for change**
DEvlinks	Displays the name of the HiperSockets Converged Interface, if any, that is associated with an OSD and statistics related to that associated interface.	HiperSockets Converged Interface support
ND	Displays ND information for the HiperSockets Converged Interfaces.	HiperSockets Converged Interface support
PORTLIST	Displays a new flag, M, to indicate whether the port or port range is explicitly enabled for SMC-R and SMC-D.	• Shared Memory Communications over RDMA enhancements • Shared Memory Communications - Direct Memory Access
	Flag N is enhanced to indicate whether the port or the port range is explicitly disabled for SMC-R and SMC-D.	Shared Memory Communications - Direct Memory Access
STATS	Displays a new SMCD statistics section. The SMC-D statistics are displayed when no PROTOCOL modifier is specified, or when PROTOCOL=TCP is specified as the modifier value.	Shared Memory Communications - Direct Memory Access
TTLS	• The report output can have a new value of Level1, Level2, or Level3 for the FIPS140 parameter. • The report output can have a new value of 128Min or 192Min for the SuiteBProfile parameter. • New field ServerCertificateLabel • New field 3DesKeyCheck • New field ClientEDHGroupSize • New field ServerEDHGroupSize • New field PeerMinCertVersion • New field PeerMinDHKeySize • New field PeerMinDsaKeySize • New field PeerMinECCKeySize • New field PeerMinRsaKeySize • New field OcspResponseSigAlgPairs • New field OcspServerStapling • New field ServerScsv	AT-TLS currency with System SSL

Netstat UNIX commands

Table 26 on page 136 lists the new and updated Communications Server z/OS UNIX netstat command. See General updates of z/OS UNIX commands for the other (the non-netstat) z/OS UNIX command entries.

See z/OS Communications Server: IP System Administrator's Commands for more detailed information about the z/OS UNIX commands.

All parameters in the following table are for the z/OS UNIX netstat command.

Table 26. New and changed Communications Server z/OS UNIX netstat commands for z/OS V2R3

Parameter	Description	Reason for change
-A	• Displays Shared Memory Communications - Direct Memory Access (SMC-D) information for TCP connections. • Accepts the SMCID filter to display the TCP connections that are associated with a specific local SMC-D link identifier.	Shared Memory Communications - Direct Memory Access
-a	Accepts the SMCID filter to display the TCP connections that are associated with a specific local SMC-D link identifier.	Shared Memory Communications - Direct Memory Access
-c	Accepts the SMCID filter to display the TCP connections that are associated with a specific local SMC-D link identifier.	Shared Memory Communications - Direct Memory Access
-d	The card generation level and speed information are displayed for RNIC interfaces representing "RoCE Express" features.	Communications Server support for RoCE Express2 features
-d	• Displays SMC-D information for OSD and HiperSockets interfaces. • Accepts the SMCID filter to display devices that are associated with a specific SMC-D local link identifier. • Accepts the SMC modifier to display detailed SMC-D information about active internal shared memory (ISM) interfaces and their associated SMC-D links. • Accepts the new PNETID modifier to display information about interfaces with a PNETID value, or information about interfaces with a specific PNETID value.	Shared Memory Communications - Direct Memory Access
-d	Displays the name of the HiperSockets Converged Interface, if any, that is associated with an OSD and statistics related to that associated interface.	HiperSockets Converged Interface support
-f	Displays the setting of the AUTOIQDC parameter.	HiperSockets Converged Interface support

Table 26. New and changed Communications Server z/OS UNIX netstat commands for z/OS V2R3 (continued)		
Parameter	**Description**	**Reason for change**
-f	Displays new ZERT Aggregation sub parameter information with INTVAL and SYNCVAL in the GLOBALCONFIG section.	z/OS Encryption Readiness Technology (zERT) aggregation recording interval
-f	• Displays new ZERTSUMMARY subparameter information in the SMFCONFIG section. • Displays new ZERT Aggregation subparameter information in the GLOBALCONFIG section. • Displays new ZERTSUMMARY subparameter information in the NETMONITOR section.	z/OS Encryption Readiness Technology (zERT) aggregation
-f	In the Global Configuration section, the SMCR PORTNUM represents the configured or learned port number used for the PFID.	Communications Server support for RoCE Express2 features
-f	• New SMF Parameters Type119 field ZertDetail • New Global Configuration Information field ZERT • New Network Monitor Configuration Information field ZertSrv	z/OS Encryption Readiness Technology (zERT)
	• Displays new SMCD parameter information in the GLOBALCONFIG section. • Displays new DYNAMICXCF SMCD subparameter information in the IPCONFIG and IPCONFIG6 section. • Displays new SMCDLINKSTATISTICS and SMCDLINKEVENT subparameter information in the SMFCONFIG section.	Shared Memory Communications - Direct Memory Access
-n	Displays ND information for the HiperSockets Converged Interfaces.	HiperSockets Converged Interface support
-o	Displays a new flag, M, to indicate whether the port or port range is explicitly enabled for SMC-R and SMC-D.	• Shared Memory Communications over RDMA enhancements • Shared Memory Communications - Direct Memory Access
	Flag N is enhanced to indicate whether the port or the port range is explicitly disabled for SMC-R and SMC-D.	Shared Memory Communications - Direct Memory Access
-R	Displays ARP information for the HiperSockets Converged Interfaces.	HiperSockets Converged Interface support

Table 26. New and changed Communications Server z/OS UNIX netstat commands for z/OS V2R3 (continued)		
Parameter	**Description**	**Reason for change**
-S	Displays a new SMCD statistics section. The SMC-D statistics are displayed when no PROTOCOL modifier is specified, or when PROTOCOL=TCP is specified as the modifier value.	Shared Memory Communications - Direct Memory Access
-x	• The report output can have a new value of LEVEL1, LEVEL2, or LEVEL3 for the FIPS140 parameter. • The report output can have a new value of 128Min or 192Min for the SuiteBProfile parameter. • New field ServerCertificateLabel • New field 3DesKeyCheck • New field ClientEDHGroupSize • New field ServerEDHGroupSize • New field PeerMinCertVersion • New field PeerMinDHKeySize • New field PeerMinDsaKeySize • New field PeerMinECCKeySize • New field PeerMinRsaKeySize • New field OcspResponseSigAlgPairs • New field OcspServerStapling • New field ServerScsv	AT-TLS currency with System SSL

TCP/IP callable NMI (EZBNMIFR)

Table 27 on page 139 lists the updates to the Communications Server TCP/IP callable NMI.

Table 27. New Communications Server TCP/IP callable NMI (EZBNMIFR) for z/OS V2R3

Request	Parameter/output	Description	Reason for change
GetConnectionDetail	• NWMConnFlag01 – NWMConnSMCDCfg • NWMConnSMCDStatus • NWMConnSMCDReason • NWMConnSMCFlags – NWMConnSMCDRsnRmt • NWMConnSMCFlags – NWMConnSMCDCached • NWMConnLclSMCLinkId • NWMConnRmtSMCLinkId	• New flag bit NWMConnSMCDCfg is set in the NWMConnFlag01 field to indicate whether the SMCD parameter is configured on the GLOBALCONFIG statement. • New NWMConnSMCDStatus field that indicates whether this connection is traversing an SMC-D link. • New NWMConnSMCDReason field that indicates why a connection is not using an SMC-D link. • New flag bit NWMConnSMCDRsnRmt is set in the NWMConnSMCFlags field to indicate whether the NWMConnSMCDReason is set by the remote peer. • New flag bit NWMConnSMCDCached is set in the NWMConnSMCFlags field to indicate whether this connection is cached to not use SMC-D. • Existing NWMConnLclSMCLinkId field that indicates the local stack link ID for the SMC-R or SMC-D link that this connection traverses. • Existing NWMConnRmtSMCLinkId field that indicates the remote stack link ID for the SMC-R or SMC-D link that this connection traverses.	Shared Memory Communications - Direct Memory Access
	NWMConnLclSMCBufSz	Existing NWMConnLclSMCBufSz field that indicates the size of the RMB or DMB element that the local host uses to receive data on this connection from the remote host.	• Shared Memory Communications over RDMA enhancements • Shared Memory Communications - Direct Memory Access
	NWMConnRmtSMCBufSz	Existing NWMConnRmtSMCBufSz field that indicates the size of the RMB or DMB element that the remote host uses to receive data on this connection from the local host.	• Shared Memory Communications over RDMA enhancements • Shared Memory Communications - Direct Memory Access
	NWMConnTTLSFIPS140Mode	Updated to support FIPS140 Level1, Level2, and Level3.	AT-TLS currency with System SSL

Table 27. New Communications Server TCP/IP callable NMI (EZBNMIFR) for z/OS V2R3 (continued)

Request	Parameter/output	Description	Reason for change
GetGlobalStats	NWMTCPSTSMCDCfg Existing TCP stats changed: • NWMTCPSTCurrEstab • NWMTCPSTActiveOpened • NWMTCPSTPassiveOpened • NWMTCPSTConnClosed • NWMTCPSTInSegs • NWMTCPSTOutSegs • NWMTCPSTOutRsts • NWMTCPSTEstabResets • NWMTCPSTAcceptCount • NWMTCPSTKeepAliveProbes • NWMTCPSTKeepAliveDrop • NWMTCPSTFinwait2Drops New SMC-D stats: • NWMTCPSTSMCDCurrEstabLnks • NWMTCPSTSMCDActLnkOpened • NWMTCPSTSMCDPasLnkOpened • NWMTCPSTSMCDLnksClosed • NWMTCPSTSMCDCurrEstab • NWMTCPSTSMCDActiveOpened • NWMTCPSTSMCDPassiveOpened • NWMTCPSTSMCDConnClosed • NWMTCPSTSMCRInSegs • NWMTCPSTSMCROutSegs • NWMTCPSTSMCRInRsts • NWMTCPSTSMCROutRsts	• New flag bit NWMTCPSTSMCDCfg is set in the NWMTCPSTFlags field to indicate whether SMC-D processing is or has been in effect. • When the NWMTCPSTSMCDCfg flag is set, the listed TCP counters reflect all TCP connections, including connections over SMC-D links. • The listed SMC-D statistics are added.	Shared Memory Communications - Direct Memory Access
GetIfs	• NWMIfFlags – NWMIfPNetIDFlg • NWMIfFlags2 – NWMIfSMCDFlg – NWMIfIsmAssoc • NWMIfType • NWMIfAssocName • NWMIfPFID • NWMIfSMCRStatus • NWMIfSMCDStatus • NWMIfGID • NWMIfPNetID	• New NWMIfSMCDFlg flag bit is set in the NWMIfFlags2 field for OSD and HiperSockets interfaces that have SMCD specified on the INTERFACE statement. • New NWMIfIsmAssoc is set in the NWMIfFlags2 field to indicate that this ISM is associated with an OSD or HiperSockets interface. • Listed fields are updated to include information for SMC-D. • The NWMIFSMCRVLAN value is obsolete from the NWMIfSMCRStatus field.	Shared Memory Communications - Direct Memory Access
GetIfs	• NWMIfFlags2 – NWMIFIQDCFLG • NWMIfIQDCName • NWMIfType – NWMIFTHIPERIQDC	• New flag bit NWMIFIQDCFLG indicates that NWMIfIQDCName contains the name of the associated HiperSockets IQDC Interface. • NWMIfIQDCName contains the name of associated HiperSockets IQDC interface. • There is a new value for NWMIfType (NWMIFTHIPERIQDC). This new value indicates that the interface being displayed is a HiperSockets Converged Interface.	HiperSockets Converged Interface support

Table 27. New Communications Server TCP/IP callable NMI (EZBNMIFR) for z/OS V2R3 (continued)

Request	Parameter/output	Description	Reason for change

GetIfStats	• NWMIfStFlags
	  – NWMIFSTIQDCFLG
	• NWMIfStType
	  – NWMIFTHIPERIQDC
	• NWMIfStInIQDCBytes
	• NWMIfStInIQDCUcastPkts
	• NWMIfStOutIQDCBytes
	• NWMIfStOutIQDCUcastPkts
		• New flag bit NWMIFSTIQDCFLG indicates that statistics for the associated HiperSockets IQDC interface are provided in NWMIfStIQDXStats area of this record.
		• There is a new value for NWMIfStType (NWMIFTHIPERIQDC). This new value indicates that the interface being displayed is a HiperSockets Converged Interface.
		• Input bytes received over associated HiperSockets IQDC interface.
		• Input unicast packets received over associated HiperSockets IQDC interface.
		• Output bytes sent over associated HiperSockets IQDC interface.
		• Output unicast packets sent over associated HiperSockets IQDC interface.
			HiperSockets Converged Interface support

GetIfStatsExtended	NWMIfStExtIType • NWMIFTHIPERIQDC	There is a new value for NWMIfStExtIType (NWMIFTHIPERIQDC). This new value indicates that the interface being displayed is a HiperSockets Converged Interface.	HiperSockets Converged Interface support
GetIsms	N/A	New poll-type request that obtains information for ISM interfaces.	Shared Memory Communications - Direct Memory Access
GetProfile	NMTP_GBCFAUTOIQDC	New byte NMTP_GBCFAUTOIQDC indicates the setting of AUTOIQDC in the GLOBALCONFIG statement.	HiperSockets Converged Interface support
GetProfile	Global configuration section: • NMTP_GBCFZERTINTV • NMTP_GBCFZERTSYNC	• New NMTP_GBCFZERTINTV flag bit that indicates the setting of the INTVAL sub-parameter on the GLOBALCONFIG statement. • New NMTP_GBCFZERTSYNC flag bit that indicates the setting of the SYNCVAL sub-parameter on the GLOBALCONFIG statement.	z/OS Encryption Readiness Technology (zERT) aggregation recording interval
GetProfile	Management section: • NMTP_MGMTSmf119Types • NMTP_MGMT119ZertSummary • NMTP_MGMTNetMonServices • NMTP_MGMTNMZertSummary	• New NMTP_MGMT119ZertSummary flag bit is set in the NMTP_MGMTSmf119Types field to indicate that the new zERT summary record was requested on the SMFCONFIG TYPE119 profile statement. • NMTP_MGMTNMZertSummary flag bit is set in the NMTP_MGMTNetMonServices field to indicate that the new zERT summary records were requested on the NETMONITOR profile statement.	z/OS Encryption Readiness Technology (zERT) aggregation
GetProfile	Global configuration section: • NMTP_GBCFPFport	Global Configuration section: NMTP_GBCPPFport represents the configured or learned port number used for its corresponding NMTP_GBCFPFid.	Communications Server support for RoCE Express2 features

Table 27. New Communications Server TCP/IP callable NMI (EZBNMIFR) for z/OS V2R3 (continued)

Request	Parameter/output	Description	Reason for change
GetProfile	Global configuration section: • NMTP_GBCFFlags • NMTP_GBCFZERT	New NMTP_GBCFZERT flag bit is set in the NMTP_GBCFFlags field to indicate that the zERT operand was specified on the GLOBALCONFIG statement.	z/OS Encryption Readiness Technology
	Management section: • NMTP_MGMTSmf119Types • NMTP_MGMT119Zert • NMTP_MGMTNetMonServices • NMTP_MGMTNMZert	• NMTP_MGMT119Zert flag bit is set in the NMTP_MGMTSmf119Types field to indicate that the new zERT connection detail record was requested on the SMFCONFIG profile statement. • NMTP_MGMTNMZert flag bit is set in the NMTP_MGMTNetMonServices field to indicate that the new zERT records were requested on the NETMONITOR profile statement.	z/OS Encryption Readiness Technology
	IPv4 configuration section: • NMTP_V4CFDynXcfSMCD	New NMTP_V4CFDynXcfSMCD value that indicates whether the dynamically generated XCF interface can be used for new TCP connections with SMC-D.	Shared Memory Communications - Direct Memory Access
	IPv6 configuration section: • NMTP_V6CFDynXcfSMCD	New NMTP_V6CFDynXcfSMCD value that indicates whether the dynamically generated XCF interface can be used for new TCP connections with SMC-D.	Shared Memory Communications - Direct Memory Access
	Global configuration section: • NMTP_GBCFFlags • NMTP_GBCFSMCD • NMTP_GBCFFixedMemoryD • NMTP_GBCFTcpKeepMinIntD	• New NMTP_GBCFSMCD flag bit is set in the NMTP_GBCFFlags field to indicate that the SMCD operand was specified on the GLOBALCONFIG statement. • New NMTP_GBCFFixedMemoryD field that specifies the SMCD FIXEDMEMORY value. FIXEDMEMORY is specified in megabyte increments. • New NMTP_GBCFTcpKeepMinIntD field that specifies the SMCD TCPKEEPMININTERVAL value.	Shared Memory Communications - Direct Memory Access
	Interface section: • NMTP_INTFFlags • NMTP_INTFSMCD	New NMTP_INTFSMCD flag bit is set in the NMTP_INTFFlags field for OSD and HiperSockets interfaces that have SMCD specified or that take the SMCD default on the INTERFACE statement.	Shared Memory Communications - Direct Memory Access
	Management section: • NMTP_MGMTSmf119Types • NMTP_MGMT119SmcDLnkStats • NMTP_MGMT119SmcDLnkEvent	• New NMTP_MGMT119SmcDLnkStats flag bit is set in the NMTP_MGMTSmf119Type field to indicate that the new SMC-D link statistics records were requested on the SMFCONFIG profile statement. • New NMTP_MGMT119SmcDLnkEvent flag bit is set in the NMTP_MGMTSmf119Type field to indicate that the new SMC-D link state start and end records were requested on the SMFCONFIG profile statement.	Shared Memory Communications - Direct Memory Access
	NMTP_GBCFSMCGFlags	New flag byte field NMTP_GBCFSMCGFlags with the following flag bits: • NMTP_GBCFAutoCache • NMTP_GBCFAutoSMC	• Shared Memory Communications over RDMA enhancements • Shared Memory Communications - Direct Memory Access
	NMTP_PORTRsvOptions	• New flag bit NMTP_PORTRSMC that indicates this port or port range is enabled for SMC-R and SMC-D. • New flag bit NMTP_PORTRNoSMC that indicates this port or port range is disabled for SMC-R and SMC-D.	• Shared Memory Communications over RDMA enhancements • Shared Memory Communications - Direct Memory Access
GetRnics	Base section: • NWMRnicBGen • NWMRnicBSpeed	Base section: • NWMRnicBGen represents the RNIC card generation level (IBM 10 GbE RoCE Express or IBM 10 GbE RoCE Express2) • NWMRnicBSpeed represents the transmission level for the RNIC.	Communications Server support for RoCE Express2 features

Table 27. New Communications Server TCP/IP callable NMI (EZBNMIFR) for z/OS V2R3 (continued)			
Request	**Parameter/output**	**Description**	**Reason for change**
GetSmcDLinks	N/A	New poll-type request to obtain information for SMC-D links.	Shared Memory Communications - Direct Memory Access
GetStorageStatistics	• NWMStgFlags – NWMStgZAGGCfg • New ZERT aggregation storage utilization fields – NWMStg64ZaggCurrent – NWMStg64ZaggMax	• New flag bit NWMStgZaggCfg indicates whether ZERT AGGREGATION is configured on the GLOBALCONFIG statement. • New fields NWMStg64ZaggCurrent and NWMStg64ZaggMax provide current and maximum storage usage statistics for ZERT AGGREGATION.	z/OS Encryption Readiness Technology (z/OS) aggregation recording interval
GetStorageStatistics	• NWMStgFlags – NWMStgSMCDCfg • New SMC-D storage utilization – NWMStg64SMCDFixedCurrent – NWMStg64SMCDFixedMax – NWMStg64SMCDFixedLimit	• New flag bit NWMStgSMCDCfg is set in the NWMStgFlags field to indicate whether the SMCD parameter is configured on the GLOBALCONFIG statement. • The SMC-D storage utilization information is added when the SMCD parameter is configured on the GLOBALCONFIG statement	Shared Memory Communications - Direct Memory Access
GetTCPListeners	• NWMTCPLSmcdCfg • NWMTCPLSmcdCurrConn • NWMTCPLSmcdTotalConn	• New NWMTCPLSmcdCfg flag set in the NWMTCPLSmcFlags field that indicates whether the SMC-D processing is or has been in effect. • New field NWMTCPLSmcdCurrConn that indicates the number of active connections to this server that use SMC-D. • New field NWMTCPLSmcdTotalConn that indicates the number of connections that this server has accepted using SMC-D.	Shared Memory Communications - Direct Memory Access

TCPIPCS subcommand

This topic describes the Communications Server TCPIPCS subcommand option changes for z/OS V2R3.

Table 28 on page 143 lists the TCPIPCS subcommand options.

The TCPIPCS command contains the OPTLOCAL specification in some displays.

Table 28. New and changed Communications Server TCPIPCS subcommand options for z/OS V2R3		
Subcommand	**Description**	**Reason for change**
CONFIG	Includes the new HiperSockets Converged Interfaces in the output.	HiperSockets Converged Interface support
PROFILE	Displays the new INTVAL and SYNCVAL sub-parameters for the GLOBALCONFIG statement.	z/OS Encryption Readiness Technology (zERT) aggregation recording interval
PROFILE	Displays the AUTOIQDC parameter.	HiperSockets Converged Interface support
PROFILE	Displays the current TCP/IP stack configuration from information in the dump by creating the profile statements that represent the configuration. See PROFILE.TCPIP statement and parameter changes for information about the profile statement changes for V2R3.	Release update

Table 28. New and changed Communications Server TCPIPCS subcommand options for z/OS V2R3 (continued)		
Subcommand	**Description**	**Reason for change**
STATE	Includes the new HiperSockets Converged Interfaces in the output.	HiperSockets Converged Interface support
TREE	Includes the new HiperSockets Converged Interfaces in the output.	HiperSockets Converged Interface support

New and changed System Management Facilities (SMF) records for z/OS V2R3

This topic lists the System Management Facilities (SMF) records for z/OS elements and features that are new or changed for z/OS V2R3.

Table 29 on page 144 provides a list of SMF records that are new or changed for z/OS V2R3. Detailed information about these SMF records can be found in one of the following documents, as indicated by the "Where documented" column in Table 29 on page 144.

Communications Server

- in *z/OS Communications Server: IP Programmer's Guide and Reference*
- in *z/OS Communications Server: SNA Network Implementation Guide*

MVS
z/OS MVS System Management Facilities (SMF)

OAM
in *z/OS DFSMS OAM Planning, Installation, and Storage Administration Guide for Object Support*

RACF
in *z/OS Security Server RACF Macros and Interfaces*

Table 29. New and changed System Management Facilities (SMF) records for z/OS V2R3			
SMF record	**z/OS element or feature**	**Description**	**Where documented**
Type 14/15	DFSMS	• APAR OA56622 added byte 2 (SMF14DSENCNP) of the SMF14DEF field and the SMF14FLGS/SMF14FLG1 field in the DASD Data Set Encryption Information (Type 9) section. • New DASD data set encryption section added. • Updated flags in SMF14RFG1 field. • Updated length to 1 (type 14) for SFM14RV3.	MVS
Type 23	IOS	New fields at offsets 88 and 93.	MVS

Table 29. New and changed System Management Facilities (SMF) records for z/OS V2R3 (continued)			
SMF record	**z/OS element or feature**	**Description**	**Where documented**
Type 30	IOS	• APAR OA59126 defined bit 5 of field SMF30SFL and added fields SMF30NumberOfDataSpacesHWM and SMF30UserDataSpaceCreateReqCount in the Storage and Paging section. • APAR OA59998 clarified the description of the SMF30SCC field in the Completion Section. • The description for byte 0 of the SMF30CAS_OA54589 field has been updated. • New fields are added at offsets 180 and 181.	MVS
Type 41	BCP	New fields at offsets 40, 48, 52, 56, and 60.	MVS
	BCP	• The description of the SMF42PSV field has been updated in the Product Section. • The descriptions of fields SMF42274 - SMF42276 have been updated, and fields SMF42277 - SMF32279 have been added in the Header/Self-defining section.	MVS
Type 42 Subtype 5	DFSMS	**APAR :** OA55711 adds new fields in the volume header section and adds the following new sections: • Volume metrics section • System I/O section • System I/O statistics section • System I/O high response time section • Volume background activity • Volume cloud activity	MVS
Type 42 Subtype 6	DFSMS	• APAR OA57718 added new fields in Synchronous I/O Section 2 and Synchronous I/O Section 3. • New fields, S42SNTWH, S42SNTDR, S42SNTDX, S42DS2FL and S42DS2DL, are added. • APAR OA55711 added new fields in the data set I/O statistics section.	MVS

Table 29. New and changed System Management Facilities (SMF) records for z/OS V2R3 (continued)

SMF record	z/OS element or feature	Description	Where documented
Type 50, Record for RoCE connection	Communications Server	• Added new fields for RoCE port and RoCE user records: **Offset 26 (36 bytes)** Reserved **Offset 62 (1 byte)** Extension length (including this field) **Offset 63 (1 byte)** Attachment type **Offset 64 (2 bytes)** Version **Offset 66 (16 bytes)** Reserved • Added new fields for RoCE port records: **Offset 82 (4 bytes)** Number of RoCE users **Offset 86 (4 bytes)** Poll EQ overflow **Offset 90 (4 bytes)** Poll EQ count **Offset 94 (4 bytes)** Poll EQ entries overflow **Offset 98 (4 bytes)** Poll EQ entries count **Offset 102 (4 bytes)** PCI real interrupts overflow **Offset 106 (4 bytes)** PCI real interrupts count **Offset 110 (4 bytes)** Unproductive PCI overflow **Offset 114 (4 bytes)** Unproductive PCI count • Added new field for RoCE user records: **Offset 118 (8 bytes)** ULP ID • Changed the offsets for the existing RoCE user record fields at offset 26 (120 bytes). Fields were shifted 100 bytes. The new offsets are 126 through 242. **Reason for change:** OA57381	Communications Server
Type 62	DFSMS	New DASD fields identify the data set as encrypted	MVS

Table 29. New and changed System Management Facilities (SMF) records for z/OS V2R3 (continued)			
SMF record	**z/OS element or feature**	**Description**	**Where documented**
Type 70 to 79: RMF Product Section	RMF	**SMF7xPRF** Existing bit 6 renamed as *Enhanced DAT facility 1 available* SMF7xPED New flag bit 7 *Enhanced DAT facility 2 available* SMF7xPE2	MVS
Type 70 Subtype 2: Cryptographic Hardware Activity	RMF (Processor Activity)	• APAR OA59330 added new FFX measurement fields in the ICSF Services Data section. • New cryptographic processor types are added.	MVS
Type 71	RMF	New fields added to the Paging Data Section	MVS
Type 72 Subtype 3: Workload Activity	RMF	Updated descriptions for R723CAMU and R723CAMD. Updated descriptions for R723MFLG and R723GGLT. Added new field R723GGML New fields R723CTET2, R723CXET2, R723CETS2, R723CTETX, R723CXETX, R723CETSX, R723CQDTX, R723CADTX, R723CCVTX, R723CIQTX	MVS
Type 72 Subtype 4: Storage Activity	RMF	Description changes and new fields: R724ETX and R724QTX.	MVS
Type 74 Subtype 1:	RMF	Updated description for fields SMF74CN2, SMF74AGC, SMF74AGS, SMF74PRF. Added new fields: SMF74SBR, SMF74SBW, SMF74SQR, SMF74SQW, SMF74SPR, SMF74SPW, SMF74SFTR. SMF74SFTW, SMF74SLBR, SMF74SLBW, SMF74SCMR, SMF74SNIS, SMF74STOR, SMF74STOW, SMF74SOR, SMF74SOW, SMF74IOS.	MVS
Type 74 Subtype 4	RMF	• APARs OA58729 and OA58724 added new fields in the Request Data Section. • Updated length to 1 for the R744QFLG field. • Updated status flag descriptions. Added new fields: R744SWDR, R744SWAC, R744SRDR, R744SRAC, R744SWEC, R744SREC, R744SWED, R744SWES, R744SRED, R744SRES	MVS
Type 74 Subtype 5	RMF	Updated status flag descriptions.	MVS

Table 29. New and changed System Management Facilities (SMF) records for z/OS V2R3 (continued)			
SMF record	**z/OS element or feature**	**Description**	**Where documented**
Type 74 Subtype 9	RMF	Updated status flag descriptions for R749LOOP , R749STOP , R749SBOP , R749RF08. Added new fields: R749PFT, SMF749SO, SMF749SL, SMF749SN, R749PORT, R749PFT, R749WWNN, R749SIOO, R749SION, R749RTDO, R749RTDN New Synchonous I/O Link Data section was added with field R749SND. New Synchonous I/O Response Time Distribution Data section was added with fields R749RFLG, R749RTSC and R749RTRV.	MVS
Type 74 Subtype 10	RMF	Added new fields: R7410CWUC, R7410CWU, R7410FLG	MVS
Type 78 Subtype 2	RMF	New fields added to the Private Area Summary Data Section R782PVT	MVS
Type 78 Subtype 3	BCP	Bit 2 is defined in the R783GFLX field.	MVS
Type 80	RACF	• Offset 80 updated for FMID, 77B0. • New relocate, 43(2B), added for class name from SETROPTS GENLIST/ NOGENLIST. • New extended-length relocate, 358(166), added for certificate requester name. • The MFA extended-length relocate, 440, is updated and new relocate, 444, is added to contain new ALTUSER subkeyword for CICS auditing support • New extended-length relocate, 445, added to support CICS client identity capability.	RACF
Type 84	JES2	New subtype 21 for JES2 fields SMF84JRU DSECT, R84MEMJ2, R84RSUJ2.	MVS
Type 85 (X'55')	DFSMS OAM	New fields R85POSUB and R85PSSID.	OAM
Type 90 Subtype 40	MVS	• APAR OA59813 updated the SMF90T40_Event, SMF90T40_Flags0, and SMF90T40_Flags1 fields. • APAR OA59813 added the SMF90T40_RP_Start_Requestor_ID and SMF90T40_RP_Duration fields.	MVS
Type 92 Subtype 8	z/OS UNIX	New subtype record that is produced when the target file system is mounted during file system migration.	MVS
Type 92 Subtype 50	zFS	New subtype; records file system events.	MVS

Table 29. New and changed System Management Facilities (SMF) records for z/OS V2R3 (continued)

SMF record	z/OS element or feature	Description	Where documented
Type 92, Subtype 51	zFS	New subtype; records zFS call counts.	MVS
Type 92, Subtype 52	zFS	New subtype; contains statistics for the zFS user file cache.	MVS
Type 92, Subtype 53	zFS	New subtype; contains statistics for the zFS metadata cache.	MVS
Type 92, Subtype 54	zFS	New subtype; contains zFS locking and sleeps statistics, including the most highly contented locks.	MVS
Type 92, Subtype 55	zFS	New subtype; contains general disk I/O statistics for zFS.	MVS
Type 92, Subtype 56	zFS	New subtype; contains information about the token manager.	MVS
Type 92, Subtype 57	zFS	New subtype; details zFS use of memory, with total bytes allocated to each zFS subcomponent	MVS
Type 92, Subtype 58	zFS	New subtype; contains per-file system usage.	MVS
Type 92, Subtype 59	zFS	New subtype; transmit/receive statistics for the sysplex.	MVS
Type 98	BCP	New fields added to the Identification section, Context summary section, and Subtype 1 Environmental section.	MVS
Type 99, Subtype 1 records	WLM	• APAR OA59366 added the SMF99_BOOSTINFO field in the System State Information section. • New fields added to the Software Licensing Information section.	MVS
Type 99, Subtype 2 records	WLM	• APAR OA59366 updated the SMF99FLAGS2 field in the Period Data section. • New fields added to the Address Space Expanded Storage Access Policy section.	MVS
Type 99, Subtype 12 records	WLM	Change to the SMF99C_VCM_DIAGCAPDECR_CONT field in the Capacity Data section.	MVS
Type 117	IOS	Type changed for WSAP and IIB.	MVS

Table 29. New and changed System Management Facilities (SMF) records for z/OS V2R3 (continued)			
SMF record	**z/OS element or feature**	**Description**	**Where documented**
Type 119, TCP connection termination record (subtype 2)	Communications Server	• New IP filter section. Indicates the inbound and outbound IP filters associated with this connection. • New SMF119AP_TTIPsecurityFlags field. Indicates whether IP security is enabled for this TCP/IP stack and whether IP filtering was done for this connection. **Reason for change:**SMF 119 TCP connection termination record (subtype 2) enhanced to provide IP filter information	Communications Server

Table 29. New and changed System Management Facilities (SMF) records for z/OS V2R3 (continued)

SMF record	z/OS element or feature	Description	Where documented
Type 119, TCP connection termination record (subtype 2)	Communications Server	• New SMF119AP_TTSMCDStatus field. Indicates whether an SMC-D link that is established for this connection is active or inactive. If SMF119AP_TTSMCDReason is also X'0000', SMC-D link establishment has not been attempted. • New SMF119AP_TTSMCDReason field. Indicates why the connection does not use an SMC-D link. • New flag bits in the SMF119AP_TTSMCFlags field: – New SMF119AP_TTSMCDRSNRMT flag bit. Indicates that the peer sets the SMC-D reason value in the SMF119AP_TTSMCDReason field. – New SMF119AP_TTSMCDCACHED flag bit. Indicates that this route cached to not use SMC-D. • New SMF119AP_TTLclSMCLinkId field. Indicates the local stack link ID for the SMC-D or SMC-R link that this connection traverses. • New SMF119AP_TTRmtSMCLinkId field. Indicates the remote stack link ID for the SMC-D or SMC-R link that this connection traverses. • New SMF119AP_TTLclSMCBufSz field that indicates the size of the RMB or DMB element that is used by the local host for receiving data on this connection from the remote host. • New SMF119AP_TTRmtSMCBufSz field. Indicates the size of the RMB or DMB element that is used by the remote host for receiving data on this connection from the local host. **Reason for change:** Shared Memory Communications - Direct Memory Access	Communications Server
Type 119, TCP connection termination record (subtype 2)	Communications Server	New SMF119AP_TTTTLSFP values **Reason for change:** AT-TLS currency with System SSL	Communications Server

Table 29. New and changed System Management Facilities (SMF) records for z/OS V2R3 (continued)			
SMF record	**z/OS element or feature**	**Description**	**Where documented**
Type 119, FTP client transfer completion record (subtype 3)	Communications Server	• New field SMF119S6Off. Contains the offset to FTP client load module name section. • New field SMF119S6Len. Contains the Length of FTP client load module name section. • New field SMF119S6Num. Contains the number of FTP client load module name sections. • New field SMF119FT_FCMemNum. Contains the number of members associated with the file transfer operation • New field SMF119FT_FCLibNameLen. Contains the length of the name of the load module or program object library associated with the file transfer operation • New field SMF119FT_FCLibName. Contains the name of the load module or program object library associated with the file transfer operation • New field SMF119FT_FCMemName. Contains the names of the members associated with the file transfer operation. There is no delimiter between the names of members. Each member name occupies 8 bytes with trailing blanks padded.	Communications Server
Type 119, FTP client transfer completion record (subtype 3)	Communications Server	Updated the SMF119FT_FCFips140 field to support FIPS140 Level1, Level2, and Level3. **Reason for change:** AT-TLS currency with System SSL	Communications Server
Type 119, TCP/IP profile event record (subtype 4)	Communications Server	Global configuration section • The new NMTP_GBCFAutoIQDC byte is set to indicate the setting of AUTOIQDC specified on the GLOBALCONFIG statement. **Reason for change:** HiperSockets Converged Interface support	Communications Server

Table 29. New and changed System Management Facilities (SMF) records for z/OS V2R3 (continued)

SMF record	z/OS element or feature	Description	Where documented
Type 119, TCP/IP Profile event record (subtype 4)	Communications Server	• NMTP_GBCFZERTAGG indicates that the AGGREGATION subparameter was specified on the GLOBALCONFIG ZERT profile statement. • NMTP_MGMT119ZertSummary indicates that ZERTSUMMARY was specified on the SMFCONFIG TYPE119 profile statement. • NMTP_MGMTNMZertSummary indicates that ZERTSUMMARY was specified on the NETMONITOR profile statement. **Reason for change:** z/OS Encryption Readiness Technology (zERT) aggregation	Communications Server
Type 119, TCP/IP profile event record (subtype 4)	Communications Server	• New flag NMTP_GBCFZERT in the NMTP_GBCFFlags field in the global configuration section. • New flag NMTP_MGMT119ZertDetail in the NMTP_MGMTSmf119Types field in the management section. • New flag NMTP_MGMTNMZert in the NMTP_MGMTNetMonServices field in the management section. **Reason for change:** z/OS Encryption Readiness Technology	Communications Server

Table 29. New and changed System Management Facilities (SMF) records for z/OS V2R3 (continued)			
SMF record	**z/OS element or feature**	**Description**	**Where documented**
Type 119, TCP/IP profile event record (subtype 4)	Communications Server	IPv4 configuration section • The new NMTP_V4CFDynXcfSMCD value is specified whether the dynamically generated XCF interface can be used for new TCP connections with SMC-D. IPv6 configuration section • The new NMTP_V6CFDynXcfSMCD value is specified whether the dynamically generated XCF interface can be used for new TCP connections with SMC-D. Global configuration section: • The new NMTP_GBCFSMCD flag bit is set in the NMTP_GBCFFlags field to indicate that the SMCD operand was specified on the GLOBALCONFIG statement. • The new NMTP_GBCFFixedMemoryD field specifies the SMCD FIXEDMEMORY value. FIXEDMEMORY is specified in megabyte increments. • The new NMTP_GBCFTcpKeepMinIntD field specifies the SMCD TCPKEEPMININTERVAL value. Interface section: • The new NMTP_INTFSMCD flag bit is set in the NMTP_INTFFlags field for OSA or HiperSocket interfaces that have SMCD specified or that take the SMCD default on the INTERFACE statement. Management section • New NMTP_MGMT119SmcDLnkStats flag bit in the NMTP_MGMTSmf119Type field. Indicates that the new SMC-D link statistics records were requested on the SMFCONFIG profile statement. • New NMTP_MGMT119SmcDLnkEvent flag bit in the NMTP_MGMTSmf119Type field. Indicates that the new SMC-D link state start and end records were requested on the SMFCONFIG profile statement. **Reason for change:** Shared Memory Communications - Direct Memory Access	Communications Server

Table 29. New and changed System Management Facilities (SMF) records for z/OS V2R3 (continued)

SMF record	z/OS element or feature	Description	Where documented
Type 119, TCP/IP profile event record (subtype 4)	Communications Server	New flag byte field NMTP_GBCFSMCGFlags with the following flag bits: • NMTP_GBCFAutoCache • NMTP_GBCFAutoSMC **Reason for change:** • Shared Memory Communications over RDMA enhancements • Shared Memory Communications - Direct Memory Access	Communications Server
Type 119, TCP/IP profile event record (subtype 4)	Communications Server	Global Configuration section: • NMTP_GBCPPFport represents the configured or learned port number used for its corresponding NMTP_GBCFPFid. **Reason for change:** Communications Server support for RoCE Express2 features	Communications Server
Type 119, TCP/IP profile event record (subtype 4)	Communications Server	New flag bits NMTP_PORTRSMC and NMTP_PORTRNoSMC in field NMTP_PORTRsvOptions. **Reason for change:** • Shared Memory Communications over RDMA enhancements • Shared Memory Communications - Direct Memory Access	Communications Server
Type 119, TCP/IP profile event record (subtype 4)	Communications Server	Global Configuration section: New field NMTP_GBCFzAGGtim_INTVAL indicates the setting of the INTVAL sub-parameter on the GLOBALCONFIG statement. New field NMTP_GBCFzAGGtim_SYNCVAL indicates the setting of the SYNCVAL sub-parameter on the GLOBALCONFIG statement. • NMTP_GBCFzAGGtim_SYNCVAL_HH is the hours part of SYNCVAL (format of 24-hour military time). • NMTP_GBCFzAGGtim_SYNCVAL_MM is the minutes portion of SYNCVAL. **Reason for change:** z/OS Encryption Readiness Technology (zERT) aggregation recording interval	Communications Server

Table 29. New and changed System Management Facilities (SMF) records for z/OS V2R3 (continued)

SMF record	z/OS element or feature	Description	Where documented
Type 119, TCP/IP statistics record (subtype 5)	Communications Server	New fields SMF119AP_TSSTZAGGCurrent and SMF119AP_TSSTZAGGMax, in the TCP statistics section, provide current and maximum storage usage statistics for ZERT AGGREGATION. **Reason for change:** z/OS Encryption Readiness Technology (zERT) aggregation recording interval	Communications Server
Type 119, TCP/IP statistics record (subtype 5)	Communications Server	• When the SMCD parameter is configured on the GLOBALCONFIG statement, the following TCP counters reflect all TCP connections, including connections over SMC-D links. The following existing TCP stats are updated: SMF119AP_TSTCEstab SMF119AP_TSTCOpenConn SMF119AP_TSTCPassConn SMF119AP_TSTCConCls SMF119AP_TSTCInSegs SMF119AP_TSTCOSegs SMF119AP_TSTCReset SMF119AP_TSTCConReset SMF119AP_TSTCOKApr SMF119AP_TSTCDropKA SMF119AP_TSTCDropF2 • The following new SMC-D stats are added at the end of the TCP statistics section. SMF119AP_TSSMCDCurrEstabLnks SMF119AP_TSSMCDActLnkOpened SMF119AP_TSSMCDPasLnkOpened SMF119AP_TSSMCDLnksClosed SMF119AP_TSSMCDCurrEstab SMF119AP_TSSMCDActiveOpened SMF119AP_TSSMCDPassiveOpened SMF119AP_TSSMCDConnClosed SMF119AP_TSSMCDInSegs SMF119AP_TSSMCDOutSegs SMF119AP_TSSMCDInRsts SMF119AP_TSSMCDOutRsts • The following new SMC-D storage stats are added in the storage statistics section. SMF119AP_TSSTSMCDFixedCurrent SMF119AP_TSSTSMCDFixedMax **Reason for change:** Shared Memory Communications - Direct Memory Access	Communications Server

Table 29. New and changed System Management Facilities (SMF) records for z/OS V2R3 (continued)

SMF record	z/OS element or feature	Description	Where documented
Type 119, Interface statistics record (subtype 6)	Communications Server	• SMF119IS_IFDesc has two new values: IPAQIQDC and IPAQIQDC6. • New SMF119IS_IFIQDCFLAG bit is set in the SMF119IS_IFFlags field to indicate that IQDC flag indicates the associated name and the associated statistics are for IQDC, not IQDX. • SMF119IS_IFIQDCName is the associated IQDC interface name for IPAQENET and IPAQENET6 interfaces that are defined with CHPIDTYPE OSD and with an associated IQDC interface. • SMF119IS_IFInIQDCBytes indicates the number of inbound bytes that were received over the associated IQDC interface. • SMF119IS_IFInIQDCUniC indicates the number of inbound unicast packets that were received over the associated IQDC interface. • SMF119IS_IFOutIQDCBytes indicates the number of outbound bytes that were sent over the associated IQDC interface. • SMF119IS_IFOutIQDCUniC indicates the number of outbound unicast packets that were sent over the associated IQDC interface. **Reason for change:** HiperSockets Converged Interface support	Communications Server
Type 119, Interface statistics record (subtype 6)	Communications Server	• New SMF119IS_IFSMCDFLGED flag bit is set in the SMF119IS_IFFlags field for OSA or HiperSocket interfaces. Contains information that is related to the SMC-D characteristics. • Updated SMF119IS_IFPNetID field. Contains the Physical network ID for active OSD, OSX, RNIC, ISM interfaces. **Reason for change:** Shared Memory Communications - Direct Memory Access	Communications Server
Type 119, z/OS Encryption Readiness Technology record (subtype 11)	Communications Server	New subtype record to report zERT connection details **Reason for change:** z/OS Encryption Readiness Technology	Communications Server

Table 29. New and changed System Management Facilities (SMF) records for z/OS V2R3 (continued)

SMF record	z/OS element or feature	Description	Where documented
Type 119, zERT summary record (subtype 12)	Communications Server	New SMF 119 subtype to report z/OS Encryption Readiness Technology (zERT) aggregation function records. **Reason for change:** z/OS Encryption Readiness Technology (zERT) aggregation	Communications Server
Type 119, SMC-D link statistics record (subtype 38)	Communications Server	New SMC-D link statistics record. Provides statistics about SMC-D links. **Reason for change:** Shared Memory Communications - Direct Memory Access	Communications Server
Type 119, SMC-D link state start record (subtype 39)	Communications Server	New SMC-D link state start record. Provides statistics for an SMC-D link at the time that the link is started. **Reason for change:** Shared Memory Communications - Direct Memory Access	Communications Server
Type 119, SMC-D link state end record (subtype 40)	Communications Server	New SMC-D link state end record. Provides statistics for an SMC-D link at the time that the link is ended. **Reason for change:** Shared Memory Communications - Direct Memory Access	Communications Server
Type 119,RNIC interface statistics record (subtype 44)	Communications Server	• SMF119SM_RSGen. Represents the RNIC card generation level. • SMF119SM_RSSpeed. Represents the transmission level for the RNIC **Reason for change:** Communications Server support for RoCE Express2 features	Communications Server
Type 119, Internal shared memory (ISM) interface statistics record (subtype 45)	Communications Server	New Internal shared memory (ISM) interface statistics record. Provides statistics about ISM interfaces. **Reason for change:** Shared Memory Communications - Direct Memory Access	Communications Server
Type 119, CSSMTP configuration record (CONFIG subtype 48)	Communications Server	• New field SMF119ML_CF_AtSign. Specifies value of AtSign option. • New field SMF119ML_CF_TLSEhlo. Specifies value of TLSEhlo option. • New field SMF119ML_TS_Charset. Specifies value of target server code page. **Reason for change:** CSSMTP mail compatibility enhancements	Communications Server

Table 29. New and changed System Management Facilities (SMF) records for z/OS V2R3 (continued)

SMF record	z/OS element or feature	Description	Where documented
Type 119, CSSMTP Connection (CONNECT subtype 49)	Communications Server	New field SMF119ML_CN_TLSSFP. Updates to support FIPS140 Level1, Level2, and Level3 **Reason for change:** AT-TLS currency with System SSL	Communications Server
Type 119, FTP server transfer completion record (subtype 70)	Communications Server	• New field SMF119S6Off. Contains the offset to FTP server load module name section. • New field SMF119S6Len. Contains the Length of FTP server load module name section. • New field SMF119S6Num. Contains the number of FTP server load module name sections. • New field SMF119FT_FSMemNum. Contains the number of members associated with the file transfer operation • New field SMF119FT_FSLibNameLen. Contains the length of the name of the load module or program object library associated with the file transfer operation • New field SMF119FT_FSLibName. Contains the name of the load module or program object library associated with the file transfer operation. • New field SMF119FT_FSMemName. Contains the names of the members associated with the file transfer operation. No delimiter between the names of members. Each member name occupies 8 bytes with trailing blanks padded.	Communications Server
Type 119, FTP server transfer completion record (subtype 70)	Communications Server	Updated the SMF119FT_FSFips140 field to support FIPS140 Level1, Level2, and Level3. **Reason for change:** AT-TLS currency with System SSL	Communications Server
Type 119, FTP server login failure record (subtype 72)	Communications Server	Updated the SMF119FT_FFFips140 field to support FIPS140 Level1, Level2, and Level3. **Reason for change:** AT-TLS currency with System SSL	Communications Server
Type 119, VTAM 3270 intrusion detection services record (subtype 81)	Communications Server	New record in response to VTAM 3270 intrusion detection services **Reason for change:** VTAM 3270 intrusion detection services	Communications Server

Table 29. New and changed System Management Facilities (SMF) records for z/OS V2R3 (continued)			
SMF record	**z/OS element or feature**	**Description**	**Where documented**
Type 119, FTP transfer initialization record (subtype 100)	Communications Server	Updated the SMF119FT_FSFips140 field to support FIPS140 Level1, Level2, and Level3. **Reason for change:** AT-TLS currency with System SSL	Communications Server
Type 119, FTP client transfer initialization record (subtype 101)	Communications Server	Updated the SMF119FT_FCFips140 field to support FIPS140 Level1, Level2, and Level3. **Reason for change:** AT-TLS currency with System SSL	Communications Server
Type 119, FTP client login failure record (subtype 102)	Communications Server	Updated the SMF119FT_FCFips140 field to support FIPS140 Level1, Level2, and Level3. **Reason for change:** AT-TLS currency with System SSL	Communications Server
Type 119, FTP client session record (subtype 103)	Communications Server	Updated the SMF119FT_FCFips140 field to support FIPS140 Level1, Level2, and Level3. **Reason for change:** AT-TLS currency with System SSL	Communications Server
Type 119, FTP server session record (subtype 104)	Communications Server	Updated the SMF119FT_FSFips140 field to support FIPS140 Level1, Level2, and Level3. **Reason for change:** AT-TLS currency with System SSL	Communications Server
Type 122, IBM Explorer for z/OS and dependent products	IBM Explorer for z/OS	New record type 122. New subtype 1 for IBM Developer for IBM Z®.	MVS
Type 124	IOS	Updated offsets 22, 24, and 140 for subtype 1. New offset 148.	MVS
Type 125	IOS	New type; data persistence.	MVS